



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

J.162

(03/2001)

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

IPCablecom

**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.162

RECOMENDACIONES UIT-T DE LA SERIE J

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

Recomendaciones generales	J.1–J.9
Especificaciones generales para transmisiones radiofónicas analógicas	J.10–J.19
Características de funcionamiento de los circuitos radiofónicos	J.20–J.29
Equipos y líneas utilizados para circuitos radiofónicos analógicos	J.30–J.39
Codificadores digitales para señales radiofónicas analógicas	J.40–J.49
Transmisión digital de señales radiofónicas	J.50–J.59
Circuitos para transmisiones de televisión analógica	J.60–J.69
Transmisiones de televisión analógica por líneas metálicas e interconexión con radioenlaces	J.70–J.79
Transmisión digital de señales de televisión	J.80–J.89
Servicios digitales auxiliares para transmisiones de televisión	J.90–J.99
Requisitos operacionales y métodos para transmisiones de televisión	J.100–J.109
Sistemas interactivos para distribución de televisión digital	J.110–J.129
Transporte de señales MPEG-2 por redes de transmisión de paquetes	J.130–J.139
Mediciones de la calidad de servicio	J.140–J.149
Distribución de televisión digital por redes locales de abonados	J.150–J.159
IPCablecom	J.160–J.179
Varios	J.180–J.199
Aplicación para televisión digital interactiva	J.200–J.209

Para más información, véase la Lista de Recomendaciones del UIT-T.

Recomendación UIT-T J.162

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

Resumen

Gran número de operadores de televisión por cable están mejorando sus facilidades para proporcionar capacidades bidireccionales y utilizar estas capacidades en la prestación de servicios de datos IP de velocidad elevada según UIT-T J.83 y UIT-T J.112. Estos operadores desean ahora ampliar la capacidad de esta plataforma de entrega para que incluya varios servicios dependientes del tiempo. Esta Recomendación es una de las Recomendaciones de una serie requerida para la consecución de este objetivo. Proporciona un protocolo de señalización de llamada basado en la red necesario para el establecimiento de conexiones.

Orígenes

La Recomendación UIT-T J.162, preparada por la Comisión de Estudio 9 (2001-2004) del UIT-T, fue aprobada por el procedimiento de la Resolución 1 de la AMNT el 9 de marzo de 2001.

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.

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 2002

Es propiedad. Ninguna parte de esta publicación puede reproducirse o utilizarse, de ninguna forma o por ningún medio, sea éste electrónico o mecánico, de fotocopia o de microfilm, 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	2
3	Términos y definiciones.....	2
4	Abreviaturas.....	2
5	Introducción	3
5.1	Relación con las normas H.323	4
5.2	Relación con las normas IETF	4
6	Interfaz de control de pasarela de medios (MGCI).....	5
6.1	Modelo y convenios de denominación	5
6.1.1	Nombres de los puntos extremos.....	5
6.1.2	Nombres de las llamadas	7
6.1.3	Nombres de las conexiones	8
6.1.4	Nombres de agentes de llamada y otras entidades	8
6.1.5	Mapas de dígitos.....	9
6.1.6	Eventos y señales.....	11
6.2	Utilización del SDP	12
6.3	Funciones de control de pasarela.....	13
6.3.1	NotificationRequest (petición de notificación).....	15
6.3.2	Notificaciones.....	21
6.3.3	CreateConnection (crear conexión).....	21
6.3.4	ModifyConnection (modificar conexión).....	27
6.3.5	DeleteConnection (suprimir conexión) (desde el agente de llamada).....	29
6.3.6	DeleteConnection (suprimir conexión) (desde el cliente incorporado).....	31
6.3.7	DeleteConnection (suprimir conexión) (suprimir múltiples conexiones desde el agente de llamada).....	31
6.3.8	Auditing (auditoría).....	32
6.3.9	Restart in Progress (rearranque en curso).....	36
6.4	Estados, condiciones de cambio-por-fallo y condiciones de carrera	37
6.4.1	Recapitulaciones y puntos notables.....	37
6.4.2	Retransmisión y detección de asociaciones perdidas	38
6.4.3	Condiciones de carrera	41
6.5	Códigos de retorno y códigos de error.....	48
6.6	Códigos de motivo	49
7	Protocolo de control de pasarela de medios.....	49

	Página
7.1 Descripción general	50
7.2 Encabezamiento de instrucción	50
7.2.1 Línea de instrucción.....	50
7.2.2 Líneas de parámetro.....	52
7.3 Formatos del encabezamiento de respuesta	63
7.3.1 CreateConnection	65
7.3.2 ModifyConnection.....	65
7.3.3 DeleteConnection	66
7.3.4 NotificationRequest.....	66
7.3.5 Notify.....	66
7.3.6 AuditEndpoint	66
7.3.7 AuditConnection.....	66
7.3.8 RestartInProgress.....	67
7.4 Codificación de la descripción de sesión.....	67
7.4.1 Utilización del servicio audio de SDP.....	67
7.4.2 Utilización del servicio vídeo SDP.....	73
7.5 Transmisión por UDP	73
7.5.1 Entrega de mensajes fiable	73
7.5.2 Estrategia de retransmisión.....	74
7.6 Transporte en remolque	75
7.7 Identificadores de transacción y toma de contacto de tres caminos	75
7.8 Respuestas provisionales	76
8 Seguridad	78
Anexo A – Lotes de eventos	78
Anexo B – Calidad de servicio dinámica.....	80
Apéndice I – Ejemplo de lote de eventos	88
Apéndice II – Ejemplos de codificaciones de instrucciones.....	89
Apéndice III – Ejemplo de flujo de llamada.....	96
Apéndice IV – Interacciones de modo.....	103
Apéndice V – Información de compatibilidad.....	107
Apéndice VI – Otros ejemplos de lotes de eventos	108

Recomendación UIT-T J.162

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

1 Alcance

Esta Recomendación describe un perfil de una interfaz de programación de aplicaciones, la interfaz de controlador de pasarela de medios (MGCI, *media gateway controller interface*), y un protocolo correspondiente, el protocolo de control de pasarela de medios (MGCP, *media gateway control protocol*), que se ocupa del control de clientes incorporados de voz sobre el protocolo Internet (VoIP, *voice-over-IP*) desde elementos de control de llamada externos. El MGCP supone una arquitectura de control de llamada donde la "inteligencia" del control de la llamada es exterior a las pasarelas y es manejada por elementos de control de llamada externos. El perfil, que se describe en este documento, se denominará protocolo de señalización de llamada basada en la red (NCS, *network-based call signalling*).

Esta Recomendación se basa en el protocolo de control de pasarela de medios (MGCP) 1.0 RFC 2705, que es el resultado de una combinación del protocolo de control de pasarela simple y de la familia de protocolos de control de dispositivos IP (IPDC, *IP device control*), así como de las aportaciones de las personas que han desarrollado este perfil.

2 Referencias

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.

2.1 Referencias normativas

- UIT-T J.83 (1997), *Sistemas digitales multiprogramas para servicios de televisión, sonido y datos de distribución por cable*.
- UIT-T J.112 Anexo A (2001), *Difusión de vídeo digital: Canal de interacción para sistemas de distribución de televisión por cable en difusión de vídeo digital*.
- UIT-T J.112 Anexo B (2001), *Especificaciones de interfaces de servicios de datos por cable: Especificación de la interfaz de radiofrecuencia*.
- UIT-T J.160 (Proyecto), *Marco arquitectural para la prestación de servicios dependientes del tiempo por redes de televisión por cable que utilizan módems de cable*.
- UIT-T J.161 (2001), *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*.
- UIT-T J.163 (2001), *Calidad de servicio dinámica para prestación de servicios dependientes del tiempo por redes de televisión por cable que utilizan módems de cable*.
- IETF RFC 821 (1982), *Simple Mail Transfer Protocol*.
- IETF RFC 1034 (1987), *Domain names – Concepts and facilities*.

- IETF RFC 1889 (1996), *RTP: A Transport Protocol for Real-Time Applications*.
- IETF RFC 2045 (1996), *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*.
- IETF RFC 2234 (1997), *Augmented BNF for Syntax Specifications: ABNF*.
- IETF RFC 2327 (1998), *SDP: Session Description Protocol*.
- IETF RFC 2543 (1999), *SIP: Session Initiation Protocol*.

NOTA – La referencia a un documento dentro de esta Recomendación no concede a éste, en cuanto documento aislado, el estatus de Recomendación.

2.2 Referencias informativas

- IETF RFC 1890 (1996), *RTP Profile for Audio and Video Conferences with Minimal Control*.
- IETF RFC 2705 (1999), *Media Gateway Control Protocol (MGCP) Version 1.0*.

3 Términos y definiciones

En esta Recomendación se definen los términos siguientes.

3.1 módem de cable: Un módem de cable es un dispositivo de terminación de dos capas que termina el extremo de abonado de la conexión J.112.

3.2 IPCablecom: Proyecto del UIT-T que incluye una arquitectura y una serie de Recomendaciones que permiten la prestación de servicios interactivos dependientes del tiempo por redes de televisión por cable.

3.3 DEBE (MUST): Se ha convenido que el término "DEBE" (MUST) o "NO DEBE" (MUST NOT) utilizado en la presente Recomendación denote un aspecto absolutamente obligatorio de la especificación.

4 Abreviaturas

En esta Recomendación se utilizan las siguientes siglas.

API	Interfaz de programación de aplicaciones (<i>application programming interface</i>)
CPE	Equipo en las instalaciones del cliente (<i>customer premise equipment</i>)
DTMF	Multifrecuencia bitono (<i>dual tone multi frequency</i>)
IP	Protocolo Internet (<i>Internet protocol</i>)
MIB	Base de información de gestión (<i>management information base</i>)
MGCI	Interfaz de controlador de pasarela de medios (<i>media gateway controller interface</i>)
MGCP	Protocolo de control de pasarela de medios (<i>media gateway control protocol</i>)
MTA	Adaptador de terminal de medios (<i>media terminal adaptor</i>)
MWD	Retardo de espera máximo (<i>maximum waiting delay</i>)
NCS	Señalización de llamada de red (<i>network call signalling</i>)
RTPC	Red telefónica pública conmutada (<i>public switched telephone network</i>)
RTP	Protocolo en tiempo real (<i>real time protocol</i>)

SDP	Protocolo de descripción de sesión (<i>session description protocol</i>)
UDP	Protocolo de datagrama de usuario (<i>user datagram protocol</i>)

5 Introducción

Esta Recomendación describe el perfil NCS de una interfaz de programación de aplicaciones (MGCI) y un protocolo correspondiente (MGCP) para el control de los clientes incorporados desde elementos de control de llamada externos. Un cliente incorporado es un elemento de red que proporciona:

- Dos o más líneas de acceso analógicas tradicionales a una red de voz sobre el protocolo Internet (VoIP).
- Una o más líneas de vídeo a una red VoIP (queda en estudio).

Los clientes incorporados no pueden ser confinados al uso residencial solamente. Por ejemplo, pueden utilizarse también para negocios. Los clientes incorporados se utilizan para el acceso del lado de línea y, como tales, tendrán equipo de lado de línea, por ejemplo líneas de acceso analógicas para los teléfonos convencionales asociados con ellas, en contraposición a las pasarelas troncales.

El MGCP supone una arquitectura de control de llamada en la que la "inteligencia" de control de llamada es exterior a las pasarelas y es manejada por elementos de control de llamada externos denominados agentes de llamada (*CA, call agents*). El MGCP supone que cada uno de estos elementos de control, o agentes de llamada, se sincronizará con todos los demás para enviar instrucciones coherentes a las pasarelas que se encuentran bajo su control. El MGCP definido en esta Recomendación no describe un mecanismo para la sincronización de los agentes de llamada, aunque futuras especificaciones del IPCablecom pueden determinar tales mecanismos.

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

Un ejemplo de punto extremo físico es una interfaz en una pasarela que termina una conexión POTS (telefonía ordinaria) analógica a un teléfono, sistema de claves, PBX, etc. Una pasarela que termina líneas POTS residenciales (a teléfonos) se denomina una *pasarela residencial*, un *cliente incorporado* o un *MTA*. Los clientes incorporados pueden opcionalmente soportar también vídeo.

Una fuente de audio en un servidor de contenido de audio constituye un ejemplo de punto extremo virtual. Para la creación de puntos extremos físicos se necesita una instalación de soporte físico, mientras que la creación de puntos extremos virtuales puede realizarse mediante soporte lógico. Sin embargo, el perfil NCS de MGCP solamente trata puntos extremos físicos.

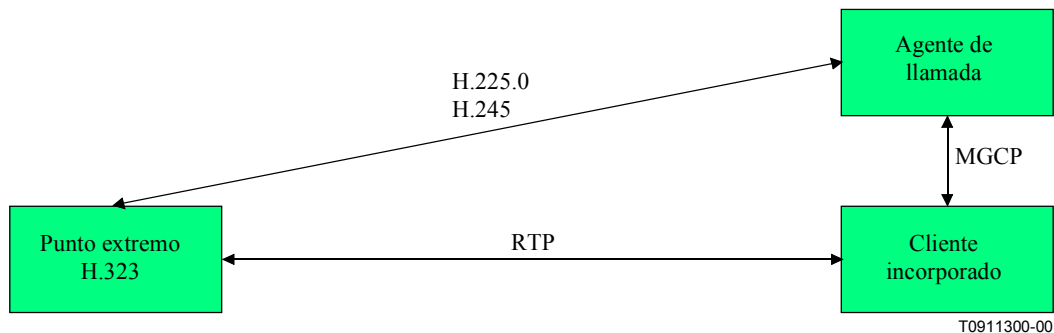
Las conexiones son conexiones punto a punto. Una conexión punto a punto es una asociación entre dos puntos extremos con el fin de transmitir datos entre los mismos. Una vez que se ha establecido esta asociación para ambos puntos extremos puede tener lugar la transferencia de datos entre ellos. La asociación se establece creando la conexión como dos mitades; una en el punto extremo de origen y la otra en el punto extremo de terminación.

Los agentes de llamada dan instrucciones a las pasarelas para que creen conexiones entre puntos extremos y detecten determinados eventos, por ejemplo descolgado, así como para que generen determinadas señales, por ejemplo, la señal de timbre. Corresponde estrictamente al agente de llamada especificar cómo y cuándo se hacen las conexiones, entre qué puntos extremos se establecen estas conexiones y los eventos y señales que son detectados y generados en los puntos extremos. La pasarela, por tanto, viene a ser un dispositivo simple, sin ningún estado de llamada, que recibe instrucciones generales del agente de llamada sin que tenga necesidad de conocer, o incluso de comprender, los conceptos de llamada, estados de llamada, características o interacciones de características. Cuando se introducen nuevos servicios, se cambian los perfiles de los abonados, etc.,

los cambios son transparentes a la pasarela. Los agentes de llamada implementan los cambios y generan la nueva combinación apropiada de instrucciones a la pasarela sobre los cambios efectuados. Siempre que la pasarela vuelve a arrancar, pasará a un estado de liberación y simplemente cursará las instrucciones del agente de llamada como las ha recibido.

5.1 Relación con las normas H.323

El MGCP se diseña como un protocolo interno dentro de un sistema distribuido de pasarelas que aparece al exterior como una sola pasarela VoIP. Este sistema está formado por un agente de llamada, que puede estar o no distribuido en varias plataformas de computador, y un conjunto de pasarelas. En una configuración H.323, este sistema distribuido de pasarelas puede conectar por un lado con o una o más líneas POTS, y por el otro lado con sistemas conformes a H.323, tal como se ilustra a continuación:



En el modelo MGCP, las pasarelas se centran en la función de traslación de la señal de audio, mientras que el agente de llamada se ocupa de las funciones de señalización y de procesamiento de la llamada. En consecuencia, el agente de llamada implementa las capas "señalización" de la norma H.323, y se presenta a sí mismo como un "controlador de acceso H.323" o como uno o más "puntos extremos H.323" a los sistemas H.323. La señalización de llamada H.225.0 y señalización de medios H.245 es por consiguiente encaminada al agente de llamada.

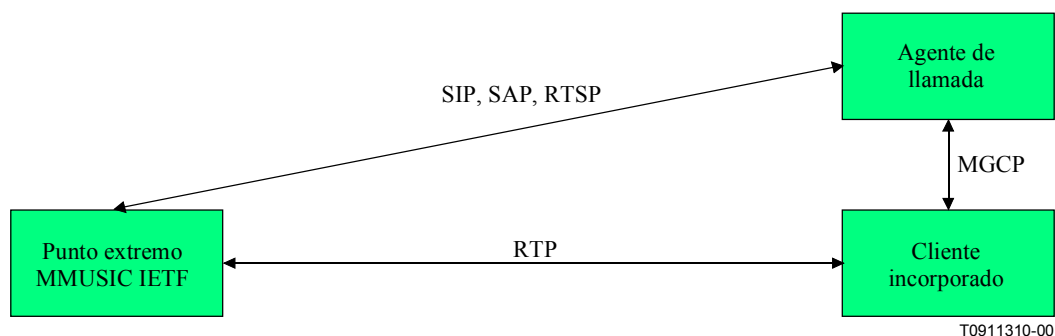
5.2 Relación con las normas IETF

Mientras que H.323 es la norma reconocida para terminales VoIP, el Grupo IETF también ha producido especificaciones para otros tipos de aplicaciones multimedios. Estas otras especificaciones incluyen:

- el protocolo de descripción de sesión (SDP, *session description protocol*), RFC 2327;
- el protocolo de anuncio de sesión (SAP, *session announcement protocol*); RFC 2974: trabajos en curso;
- el protocolo de iniciación de sesión (SIP, *session initiation protocol*), RFC 2543;
- el protocolo de transmisión de flujo continuo en tiempo real (RTSP, *real-time streaming protocol*), RFC 2326.

Las tres últimas especificaciones son, de hecho, normas de señalización alternativas que permiten la transmisión de una descripción de sesión a una parte interesada. El SAP es utilizado por los gestores de las sesiones de multidifusión para distribuir una descripción de sesión de multidifusión a un grupo extenso de recipientes. El SIP se utiliza para invitar a un usuario en particular a que forme parte de una sesión de unidifusión punto a punto. El RTSP se utiliza para conectar un servidor que proporciona datos en tiempo real. En los tres casos, la descripción de sesión se hace de acuerdo con SDP; en el caso de audio, la transmisión se realiza a través del protocolo de transporte en tiempo real (RTP, *real-time transport protocol* y RTCP).

Los sistemas distribuidos de pasarelas y el MGCP posibilitarán la comunicación vocal por la RTPC y el acceso de los clientes incorporados al establecimiento de sesiones mediante la utilización de los SAP, SIP o RTSP definidos por el Grupo de Trabajo MMUSIC de IETF. El agente de llamada proporciona la conversión de señalización, tal como se ilustra a continuación:



La norma SDP tiene un estatus pivotante en esta arquitectura. En la descripción que sigue veremos también que la utilizamos para cursar descripciones de sesión en MGCP.

6 Interfaz de control de pasarela de medios (MGCI)

Las funciones MGCI realizan el control de la conexión, el control del punto extremo, la auditoría y el informe de situación. Cada una de ellas utiliza el mismo modelo de sistema y los mismos convenios de denominación.

6.1 Modelo y convenios de denominación

El MGCP supone un modelo de conexión en el cual los elementos constructivos básicos son puntos extremos y conexiones. Las conexiones se agrupan en llamadas. Una o más conexiones pueden pertenecer a una llamada. Las conexiones y las llamadas se pueden establecer por iniciativa de uno o varios agentes de llamada.

6.1.1 Nombres de los puntos extremos

Los nombres de puntos extremos, a.k.a. identificadores de puntos extremos, tienen dos componentes, definidos ambos aquí para que sean insensibles a la posición mayúsculas/minúsculas:

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

Los nombres de punto extremo serán de la forma

```
local-endpoint-name@domain-name
```

donde `domain-name` es el `domain-name` (nombre de dominio) absoluto definido en RFC 1034 e incluye una porción anfitriona, por lo que un ejemplo de `domain-name` podría ser:

```
MyEmbeddedClient.cablelabs.com
```

Asimismo, `domain-name` puede ser una dirección del protocolo Internet versión 4 (IPv4) en forma decimal con puntos representada como una cadena de texto y encerrada por un corchete izquierdo y un corchete derecho ("[" y "]") como en "[128.96.41.1]" — consúltese RFC 821 para más detalles. Sin embargo, se desaconseja por lo general la utilización de direcciones IP.

Los clientes incorporados pueden tener uno o más puntos extremos (por ejemplo, uno para cada conector RJ11 de teléfonos negros) asociados con ellos, y cada uno de los puntos extremos se identifica como por un nombre de punto extremo local independiente. Del mismo modo que el

nombre de dominio, el nombre de punto extremo local es insensible a la posición mayúsculas/minúsculas. Asociado con el nombre de punto extremo local hay un tipo de punto extremo, que define el tipo de punto extremo, tal como teléfono analógico o videoteléfono. El tipo de punto extremo puede obtenerse a partir del nombre de punto extremo local. El nombre de punto extremo local es un nombre jerárquico, en el cual el componente menos específico del nombre es el término situado más a la izquierda, y el componente más específico el término más a la derecha. De modo más formal, el nombre de punto extremo local debe cumplir las siguientes reglas de denominación:

- Los términos individuales del nombre de punto extremo local deben separarse por una barra de fracción ("/", carácter 2F hex ASCII).
- Los términos individuales son cadenas de caracteres ASCII compuestas de letras, dígitos u otros caracteres imprimibles, salvo los caracteres utilizados como delimitadores en los nombres de punto extremo ("/", "@"), caracteres utilizados como comodines ("*", "\$") y caracteres de espacio en blanco.
- El comodín se representa mediante un asterisco ("*") o un signo de dólar ("\$") para los términos del trayecto de denominación que han de ser sustituidos por un comodín. De este modo, si el nombre de punto extremo local completo se presenta como:

term1/term2/term3

y uno de los términos de este nombre de este punto extremo local está representado por un comodín, y el nombre de punto extremo local aparece como:

term1/term2/* si term3 está sustituido por un comodín.

term1/*/* si term2 y term3 están sustituidos por comodines.

En cada uno de los ejemplos, puede haber aparecido el signo de dólar en vez del signo asterisco.

- El uso de comodines solamente está permitido empezando por la derecha, por lo que si un término está tratado con comodín, todos los términos a la derecha de este término deben también tratarse con comodines.
- En los casos en que se utilizan mezclados como comodines el signo de dólar y el asterisco, los signos de dólar sólo están autorizados a partir de la derecha, por lo que si un término tiene un signo de dólar como comodín, todos los demás términos a la derecha del mismo deben contener también como comodines el signo de dólar.
- Un término representado por un asterisco debe interpretarse como que: "utiliza *todos* ("*all*") los valores de este término conocidos dentro del ámbito del cliente incorporado en cuestión".
- Un término representado por un signo de dólar debe interpretarse como que: "utiliza *uno cualquiera* ("*any one*") de los valores de este término conocido dentro del ámbito del cliente incorporado en cuestión".
- Cada tipo de punto extremo puede especificar detalles adicionales en las reglas de denominación para este tipo de punto extremo, pero tales reglas no deben entrar en conflicto con las anteriormente señaladas.

Debe señalarse que diferentes tipos de punto extremo o incluso diferentes subtérminos, por ejemplo, "líneas", dentro del mismo tipo de punto extremo darán como resultado diferentes nombres de punto extremo local. Consecuentemente, cada "línea" será tratada como un punto extremo separado.

6.1.1.1 Nombres de puntos extremos de clientes incorporados

Los puntos extremos en clientes incorporados DEBEN soportar los convenios de denominación adicionales especificados en esta cláusula.

Los clientes incorporados PUEDEN soportar uno o más tipos de punto extremo, incluidos los siguientes:

- **Analogue Telephone (Teléfono analógico)** – El teléfono analógico se representa como una línea de acceso analógica (aaln, *analogue access line*). Ésta es básicamente el equivalente de la línea telefónica analógica conocida de la RTPC.
- **Video (Vídeo)** – Los detalles del tipo de dispositivo vídeo quedan en estudio.
- **Basic Access ISDN (Acceso básico RDSI)** – Los detalles del tipo de dispositivo RDSI quedan en estudio.

6.1.1.1 Puntos extremos de línea de acceso analógica

Además de adherirse a los convenios de denominación especificados anteriormente, los nombres de punto extremo local para puntos extremos del tipo de "línea de acceso analógica" (aaln) de clientes incorporados deben adherirse a los siguientes:

- Los nombres de punto extremo local contienen al menos un término, y como máximo dos términos.
- term1 DEBE ser el término "aaln" o un carácter comodín. Debe señalarse que la utilización de un carácter comodín para term1 puede hacer referencia a cualquiera o a todos los tipos de punto extremo de los clientes incorporados con independencia de sus tipos. Se espera, por lo general, que esta característica se aplique con fines administrativos, por ejemplo, en la auditoría o el reorganización.
- term2 DEBE ser un número comprendido entre uno y el número de líneas de acceso analógicas soportado por el cliente incorporado en cuestión. El número así definido identifica una línea de acceso analógica específica en el cliente incorporado.
- Si un nombre de punto extremo local está formado por uno solo término, éste será term1.
- Si term1 *no es* un carácter comodín, se adopta entonces el carácter comodín signo de dólar (referente a "uno cualquiera") para term2, es decir, "aaln" es equivalente a "aaln/\$".
- Si term1 *es* un carácter comodín, se adopta entonces el carácter comodín asterisco (referente a "todos") para term2, es decir, "*" y "\$"son equivalentes, respectivamente, a "*/*" y "\$/*".

Por ello, los siguientes podrían ser ejemplos de nombres de punto extremo local de línea de acceso analógica:

- aaln/1 La primera línea de acceso analógica en el cliente incorporado en cuestión.
- aaln/2 La segunda línea de acceso analógica en el cliente incorporado en cuestión.
- aaln/\$ Cualquiera línea de acceso analógica en el cliente incorporado en cuestión.
- aaln/* Todas las líneas de acceso analógicas en el cliente incorporado en cuestión.
- * Todos los puntos extremos (independientemente del tipo de punto extremo) en el cliente incorporado en cuestión.

El proceso de aprovisionamiento/(auto)configuración se ocupa de la obtención y suministro de información acerca del número de puntos extremos que tiene un cliente incorporado, así como del tipo de punto extremo de cada uno de los puntos extremos. Aunque son lógicamente diferentes, debe señalarse que el *tipo de punto extremo* puede obtenerse de la porción local del nombre de punto extremo.

6.1.2 Nombres de las llamadas

Las llamadas se identifican mediante identificadores exclusivos, con independencia de los agentes o plataformas subyacentes. Los identificadores de llamada son cadenas hexadecimales creadas por el agente de llamada. DEBEN soportarse identificadores de llamada de una longitud máxima de 32 caracteres.

Los identificadores de llamada DEBEN, como mínimo, ser exclusivos dentro del conjunto de agentes de llamada que controlan las mismas pasarelas. Sin embargo, la coordinación de estos identificadores de llamada entre los agentes de llamada cae fuera del alcance de esta Recomendación. Cuando un agente de llamada establece varias conexiones que pertenecen a la misma llamada, bien en la misma pasarela o en pasarelas diferentes, todas estas conexiones se asignan a la misma llamada por medio del identificador de llamada. Este identificador puede utilizarse entonces en los procedimientos de gestión o contabilidad, los cuales caen fuera del ámbito del MGCP.

6.1.3 Nombres de las conexiones

Los identificadores de conexión son creados por la pasarela cuando se solicita crear una conexión. Ellos identifican la conexión dentro del contexto de un punto extremo. Los identificadores de conexión son tratados en el MGCP como cadenas hexadecimales. La pasarela DEBE garantizar que transcurre un periodo de espera adecuado, al menos tres minutos, entre el final de una conexión que ha utilizado un identificador y el empleo de este identificador en una nueva conexión para el mismo punto extremo. DEBEN soportarse nombres de conexión con una longitud máxima de 32 caracteres.

6.1.4 Nombres de agentes de llamada y otras entidades

El protocolo de control de pasarela de medios ha sido diseñado para mejorar la fiabilidad de la red de modo que permita la implementación de agentes de llamada redundantes. Esto significa que no existe una vinculación fija entre entidades y plataformas de soporte físico o interfaces de red.

Los nombres de agentes de llamada se componen de dos partes, de forma similar a los nombres de punto extremo. La parte local del nombre no exhibe ninguna estructura interna. Un ejemplo de nombre de agente de llamada es:

```
cal@ca.whatever.net
```

La fiabilidad es aportada por la adopción de las siguientes precauciones:

- Entidades tales como los clientes incorporados o los agentes de llamada son identificadas por su nombre de dominio, y no por sus direcciones de red. A un nombre de dominio se pueden asociar varias direcciones. Si una instrucción no puede ser reenviada a una de las direcciones de red, las implementaciones DEBEN reintentar la transmisión utilizando otra dirección.
- Las entidades se pueden trasladar a otra plataforma. La asociación entre un nombre lógico (nombre de dominio) y la plataforma real se mantiene en el servicio de nombres de dominio DNS, *domain name service*). Los agentes de llamada y las pasarelas DEBEN vigilar el tiempo de vida del registro leído en el DNS. DEBEN pedir al DNS que renueve la información si ha expirado el tiempo de vida.

Además del direccionamiento proporcionado por la utilización de nombres de dominio y el DNS, el concepto de "entidad notificada" es básico para la fiabilidad y el cambio-por-fallo en MGCP. La "entidad notificada" de un punto extremo es el agente de llamada que está actualmente controlando ese punto extremo. En cualquier instante, un punto extremo tiene una, y sólo una, "entidad notificada" asociada con él, y cuando el punto extremo necesita enviar una instrucción al agente de llamada, DEBE enviar la instrucción a la "entidad notificada" actual a cuyo(s) punto(s) extremo(s) pertenece la instrucción. Tras el arranque, la "entidad notificada" DEBE fijarse a un valor aprovisionado. La mayor parte de las instrucciones enviadas por el agente de llamada incluyen la capacidad de nombrar explícitamente la "entidad notificada" mediante la utilización de un parámetro "NotifiedEntity". La "entidad notificada" DEBE permanecer la misma hasta que se reciba un nuevo parámetro "NotifiedEntity" o hasta que se reinicie el punto extremo. Si la "entidad notificada" para

un punto extremo está vacía o no ha sido fijada explícitamente¹, la "entidad notificada" utilizará como valor por defecto la dirección de fuente de la última instrucción de tratamiento de conexión o la petición de notificación recibida para el punto extremo. La auditoría de este modo no cambiará la "entidad notificada".

La cláusula 6.4 contiene una descripción más detallada de la fiabilidad y la superación de fallos.

6.1.5 Mapas de dígitos

El agente de llamada puede pedir a la pasarela que recoja los dígitos marcados por el usuario. Esta facilidad es utilizada por las líneas de acceso analógico con pasarelas residenciales para recopilar los números que marca un usuario; esto puede utilizarse para recopilar códigos de acceso, números de tarjetas de crédito y otros números requeridos por los servicios de control de llamadas. Los puntos extremos DEBEN soportar mapas de dígitos como lo definidos en esta cláusula.

Un procedimiento alternativo incluye que la pasarela notifique al agente de llamada los dígitos marcados lo más pronto posible desde que son marcados, a.k.a. solapándose con la emisión. Sin embargo, este procedimiento genera un número elevado de interacciones. Es preferible acumular los números marcados en una memoria intermedia y transmitirlos después en un solo mensaje.

El problema que aparece con este método de acumulación consiste, sin embargo, en la dificultad que tiene la pasarela de predecir cuántos números necesita acumular antes de la transmisión. Por ejemplo, al utilizar el teléfono de nuestra mesa de trabajo podemos marcar los siguientes números:

0	Operador local
00	Operador de larga distancia
xxxx	Número de la extensión local
8xxxxxxx	Número local
#xxxxxxx	Acceso a número local en otros lugares corporativos
*xx	Servicios estrella
91xxxxxxxxxxx	Número de larga distancia
9011 + hasta 15 dígitos	Número internacional

Este problema se resuelve cargando la pasarela con un mapa de dígitos que corresponda al plan de marcación para la zona en la que reside la pasarela. Por ello el mapa de dígitos real utilizado puede diferir de unas regiones a otras. Este mapa de dígitos se expresa utilizando una sintaxis derivada de la instrucción del sistema UNIX, *egrep*. Por ejemplo, el plan de marcación descrito anteriormente da como resultado el siguiente mapa de dígitos:

```
(0T| 00T| [1-7]xxx| 8xxxxxxx| #xxxxxxx| *xx| 91xxxxxxxxxxx| 9011x.T)
```

La sintaxis formal del mapa de dígitos se describe mediante la siguiente notación BNF:

```
Digit      ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
Timer      ::= "T" | "t" -- matches the detection of a timer
Letter     ::= Digit | Timer | "#" | "*" | "A" | "a" | "B" | "b" | "C" | "c" | "D" | "d"
Range      ::= "X" | "x" -- matches any digit
           | "[" Letters "]" -- matches any of the specified letters
Letters    ::= Subrange | Subrange Letters
Subrange   ::= Letter -- matches the specified letter
           | Digit "-" Digit -- matches any digit between first and last
Position   ::= Letter | Range
```

¹ Esto puede ocurrir como resultado de especificar un parámetro NotifiedEntity vacío.

```

StringElement ::= Position          -- matches an occurrence of the position
                | Position "."      -- matches an arbitrary number of occurrences
                                     -- of the position, including 0
String         ::= StringElement | StringElement String
StringList    ::= String | String "|" StringList
DigitMap      ::= String | "(" StringList ")"

```

Un digit map, de acuerdo con esta sintaxis, estará definido, bien por una "cadena" (insensible a la posición mayúsculas/minúsculas) o bien por una "lista de cadenas". Con independencia de la sintaxis anterior, actualmente sólo está permitido un temporizador si aparece en la última posición de una cadena². Cada cadena de la lista es un esquema de numeración alternativo. Una pasarela que detecta dígitos, letras o temporizadores:

- 1) Añadirá el código de parámetro evento al dígito, letra o temporizador, como testigo al final de la variable de estado interna "cadena de marcación vigente".
- 2) Aplicará la "cadena de marcación vigente" a la tabla del mapa de dígitos, intentando una concordancia, en orden lexicológico, para cada expresión regular en el mapa de dígitos.
- 3) Si el resultado está infracalificado (concuera parcialmente al menos en una entrada en el mapa de dígitos), no hará nada más.

Si el resultado concuerda, o esta sobrecalificado (es decir, ningún dígito adicional podría posiblemente producir una concordancia), enviará la lista de dígitos al agente de llamada³ y liberará la "cadena de marcación vigente".

El temporizador T es un temporizador de entrada de dígitos que puede utilizarse de dos modos:

- Cuando se utiliza el temporizador T con un mapa de dígitos⁴, el temporizador no arrancará hasta que entre el primer dígito, y se rearrancará después de la entrada de cada nuevo dígito hasta que se produzca una concordancia o discordancia del mapa de dígitos. En este caso, el temporizador T funciona como un temporizador entre dígitos.
- Cuando se utiliza el temporizador T sin un mapa de dígitos, se arrancará el temporizador inmediatamente y se cancelará sencillamente (pero no se rearrancará) tan pronto como entra un dígito. En este caso el temporizador T puede utilizarse como temporizador entre dígitos si se aplica el solapamiento de la emisión.

Cuando se utiliza con un mapa de dígitos, el temporizador T toma uno de los dos valores, T_{par} o T_{crit} . Cuando se necesita al menos un dígito más para que la cadena de dígitos concuerde con alguno de los esquemas del mapa de dígitos, el temporizador T toma el valor T_{par} , correspondiente a la temporización de marcación parcial. Si un temporizador es todo lo que se requiere para producir una concordancia, el temporizador T toma el valor T_{crit} que corresponde a la temporización crítica. Si se utiliza sin un mapa de dígitos, el temporizador T toma el valor T_{crit} . El valor por defecto de T_{par} es 16 segundos y el valor por defecto de T_{crit} es 4 segundos. El proceso de aprovisionamiento puede alterar ambos valores.

El agente de llamada puede proporcionar mapas de dígitos a la pasarela, siempre que el agente de llamada de instrucciones a ésta para que preste atención a los dígitos. De nuevo debe resaltarse que los detalles del mapa de dígitos utilizado dependerán de la zona en que resida la pasarela y de acuerdo con la cual el mapa de dígitos es programable. Los mapas de dígitos, cuando son proporcionados por el agente de llamada, DEBEN ser los definidos en esta cláusula.

² Por ejemplo, "123T" y "123[1-2T5]" satisfacen esta regla, pero "12T3" no.

³ La lista de dígitos puede incluir también otros eventos – véase 6.4.3.1.

⁴ Técnicamente hablando, con la acción "acumular de conformidad con el mapa de dígitos".

6.1.6 Eventos y señales

Eventos y señales es un concepto central en el MGCP. Un agente de llamada puede solicitar ser informado acerca de determinados eventos que ocurran en un punto extremo, por ejemplo, eventos señal de descolgado. Un agente de llamada puede también pedir que se apliquen determinadas señales a un punto extremo, por ejemplo, el tono de invitación a marcar.

Los eventos y señales están agrupados en lotes dentro de los cuales comparten el mismo espacio de nombres, a los cuales nos referiremos en lo sucesivo como nombres de evento. Un lote es un conjunto de eventos y señales soportados por un tipo de punto extremo particular. Por ejemplo, un lote puede soportar un determinado grupo de eventos y señales para líneas de acceso analógicas, y otro lote puede soportar otro grupo de eventos y señales para líneas de vídeo. Puede haber uno o más lotes para un tipo de punto extremo dado, y cada tipo de punto extremo tiene un lote por defecto con el que está asociado.

Los nombres de evento están formados por un nombre de lote y un código de evento y, puesto que cada lote define un "namespace" (espacio de nombres) separado, pueden utilizarse los mismos códigos de evento en distintos lotes. Los nombres de lote y los códigos de evento son cadenas de letras, dígitos y guiones insensibles a la posición mayúsculas/minúsculas, con la restricción de que los guiones NO DEBEN ser el primer o el último carácter de un nombre. Puede ser preciso parametrizar algunos códigos de evento con datos adicionales, lo que se efectúa añadiendo los parámetros entre un conjunto de paréntesis. El nombre del lote está separado del código de evento por una barra de fracción ("/"). El nombre del lote puede ser excluido del nombre del evento, en cuyo caso se supone el nombre de lote por defecto para el tipo de punto extremo en cuestión. Por ejemplo, para una línea de acceso analógico en la que el ejemplo de lote de línea (nombre de lote "X") es el lote por defecto, se consideran iguales los dos siguientes nombres de evento:

- X/dl tono de invitación a marcar en el ejemplo de lote de línea para una línea de acceso analógica.
- dl tono de invitación a marcar en el lote (por defecto) de línea del ejemplo para una línea de acceso analógica.

En el anexo A se define un conjunto inicial de lotes. Los códigos de evento y nombres de lote adicionales pueden ser definidos y/o registrados por IPCablecom. Ningún cambio en los lotes definido en esta Recomendación DEBE acarrear un cambio del nombre del lote, o un cambio en el número de versión del perfil NCS, o ambos cambios a la vez.

Cada lote DEBE tener una definición del lote, la cual DEBE definir el nombre del lote y cada evento perteneciente al lote. La definición de un evento DEBE incluir el nombre preciso del evento, es decir, el código de evento, una definición del evento en lenguaje claro y, si procede, la definición precisa de las señales correspondientes, por ejemplo las frecuencias exactas de las señales de audio, como el tono de invitación a marcar o los tonos de multifrecuencia bitono (DTMF). Los eventos deben además especificar si son persistentes (por ejemplo, el descolgado, véase 6.3.1) y si contienen estados de eventos auditables (por ejemplo, descolgado, véase 6.3.8.1). Las señales DEBEN también tener sus tipos definidos (On/Off, Time-Out o Brief), y las señales de temporización DEBEN tener un valor de temporización por defecto definido – véase 6.3.1.

Además de lotes IPCablecom, los implementadores PUEDEN aumentar su experiencia definiendo lotes experimentales. El nombre de lote de los lotes experimentales DEBE comenzar con los dos caracteres "x-" o "X-"; IPCablecom NO DEBE registrar nombres de lote que comiencen con estos dos caracteres. Un cliente incorporado que recibe un instrucción referente a un lote no soportado DEBE devolver un error (código de error 518 – lote no soportado).

Los nombres de lote y los códigos de evento soportan cada uno una notación con comodín. El carácter comodín "*" (asterisco) se puede utilizar para hacer referencia a todos los lotes soportados por el punto extremo en cuestión, y el código de evento "all" ("todos") para referirse a todos los eventos en el lote en cuestión. Por ejemplo:

- X/all hace referencia a todos los eventos en el ejemplo de lote de línea para una línea de acceso analógica.
- */all para una línea de acceso analógica; hace referencia a todos los lotes y todos los eventos en estos lotes soportados por el punto extremo en cuestión.

Consecuentemente, el nombre de lote "*" NO DEBE ser asignado a un lote, y el código de evento "all"("todos") NO DEBE utilizarse en ningún lote.

Los eventos y señales son detectados y generados por defecto en los puntos extremos, pero algunos eventos y señales pueden ser detectados y generados en conexiones además de, o en lugar de, en un punto extremo. Por ejemplo, los puntos extremos pueden recibir la petición de proporcionar 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 determinar explícitamente que el evento/señal puede ser detectado o generado en una conexión.

Cuando se aplique una señal en una conexión, el nombre de la conexión deberá añadirse al nombre del evento utilizando un signo "at" (@) como delimitador, tal como en:

```
X/rt@0A3F58
```

El carácter comodín "*" (asterisco) se puede utilizar para indicar "todas las conexiones" en el(los) punto(s) extremo(s) afectado(s). Cuando se aplica este convenio, la pasarela DEBE generar o detectar el evento en todas las conexiones que están establecidas con el(los) punto(s) extremo(s). Un ejemplo de este convenio es:

```
X/rt@*
```

El carácter comodín "\$" (signo de dólar) puede utilizarse para indicar "la conexión actual". Este convenio NO DEBE utilizarse a menos que la petición de notificación de evento esté encapsulada dentro de una instrucción crear conexión (CreateConnection) o modificar conexión (ModifyConnection). Cuando se aplica el acuerdo, la pasarela DEBE generar o detectar el evento en la conexión que se está actualmente creando o modificando. Un ejemplo de este convenio es:

```
X/rt@$
```

El id de conexión, o un comodín de sustitución, puede utilizarse junto con los convenios "todos los lotes" y "todos los eventos". Por ejemplo, la notación:

```
*/all@*
```

puede utilizarse para designar todos los eventos en todas las conexiones para el(los) punto(s) extremo(s) afectado(s).

6.2 Utilización del SDP

El agente de llamada utiliza el MGCP para dotar a las pasarelas de la descripción de parámetros de conexión tales como direcciones IP, puerto UDP y perfiles RTP. Salvo cuando están indicadas o supuestas en esta Recomendación de modo distinto, las descripciones SDP DEBEN cumplir los convenios delineados en el protocolo de descripción de sesión (SDP), que es en la actualidad una norma RFC 2327 propuesta por el IETF.

El SDP tiene en cuenta la descripción de las conferencias multimedios. El perfil NCS sólo soportará el establecimiento de conexiones audio y vídeo utilizando los tipos de medios "audio" y "vídeo". Actualmente sólo se han especificado las conexiones "audio".

6.3 Funciones de control de pasarela

Esta cláusula describe las instrucciones del MGCP en la forma de una API propia de llamada de procedimiento distante (RPC, *remote procedure call*), a la que denominaremos interfaz de controlador de pasarela de medios (MGCI). Para cada instrucción MGCP se define una función MGCI, donde la función MGCI toma y devuelve los mismos parámetros que la instrucción MGCP correspondiente. Las funciones que se muestran en esta cláusula proporcionan una descripción de alto nivel del funcionamiento de MGCP y describen un ejemplo de una API propia de RPC que PUEDE utilizarse para una implementación de MGCP. Aunque la API MGCI es exclusivamente un ejemplo de API, el comportamiento semántico definido por MGCI forma parte integrante de la Recomendación, y todas las implementaciones DEBEN adecuarse a la semántica especificada para MGCI. Los mensajes MGCP reales intercambiados, incluidos los formatos y las codificaciones de mensaje utilizados, se definen en la sección de protocolo (cláusula 7). Los clientes incorporados DEBEN implementar aquellos en la forma exacta que se especifica.

El servicio MGCI está formado por instrucciones de gestión de conexión y de gestión de punto extremo. A continuación se da una visión general de las instrucciones:

- El agente de llamada puede emitir una instrucción NotificationRequest (petición de notificación) a una pasarela, instruyendo a ésta para que espere eventos específicos tales como acciones de gancho conmutador o tonos DTMF en un punto extremo especificado.
- La pasarela utilizará entonces la instrucción Notify (notificar) para informar al agente de llamada acerca de cuándo ocurren los eventos solicitados en el punto extremo especificado.
- El agente de llamada puede utilizar la instrucción CreateConnection (crear conexión) para crear una conexión que termine en un punto extremo dentro de la pasarela.
- El agente de llamada puede utilizar la instrucción ModifyConnection (modificar conexión) para cambiar los parámetros asociados con una conexión anteriormente establecida.
- El agente de llamada puede utilizar la instrucción DeleteConnection (suprimir conexión) para eliminar una conexión existente. En algunas circunstancias, la instrucción DeleteConnection puede ser utilizada también por una pasarela para indicar que una conexión no puede mantenerse por más tiempo.
- El agente de llamada puede utilizar las instrucciones AuditEndpoint (auditar punto extremo) y AuditConnection (auditar conexión) para auditar la situación de un "punto extremo" y cualesquiera conexiones asociadas al mismo. Es deseable por lo general una gestión de red que abarque más capacidades que las proporcionadas por estas instrucciones, por ejemplo, la información sobre la situación del cliente incorporado. Se espera que tales capacidades estén sustentadas por la utilización del protocolo de gestión de red simple (SNMP, *simple network management protocol*) y la definición de una base MIB, lo que cae fuera del alcance de esta Recomendación.
- La pasarela puede utilizar la instrucción RestartInProgress (rearranque en curso) para notificar al agente de llamada que el punto extremo, o un grupo de puntos extremos gestionado por la pasarela, está retirado del servicio o se está pasando a la situación de reserva.

Estos servicios permiten a un controlador (normalmente un agente de llamada) dar instrucciones a una pasarela sobre la creación de conexiones que terminan en un punto extremo asociado a la pasarela, y ser informado acerca de los eventos que ocurren en dicho punto extremo. Generalmente, un punto extremo está limitado a una línea de acceso analógica específica dentro de un cliente incorporado.

Las conexiones se agrupan en "llamadas" ("calls"). Varias conexiones, que pueden pertenecer o no a la misma llamada, pueden terminar en el mismo punto extremo. Cada conexión es calificada por un parámetro "mode" ("modo"), que puede ser fijado a "send only" (sendonly, emisión solamente), "receive only"(recvonly, recepción solamente), "send/receive" (sendrecv, emisión/recepción),

"conference" (confrnce, conferencia), "inactive" (inactive, inactivo), "replicate" (replcate, repetir), "network loopback" (netwloop, bucle de red) o "network continuity test" (netwtest, prueba de continuidad de red). El parámetro "mode" determina si los paquetes de medios pueden ser enviados a la conexión y/o recibidos por la misma; sin embargo, la RTPC no se ve afectada.

Las señales de audio recibidas del punto extremo se enviarán por cualquier conexión para el punto extremo cuyo modo sea "send only", "send/receive", "conference", o "replicate".

El manejo de las señales audio recibidas por estas conexiones está determinado también por los parámetros de modo:

- Las señales audio recibidas en paquetes de datos a través de conexiones en modo "inactive" o "replicate" son descartadas.
- Las señales audio recibidas en paquetes de datos a través de conexiones en modo "receive only", "conference" o "send/receive" son mezcladas juntas y enviadas a continuación al punto extremo.
- Las señales audio que se originan en el punto extremo son transmitidas por todas las conexiones cuyo modo sea "send only", "conference" o "send/receive".
- Además de enviarse al punto extremo, las señales audio recibidas en paquetes de datos a través de conexiones en modo "conference" son repetidas a todas las demás conexiones para los puntos extremos con modo "conference". Los detalles de este reenvío, por ejemplo, el mezclador o traslator RTP, etc., cae fuera del alcance de esta Recomendación.
- Las señales audio enviadas al punto extremo, o procedentes del mismo, son mezcladas y transmitidas por todas las conexiones que tienen el modo "replicate". Esta operación DEBERÍA incluir señales audio generadas por señales.
- Las señales audio recibidas en paquetes de datos a través de conexiones en modo "network loopback" (bucle de red) o "network continuity test" (prueba de continuidad de red) serán devueltas por la conexión como se describe más adelante.

Si el modo es fijado a "network loopback," las señales audio recibidas de la conexión serán devueltas por eco en la misma conexión. El modo "network loopback" DEBERÍA operar simplemente como un reflector de paquetes RTP.

El modo "network continuity test" se utiliza para comprobar la continuidad a través de la red IP. Una señal específica del tipo de punto extremo es enviada a los puntos extremos por la red IP, y al punto extremo le corresponde a continuación devolver por eco la señal a través de la red IP después de que atravesase el equipo interno de la pasarela para verificar que la operación es adecuada. La señal DEBE atravesar la decodificación y recodificación internas antes de ser devuelta hacia atrás. Para las líneas de acceso analógicas, la señal será una señal de audio y NO DEBE ser transferida a un teléfono conectado a la línea de acceso analógica, independientemente del estado actual del gancho conmutador del microteléfono, es decir colgado o descolgado.

Las conexiones nuevas o existentes para el punto extremo NO DEBEN verse afectadas por conexiones puestas en el modo "network loopback" o el modo "network continuity test". Sin embargo, las constricciones de los recursos locales pueden limitar el número de nuevas conexiones que pueden efectuarse.

El modo "replicate" DEBE soportar como mínimo la repetición del tren procedente del punto extremo y de otra conexión, con independencia del método de codificación utilizado para esta otra conexión. No obstante, la conexión "replicate" sólo es REQUERIDA para soportar un tren de medios resultante de la codificación⁵ G.711. El soporte del modo "conference" es opcional. En el apéndice IV se pueden ver ilustraciones de las interacciones de modo.

⁵ La conexión "replicate" puede utilizarse, por ejemplo, para soportar la "verificación de línea ocupada" con una repercusión mínima en cuanto a recursos sobre el cliente incorporado.

6.3.1 NotificationRequest (petición de notificación)

La instrucción NotificationRequest se utiliza para pedir a la pasarela que envíe una notificación después de la ocurrencia de eventos concretos en un punto extremo. Por ejemplo, se puede necesitar una notificación cuando se detectan en el punto extremo los tonos asociados con una comunicación fax. La entidad que recibe esta notificación, normalmente el agente de llamada, puede entonces decidir qué tipo de codificación diferente debe utilizarse en las conexiones vinculadas a este punto extremo y dar a la pasarela las instrucciones consiguientes⁶.

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

EndpointId es el identificador para el(los) punto(s) extremo(s) en la pasarela en la que se ejecuta la NotificationRequest. El endpointId DEBE ajustarse a las reglas sobre nombres de puntos extremos especificadas en 6.1.1. NO DEBE utilizarse el comodín "cualquiera de" ("any of").

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

RequestIdentifier se utiliza para relacionar esta petición con la notificación que puede desencadenar. Será repetida en la instrucción Notify correspondiente.

SignalRequests es un parámetro que contiene el conjunto de señales que se pide que aplique la pasarela. Salvo que se especifique lo contrario, las señales se aplicarán al punto extremo, aunque algunas señales se pueden aplicar a una conexión. Son ejemplos de señales los siguientes⁷:

- tono de llamada;
- tono de ocupado;
- tono de llamada en espera;
- tono de aviso indicativo de descolgado;
- tonos de llama de retorno en una conexión.

Según su comportamiento, las señales se dividen en diferentes tipos:

- **On/off (OO) (Activado/desactivado)** – Una vez aplicadas, estas señales duran hasta que son desactivadas. Esto solamente puede ocurrir como resultado de un nuevo parámetro SignalRequests donde la señal es desactivada (véase antes). Las señales de tipo OO tienen la misma potencia, por lo que múltiples peticiones para activar (o desactivar) una señal dada OO son perfectamente válidas y NO DEBEN producir ningún error. Una señal On/Off podría ser un indicador de espera de un mensaje visual (VMWI, *visual message waiting indicator*). Una vez activado NO DEBE desactivarse hasta no recibir instrucciones explícitas al respecto del agente de llamada o hasta que reanuda el punto extremo.
- **Time-out (TO) (Temporización)** – Una vez aplicadas, estas señales duran, bien hasta que son canceladas (por la ocurrencia de un evento o por no estar incluidas en una lista [posiblemente vacía] de señales), o bien hasta que ha transcurrido un periodo de tiempo específico de la señal. Una señal que expira generará un evento "operación completada"

⁶ La nueva instrucción sería una instrucción ModifyConnection (modificar conexión).

⁷ Referirse a 0 para una lista completa de señales.

(véase en el anexo A la definición más detallada de este evento). Una señal TO podría ser la temporización del tono de llamada de retorno que expira después de 180 segundos. Si ocurre un evento antes de los 180 segundos, la señal será, por defecto, detenida⁸. Si la señal no es detenida, la señal expirará, se detendrá y generará un evento "operación completada", acerca del cual el agente de llamada puede o no haber solicitado ser informado. Si el agente de llamada ha pedido que el evento "operación completada" le sea notificado, el evento "operación completada" enviado al agente de llamada incluirá el(los) nombre(s) de la(s) señal(es) que han expirado⁹. La(s) señal(es) generadas en una conexión incluirán el nombre de esta conexión. Las señales de temporización tiene un valor de temporización por defecto definido para ellas, el cual puede ser alterado por el proceso de aprovisionamiento. Asimismo, el periodo de temporización puede ser proporcionado a la señal como un parámetro. Un valor cero indica que el periodo de temporización es infinito. Una señal TO que falla después de haber comenzado pero antes de haber generado un evento "operación completada", generará un evento "operación fracasada" que incluirá el(los) nombre(s) de la(s) señal(es) que han expirado⁹.

- **Brief (BR) (Breve)**— La duración de estas señales es tan corta que se detienen por sí mismas. Si ocurre un evento de detención de señal, o se aplica un nuevo parámetro `SignalRequests`, una señal BR activa actualmente no se detendrá. Sin embargo, cualquier señal BR pendiente que no ha sido todavía aplicada será cancelada. Un tono breve podría ser un dígito DMTF (multifrecuencia bitono). Si se está ejecutando el dígito DMTF "1", y ocurre un evento de detención de señal, el "1" finalizaría su "ejecución".

Las señales se aplican, por defecto, a los puntos extremos. Si una señal aplicada a un punto extremo da lugar a la generación de un tren de medios (audio, vídeo, etc.), el tren de medios NO SE DEBE reenviar a ninguna conexión asociada con ese punto extremo, independientemente del modo de la conexión. Por ejemplo, si se aplica un tono de indicación de llamada en espera a un punto extremo involucrado en una llamada activa, solamente la parte que utilice el punto extremo en cuestión oírá el tono de indicación de llamada en espera. Sin embargo, las señales individuales pueden definir un comportamiento diferente.

Cuando se aplica una señal a una conexión que ha recibido un `RemoteConnectionDescriptor` (descriptor de conexión distante) (véase 6.3.3), el tren de medios generado por dicha señal DEBE ser reenviado por la conexión *con independencia del* modo actual de la conexión. Si no se ha recibido un `RemoteConnectionDescriptor`, la pasarela DEBE devolver un error (código de error 527 – falta `RemoteConnectionDescriptor`).

Cuando se aplica una lista (posiblemente vacía) de señales, esta lista sustituye completamente a la lista actual de señales de temporización activas. Las señales de temporización en ese momento activas que no son proporcionadas en el nueva lista DEBEN ser detenidas, y las nuevas señales proporcionadas pasarán ahora a ser activas. Las señales de temporización actualmente activas que son proporcionadas en la nueva lista de señales DEBEN permanecer activas sin interrupción, por lo que el temporizador para tales señales de temporización no se verá afectado. En consecuencia, no hay normalmente ningún medio de reiniciar el temporizador para una señal de temporización actualmente activa sin desactivar antes la señal. Si la señal de temporización está parametrizada, el conjunto original de parámetros DEBE permanecer vigente, sin tener en cuenta los valores proporcionados posteriormente. Una señal determinada NO DEBE aparecer más de una vez en una lista `SignalRequests`.

Las señales definidas normalmente se pueden ver en el anexo A.

⁸ La acción "mantener señal (es) activa(s)" puede anular este comportamiento.

⁹ Si se han transferido parámetros a la señal, los parámetros no serán comunicados.

RequestedEvents (eventos solicitados) es una lista de eventos que la pasarela debe detectar en el punto extremo. Salvo que se especifique lo contrario, los eventos son detectados en el punto extremo, aunque algunos eventos pueden ser detectados en una conexión. Son ejemplos de eventos los siguientes¹⁰:

- transición a colgado (cuando en un aparato telefónico clásico el usuario cuelga el microteléfono);
- transición a descolgado (cuando en un aparato telefónico clásico el usuario descuelga el microteléfono);
- dígitos DMTF (o dígitos de impulsos).

En el anexo A se pueden ver los eventos definidos normalmente.

A cada evento están asociadas una o más **actions (acciones)** que definen las acciones que debe acometer la pasarela cuando ocurre el evento en cuestión. Las acciones posibles son:

- Notify (Notificar) el evento inmediatamente, junto con la lista acumulada de eventos observados.
- Accumulate (acumular) el evento.
- Accumulate according to digit map (acumular de acuerdo con mapa de dígitos),
- Ignore the event (ignorar el evento).
- Keep signal(s) active [mantener la(s) señal(es) activa(s)].
- Embedded NotificationRequest (NotificationRequest incorporada).
- Embedded ModifyConnection (ModifyConnection incorporada).

El punto extremo detectará dos conjuntos de eventos solicitados: persistente y no persistente.

Los eventos persistentes se detectan siempre en un punto extremo. Si un evento persistente no está incluido en la lista de RequestedEvents, y el evento ocurre, el evento será detectado en cualquier caso y procesado como los otros eventos, del mismo modo que si el evento persistente hubiera sido pedido con una acción Notify (notificar)¹¹. De este modo, de manera informal, puede considerarse que los eventos persistentes están siempre incluidos implícitamente en la lista de RequestedEvents mediante una acción Notify, aunque no se realizará¹² ninguna detección etc., clara. Los eventos se identifican como persistentes por medio de su definición – véase el anexo A.

Son eventos no persistentes aquellos que deben incluirse explícitamente en la lista de RequestedEvents. La lista (posiblemente vacía) de eventos solicitados reemplaza completamente a la lista anterior de eventos solicitados. Además de los eventos persistentes, el punto extremo solamente detectará los eventos especificados en la lista de eventos solicitados. Si se incluye un evento persistente en la lista de RequestedEvents, la acción especificada sustituirá a la acción por defecto asociada con el evento durante la vida de la lista de RequestedEvents, después de lo cual se restablecerá la acción por defecto. Por ejemplo, si se ha especificado "Ignore off-hook" ("ignorar descolgar"), y se ha recibido una nueva petición sin ninguna instrucción de descolgar, la operación por defecto "Notify off-hook" (notificar descolgar) debería ser entonces restaurada. Un evento concreto NO DEBE aparecer más de una vez en una lista de RequestedEvents.

Se puede especificar más de una acción para un evento, si bien una acción determinada no puede aparecer más de una vez para un evento dado. En la siguiente matriz se recogen las combinaciones legales de acciones:

¹⁰ Estos son meros ejemplos tomados del ejemplo de paquete de línea en el apéndice I.

¹¹ Por ello el RequestIdentifier será el RequestIdentifier de la NotificationRequest actual.

¹² Normalmente, si se efectúa una petición de esperar, por ejemplo descolgar, la petición solamente será fructuosa si el teléfono no ha sido todavía descolgado.

	Notify	Accumulate	Accumulate according to digit map	Ignore	Keep Signal(s) Active	Embedded Notification Request	Embedded Modify Connection
Notify	–	–	–	–	√	–	√
Accumulate	–	–	–	–	√	√	√
Accumulate according to digit map	–	–	–	–	√	–	√
Ignore	–	–	–	–	√	–	√
Keep Signal(s) active	√	√	√	√	–	√	√
Embedded Notification Request	–	√	–	–	√	–	√
Embedded Modify Connection	√	√	√	√	√	√	–

Si un cliente recibe una petición con una acción no válida o una combinación ilegal de acciones, DEBE devolver un error al agente de llamada (código de error 523 – combinación de acciones desconocida o ilegal).

Si se especifican varias acciones, por ejemplo, "Keep signal(s) active" y "Notify", se supone que las acciones individuales ocurren a la vez.

El agente de llamada puede enviar a la pasarela una NotificationRequest con una lista de RequestedEvents vacía. El agente de llamada puede actuar de este modo, por ejemplo, con un cliente incorporado cuando éste no desea recopilar más dígitos DTMF. Sin embargo, los eventos persistentes seguirán siendo detectados y notificados.

DigitMap (mapa de dígitos) es un parámetro opcional que permite al agente de llamada proporcionar al punto extremo un mapa de dígitos conforme al cual serán acumulados los dígitos cuando el agente de llamada suministre un parámetro RequestedEvents con la acción "accumulate according to digit map" para ese punto extremo. El mapa de dígitos suministrado es persistente y por tanto no es necesario proporcionarlo cada vez que se efectúe una petición de "accumulate according to digit map"; no obstante, los agentes de llamada pueden proporcionar un mapa de dígitos en cualquier momento. Un mapa de dígitos para el punto extremo DEBE suministrarse a más tardar con la primera petición de "accumulate according to digit map". Si se pide a la pasarela "accumulate according to digit map" y la pasarela no dispone en ese momento de un mapa de dígitos para el punto extremo en cuestión, la pasarela deberá devolver un error (código de error 519 –el punto extremo no tiene un mapa de dígitos).

Cada uno de los puntos extremos tiene una variable denominada "cadena de marcación vigente" ("current dial string") en la cual se recopilan dígitos para que concuerden con el mapa de dígitos, tal como se especifica en 6.1.5. Siempre que se envía una Notify (notificar) o se ha de procesar una NotificationRequest (petición de notificación), la "cadena de marcación vigente" se inicializa con una cadena nula. Los dígitos que se han de procesar pueden ahora, bien ser detectados como entrada, o bien ser recuperados de una zona de registro de entradas de eventos conocida como "memoria intermedia de (mantenimiento en) cuarentena" ("quarantine buffer") – en 6.4.3.1 se dan más detalles.

Las señales aplicadas por SignalRequests son sincronizadas con el conjunto de eventos especificados o implícitos en el parámetro RequestedEvents, excepto si son sustituidas por la acción "Keep signal(s) active". Por ejemplo, si la NotificationRequest ordenó una señal de "timbre" y la petición de evento pidió que se esperara un evento "descolgar", la señal de timbre debe, por defecto, detenerse tan pronto como la pasarela haya detectado un evento descolgar. Si "descolgar" se ha definido como un evento persistente y la petición de evento no pidió esperar un evento "descolgar", el timbre de detendría en cualquier caso puesto que descolgar estaría entonces implícito en el parámetro RequestedEvents. La definición formal es que la generación de todas las señales "temporización" DEBE detenerse tan pronto como se detecta uno de los eventos requeridos, a menos que la acción "Keep signal(s) active" esté asociada al evento especificado. En el caso de la acción "accumulate according to digit map", el comportamiento por defecto sería detener todas las señales de temporización activas cuando se acumula el primer dígito¹³ – es irrelevante para esta sincronización si los dígitos acumulados dan como resultado un concordancia, una discordancia o una concordancia parcial con el mapa de dígitos.

Si se desea que las señales de temporización continúen cuando ocurre un evento "de espera" (looked-for), puede utilizarse la acción "Keep signal(s) active". Esta acción tiene el efecto de mantener activas todas las señales de temporización que lo están actualmente, negando de ese modo la parada por defecto de las señales de temporización tras la ocurrencia del evento.

Si se desea que las señales arranquen cuando ocurre un evento "de espera", se puede utilizar la acción "Embedded NotificationRequest". La NotificationRequest incorporada puede incluir también una nueva lista de RequestedEvents, SignalRequests y un nuevo mapa de dígitos. Sin embargo, la "Embedded NotificationRequest" no puede incluir otra "Embedded NotificationRequest". Cuando es activada la "Embedded NotificationRequest", se liberará la "cadena de marcación vigente"; la lista de eventos observados y la memoria de cuarentena no se verán afectados (véase 6.4.3.1).

La acción NotificationRequest incorporada permite al agente de llamada establecer un "mini-script" para ser procesado por la pasarela inmediatamente después de la detección del evento asociado. Cualquier SignalRequests especificada en la NotificationRequest incorporada arrancará inmediatamente. Debe prestarse una atención considerable para evitar que haya discrepancias entre el agente de llamada y la pasarela. Sin embargo, las discrepancias a largo plazo no deberían ocurrir puesto que la nueva SignalRequests reemplaza enteramente a la antigua lista de señales de temporización activas, y las señales de tipo BR se detienen siempre por sí mismas. Se recomienda limitar el número de señales del tipo activado/desactivado (On/Off). Se considera una buena práctica que el agente de llamada active de vez en cuando todas las señales On/Off que deban estar activadas y desactive todas las señales On/Off que deban estar desactivadas.

Si se desea que los modos de conexión cambien cuando ocurre un evento de espera, puede utilizarse la acción "Embedded ModifyConnection". La acción ModifyConnection incorporada puede incluir una lista de cambios de modo de conexión formado cada uno de ellos por el cambio de modo y el identificador de conexión afectado. El comodín "\$" se puede utilizar para indicar la "conexión actual"; sin embargo esta notación NO DEBE utilizarse fuera de una instrucción de tratamiento de la conexión – el comodín se refiere a la conexión en cuestión para la instrucción de tratamiento de la conexión.

La acción ModifyConnection incorporada permite al agente de llamada ordenar al punto extremo que cambien el modo de conexión de una o más conexiones inmediatamente después de la detección del evento asociado. Cada uno de los cambios del modo de conexión actúa de manera similar a la correspondiente instrucción ModifyConnection¹⁴. Cuando se suministra una lista de cambios de

¹³ Dígito definido en los mapas de dígitos, es decir, incluidos asterisco, temporizador, etc.

¹⁴ De este modo, si, por ejemplo, se utiliza D-QoS en la conexión, se ejecutará todavía la acción D-QoS por defecto cuando se lleva a cabo la acción ModifyConnection incorporada.

modos de conexión, los cambios de modo de conexión DEBEN aplicarse uno cada vez y de izquierda a derecha. Cuando han finalizado todos los cambios de modo de conexión, se generará un evento "operación completada" parametrizado con el nombre de acción completada (véase el anexo A para más detalles). Si fracasa alguno de los cambios de modo de conexión, se generará un evento "operación fracasada" parametrizado con el nombre de la acción y el cambio de modo de conexión (véase el anexo A para más detalles) – el resto de los cambios de modo de conexión NO DEBEN ser intentados, y los cambios de modo de conexión fructuosos anteriores de la lista NO DEBEN ser cambiados tampoco.

Por último, se puede utilizar la acción ignorar para pasar por alto un evento, por ejemplo, para impedir que un evento persistente sea notificado. Sin embargo, todavía ocurrirá por defecto la sincronización entre el evento y una señal activa.

En la cláusula 6.4.3.1 se recogen detalles adicionales sobre la semántica de la detección e informe de eventos. Se alienta al lector a estudiarla cuidadosamente.

La definición específica de las acciones solicitadas a través de estas listas SignalRequests (por ejemplo, la duración y frecuencia de un dígito DTMF) cae fuera del alcance de la especificación básica de NCS. Esta definición puede variar de una ubicación a otra y, por ello, de una pasarela a otra. En consecuencia, las definiciones se proporcionan en lotes de eventos, los cuales pueden ser suministrados fuera de la especificación base. En el anexo A se presenta una lista inicial de lotes de eventos.

Las listas RequestedEvents y SignalRequests generalmente se refieren a los mismos eventos. En un caso se pide a la pasarela que detecte la ocurrencia del evento, y en el otro caso se la pide que lo genere. Hay excepciones a esta regla, por ejemplo, los tonos de fax y módem, los cuales pueden ser detectados pero no puede pedirse su generación. Sin embargo, no podemos necesariamente esperar que todos los puntos extremos detecten todos los eventos. Los eventos y señales específicos que un punto extremo determinado puede detectar o generar vienen determinados por la lista de lotes de eventos soportados por ese punto extremo. Cada lote especifica una lista de eventos y señales que pueden ser detectados o aplicados. Una pasarela a la que se pide detectar o aplicar un evento que no es soportado por el punto extremo especificado DEBE devolver un error (código de error 512 ó 513 – no equipado para detectar evento o generar señal). Cuando el nombre del evento no es calificado por un nombre de lote, se adopta el nombre de lote por defecto para el punto extremo. Si el nombre del evento no está registrado en este lote por defecto, la pasarela DEBE devolver un error (código de error 522 – no admitido tal evento o señal).

El agente de llamada puede enviar una NotificationRequest cuya lista de señales solicitadas está vacía. Esto puede producir la parada de todas las señales de temporización activas. Puede hacerse así cuando, por ejemplo, debe detenerse la generación de tonos, como la señal de llamada de retorno.

QuarantineHandling es un parámetro opcional que especifica las opciones de tratamiento para la memoria intermedia de cuarentena (véase 6.4.3.1). Permite al agente de llamada especificar si deben procesarse o descartarse los eventos en cuarentena. Si el parámetro está ausente, los eventos en cuarentena DEBEN ser procesados.

DetectEvents es un parámetro opcional que especifica una lista mínima de eventos que se pide a la pasarela que detecte en la "notificación" y el estado "lockstep". La lista es persistente hasta que se especifica un nuevo valor. En 6.4.3.1 se presenta una explicación más detallada de este parámetro.

ReturnCode (código de retorno) es un parámetro devuelto por la pasarela. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

6.3.2 Notificaciones

Las notificaciones son enviadas por la pasarela mediante la instrucción Notify cuando se ha de notificar un evento observado:

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

EndpointId es el nombre del punto extremo en la pasarela, el cual está enviando la instrucción Notify, como se define en 6.1.1. El identificador DEBE ser un nombre de punto extremo totalmente calificado, que incluya el nombre de dominio y la pasarela. La parte local del nombre NO DEBE utilizar el convenio de comodines.

NotifiedEntity es un parámetro opcional que identifica la entidad a la cual se envía la notificación. Este parámetro es igual al parámetro NotifiedEntity de la NotificationRequest que desencadenó esta notificación. El parámetro está ausente si no había ninguno de tales parámetros en la petición de desencadenamiento. Con independencia del valor del parámetro "NotifiedEntity", la notificación DEBE ser enviada a la "entidad notificada" actual para el punto extremo.

RequestIdentifier es un parámetro que repite el parámetro RequestIdentifier de la NotificationRequest que desencadenó esta notificación. Se utiliza para relacionar esta notificación con la petición de notificación que la desencadenó. Los eventos persistentes serán contemplados aquí como si se hubieran incluido en la última NotificationRequest. Cuando no se ha recibido ninguna NotificationRequest, el RequestIdentifier utilizado será cero ("0").

ObservedEvents es una lista de los eventos que la pasarela ha detectado y acumulado, sea por la acción "accumulate", "accumulate according to digit map" o "notify". Una sola notificación puede informar sobre una lista de eventos que serán comunicados en el orden en que fueron detectados. La lista solamente puede contener eventos persistentes y eventos que fueron solicitados en el parámetro RequestedEvents de la NotificationRequest desencadenante. Los eventos que fueron detectados en una conexión incluirán el nombre de esta conexión. La lista contendrá los eventos que fueron, bien acumulados (pero no notificados) o bien acumulados de conformidad con el mapa de dígitos (pero sin concordancia todavía), y el evento final que desencadenó la notificación o proporcionó una concordancia final en el mapa de dígitos. Debe señalarse que los dígitos son añadidos a la lista de eventos observados a medida que son acumulados, independientemente de que hayan sido acumulados de conformidad con el mapa de dígitos o no. Por ejemplo, si un usuario introduce los dígitos "1234" y un evento E es acumulado entre los dígitos "3" y "4" que se están introduciendo, la lista de eventos observados sería "1, 2, 3, E, 4".

ReturnCode es un parámetro devuelto por la pasarela. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

6.3.3 CreateConnection (crear conexión)

Esta instrucción se utiliza para crear una conexión.

```
ReturnCode
, ConnectionId
[, SpecificEndPointId]
, LocalConnectionDescriptor
[, ResourceID]
    ← CreateConnection(CallId
                       , EndpointId
                       [, NotifiedEntity]
                       , LocalConnectionOptions
                       , Mode
                       [, RemoteConnectionDescriptor])
```

```
[, RequestedEvents]
[, RequestIdentifier]
[, DigitMap]
[, SignalRequests]
[, QuarantineHandling]
[, DetectEvents])
```

Se utiliza esta función cuando se establece una conexión entre dos puntos extremos. Una conexión está definida por sus atributos y los puntos extremos asociados. Los parámetros de entrada de CreateConnection proporcionan los datos necesarios para construir uno de los dos puntos extremos "vistos" de una conexión.

CallId (identificador de llamada) es un parámetro que identifica la llamada (o sesión) a la que pertenece la conexión. Este parámetro es exclusivo, como mínimo, dentro del conjunto de agentes de llamada que controlan las mismas pasarelas; las conexiones que pertenecen a la misma llamada comparten el mismo id de llamada. El id de llamada puede utilizarse para identificar llamadas a los fines de informe y contabilidad.

EndPointId (identificador de punto extremo) es el identificador del punto extremo en la pasarela en la cual se ejecuta CreateConnection. El punto extremo puede especificarse totalmente asignando un valor que no sea comodín al parámetro EndPointId en la función llamada o puede ser subespecificado utilizando el convenio de comodín "anyone" ("cualquiera"). Si el punto extremo está subespecificado, el identificador de punto extremo será asignado por la pasarela y su valor completo devuelto en el parámetro **SpecificEndPointId** de la respuesta. El convenio de comodín "all" ("todos") NO DEBE utilizarse.

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

LocalConnectionOptions (opciones de conexión local) es una estructura que describe las características de la conexión de datos de medios desde el punto de vista de la pasarela que ejecuta CreateConnection. Instruye al punto extremo acerca de las características de emisión y recepción de la conexión de medios. Los campos básicos contenidos en LocalConnectionOptions son:

- **Encoding Method (método de codificación):** DEBE especificarse una lista de nombres literales del algoritmo de compresión (codificación/decodificación) utilizado para enviar y recibir medios por la conexión con un valor como mínimo. La inserciones en la lista se ordenan por preferencia. El punto extremo DEBE seleccionar solamente uno de los códecs, y éste DEBERÍA elegirse de acuerdo con la preferencia indicada. Si el punto extremo recibe algún tren de medios por la conexión codificado con un método de codificación diferente, éste tren PUEDE ser descartado. El punto extremo DEBE además indicar cuáles de los algoritmos de compresión restantes está dispuesto a soportar como alternativas – en 7.4.1 se dan más detalles. En un documento de IPCablecom separado se especifica una lista de los métodos de codificación admisibles.
- **Packetization Period (periodo de paquetización):** El periodo de paquetización en milisegundos, como se define en la norma SDP (RFC 2327), DEBE especificarse con un valor solamente. El valor únicamente pertenece al tren de medios enviado. En un documento de IPCablecom separado se especifica una lista de periodos de paquetización admisibles.
- **Echo Cancellation (cancelación de eco):** Determina si debe utilizarse o no la cancelación de eco en el lado de línea¹⁵. El parámetro puede tener el valor "activado" ("on") (si se requiere cancelación de eco) o "desactivado" ("off") (si no se requiere). El parámetro es opcional. Cuando se ha omitido el parámetro, el cliente incorporado DEBE aplicar la cancelación de eco.

¹⁵ No se soporta la cancelación de eco en el lado paquetes.

- **Type of Service (tipo de servicio):** Especifica la clase de servicio que se utilizará para la emisión de medios por la conexión mediante la codificación del parámetro valor de tipo de servicio de 8 bits del encabezamiento IP como dos dígitos hexadecimales. El parámetro es opcional. Cuando se ha omitido el parámetro, se aplica un valor por defecto de A0_H correspondiente a una fijación de bits de precedencia IP de cinco.
- **Silence Suppression (supresión de silencio):** Determina si ha de utilizarse o no la supresión de silencio en el sentido de emisión. El parámetro puede tener el valor "activado" ("on") (cuando ha de suprimirse el silencio) o "desactivado" ("off") (cuando no ha de suprimirse el silencio). El parámetro es opcional. Cuando se ha omitido el parámetro, el valor por defecto es no utilizar la supresión de silencio.

Para soportar la calidad de servicio dinámica (D-QoS, *dynamic quality of service*) se utilizan los campos LocalConnectionOptions siguientes (en el anexo B se recogen más detalles al respecto):

- **D-QoS GateID (identificador de puerta de D-QoS):** El GateID es el identificador de puerta para la puerta que ha sido establecida en el encaminador en borde. El Gate-ID es un identificador de 32 bits codificado como una cadena de hasta 8 caracteres hexadecimales. Este parámetro es opcional por lo general, pero es obligatorio cuando ha de realizarse una reserva y/o compromiso de recursos D-QoS. La presencia de este parámetro implica que ha de realizarse D-QoS para esta instrucción, mientras que su ausencia implica que D-QoS no ha de efectuarse.
- **D-QoS Resource Reservation (reserva de recursos D-QoS):** Permite el control explícito acerca de si la reserva y/o el compromiso de recursos D-QoS deben ejecutarse o no en el sentido de emisión y/o recepción. El parámetro es opcional y puede adoptar uno o más de los siguientes valores:

Valores de reserva:

- "SendReserve" Los recursos se reservan solamente en el sentido de emisión.
- "ReceiveReserve" ("reserva en recepción") Los recursos se reservan solamente en el sentido de recepción.
- "SendReceiveReserve" ("reserva en emisión y recepción") Los recursos se reservan en los sentidos de emisión y recepción.

Valores de compromiso:

- "SendCommit" ("compromiso en emisión") Los recursos se comprometen solamente en el de sentido de emisión.
- "ReceiveCommit" ("compromiso en recepción") Los recursos se comprometen solamente en el sentido de recepción.
- "SendReceiveCommit" ("compromiso en emisión y en recepción") Los recursos se comprometen en el sentido de emisión y de recepción.

El parámetro es opcional, y en el caso de varios valores éstos se separan mediante comas. Cuando ha de realizarse la D-QoS, y el parámetro, bien ha sido omitido o bien no tiene ningún valor presente, la reserva de recursos DEBE efectuarse para ambos sentidos, de emisión y de recepción. Los recursos reservados están determinados por los parámetros de codificación aplicados a la conexión, es decir, método de codificación, periodo de paquetización, supresión de silencio, ciphersuite, etc. Los parámetros externos, como el empleo de la supresión del encabezamiento de cabida útil, pueden afectar también a la cantidad de recursos reservados –para más detalles véase la Especificación UIT-T J.163 sobre calidad de servicio dinámica de IPCablecom.

Los recursos de recepción pueden ser reservados y comprometidos sin que se haya obtenido previamente un RemoteConnectionDescriptor, mientras que los recursos en emisión pueden ser reservados, pero no comprometidos, hasta que se suministre un RemoteConnectionDescriptor. Cuando haya que efectuar la reserva de D-QoS y el parámetro, bien ha sido omitido o bien no está presente, SE DEBEN comprometer los recursos por defecto basados en el modo de conexión tal como se especifica en el cuadro a continuación:

Modo de conexión	D-QoS
"inactive"	No comprometido
"send only", "replicate"	Comprometido en emisión
"receive only"	Comprometido en recepción
"send/receive", "conference", "network loopback", "network continuity test"	Comprometido en emisión y recepción

Si se desea una operación de compromiso diferente, se proporciona el valor de compromiso adecuado y se utilizará en su lugar. Si ha de realizarse una operación de compromiso pero no se ha hecho ninguna reserva, o una reserva existente no satisface plenamente los recursos que han de comprometerse¹⁶, se efectuará una reserva automáticamente. Si se especifica un valor de reserva pero no se especifica ningún valor de compromiso, no se efectuará una operación de compromiso.

- **ResourceID (id de recursos):** ResourceID existente para recursos ya reservados en el encaminador en borde. La utilización del ResourceID permite separar las reservas relativas al mismo recurso; sin embargo sólo puede activarse una de las reservas en un momento dado. El ResourceID es un identificador de 32 bits codificado como una cadena de hasta 8 caracteres hexadecimales. El parámetro es opcional.
- **ReserveDestination (destino de la reserva):** Este parámetro opcional puede especificar una dirección del protocolo Internet versión 4 (IPv4), seguida opcionalmente por un carácter dos puntos y un número de puerto UDP, que es el destino de la reserva de recursos. Cuando no se especifica un número de puerto UDP, se aplica un valor por defecto de 9. El parámetro ReserveDestination se utiliza típicamente cuando ha de realizarse la reserva de recursos y aún no se ha proporcionado el RemoteConnectionDescriptor para la conexión. Esto permite enviar las reservas y los compromisos de tren descendente al encaminador en borde cuando no se conoce todavía la fuente de un tren de medios¹⁷. Cuando se ha suministrado un RemoteConnectionDescriptor, el parámetro se ignora.

Para soportar los servicios de seguridad de IPCablecom se utilizan los siguientes campos LocalConnectionOptions:

- **Secret (secreto):** El campo opcional secret es un valor semilla que DEBE utilizarse para derivar las claves de criptación de extremo a extremo de los servicios de seguridad RTP y RTCP definidos en la especificación de seguridad IPCablecom (en desarrollo). El campo secret DEBERÍA codificarse en texto claro si solamente contiene valores de la gama de caracteres ASCII de 21_H a 7E_H. En caso contrario, el campo secret DEBE codificarse utilizando la codificación en base 64. Si no se suministra ningún valor, o si se ha omitido el parámetro y no se van a utilizar los servicios de seguridad, el punto extremo DEBE generar

¹⁶ Esto no es posible para la instrucción CreateConnection pero se señala aquí para una exposición completa. Es posible, sin embargo, para la instrucción ModifyConnection (véase 6.3.4).

¹⁷ Obsérvese que esto posibilitaría determinados escenarios de robo de servicio. Véase la Especificación de calidad de servicio dinámica (UIT-T J.163) para más detalles.

un campo secret por sí mismo¹⁸. Cuando la CA proporciona un campo secret, éste DEBERÍA utilizarse.

- **RTP ciphersuite (serie de texto cifrado RTP):** Lista de ciphersuites para la seguridad RTP por orden de preferencia. Las inserciones en la lista se ordenan por preferencia, donde la primera ciphersuite es la elección preferida. El punto extremo DEBE elegir solamente una de las ciphersuites. El punto extremo DEBERÍA además indicar cuales de las ciphersuites restantes está dispuesto a soportar como alternativas (véase 7.4.1 para más detalles). Cada ciphersuite está representada por cadenas ASCII formadas por dos subcadenas (que pueden estar vacías) separadas por una barra de fracción ("/"), donde la primera subcadena identifica el algoritmo de autenticación y la segunda subcadena el algoritmo de criptación. En la especificación de seguridad de IPCablecom (en desarrollo) se presenta una lista de ciphersuites admisibles.
- **RTCP ciphersuite (serie de texto cifrado RTCP):** Lista de ciphersuites para la seguridad de RTCP por orden de preferencia. Las inserciones en la lista están ordenadas por preferencia donde la primera ciphersuite es la elección preferida. El punto extremo DEBE elegir solamente una de las ciphersuites. El punto extremo DEBERÍA además indicar cuales de las ciphersuites restantes está dispuesto a soportar como alternativas (véase 7.4.1 para más detalles). Cada ciphersuite está representada por cadenas ASCII formadas por dos subcadenas (posiblemente vacías) separadas por una barra de fracción ("/"), donde la primera subcadena identifica el algoritmo de autenticación y la segunda subcadena el algoritmo de criptación. En la Especificación UIT-T J.170 de seguridad de IPCablecom (en desarrollo) se presentará una lista de ciphersuites admisibles.

El cliente incorporado DEBE responder con un error (código de error 524 –incoherencia de LocalConnectionOptions) si es violada alguna de las reglas anteriores. Todos los valores por defecto mencionados anteriormente pueden ser alterados por el proceso de aprovisionamiento.

RemoteConnectionDescriptor (descriptor de conexión distante): Descriptor de conexión para el lado distante de una conexión, en el otro lado de la red IP. Incluye los mismos campos que el LocalConnectionDescriptor (que no debe confundirse con LocalConnectionOptions), es decir, los campos que describen una sesión según la norma SDP. En la cláusula 7.4 se dan detalles sobre la utilización soportada de SDP en el perfil NCS. Este parámetro puede tener un valor nulo cuando no se conoce la información para el extremo distante. Esto sucede porque la entidad que establece una conexión comienza enviando una instrucción CreateConnection a una de las dos pasarelas involucradas. Para la primera CreateConnection expedida no se dispone de ninguna información referente al otro lado de la conexión. Esta información puede proporcionarse posteriormente a través de una llamada ModifyConnection.

El perfil NCS supone normalmente que se aplican los mismos parámetros de medios a una conexión en ambos sentidos de emisión y de recepción. Por consiguiente, parte de la información del RemoteConnectionDescriptor es redundante y hay una posibilidad de que aparezca una incoherencia con LocalConnectionOptions. Es sin embargo responsabilidad plena del agente de llamada garantizar que envía instrucciones coherentes a cada punto extremo a fin de asegurar que se especifican parámetros de medios coherentes. No obstante, si una pasarela detecta una incoherencia, LocalConnectionOptions tomará sencillamente la prioridad. Cuando los códecs han cambiado durante una llamada, puede haber periodos pequeños de tiempo en los que los puntos extremos utilicen códecs diferentes. Como se ha estipulado anteriormente, los clientes incorporados PUEDEN descartar los trenes de medios recibidos que están codificados con un códec diferente del especificado en LocalConnectionOptions para una conexión.

¹⁸ Esto incluye la generación de un nuevo secreto y la utilización de un secreto suministrado en un RemoteConnectionDescriptor.

Mode (modo) indica el modo de operación para este lado de la conexión. Las opciones son "send only" (enviar solamente), "receive only" (recibir solamente), "send/receive" (enviar/recibir), "conference" (conferencia), "inactive" (inactivo), "replicate" (repetir), "network loopback" (bucle de red) o "network continuity test" (prueba de continuidad de red). El tratamiento de estos modos se especifica al principio de 6.3. Puede ocurrir que algunos puntos extremos no sean capaces de soportar todos los modos. Si la instrucción especifica un modo que el punto extremo no soporta, DEBE devolverse un error (código de error 517 – modo no soportado). Asimismo, si una conexión no ha recibido todavía un RemoteConnectionDescriptor, DEBE devolverse un error si se intenta establecer la conexión en alguno de los modos "send only", "send/receive", "replicate" o "conference" (código de error 527 – falta RemoteConnectionDescriptor).

ConnectionId (id de conexión) es un parámetro devuelto por la pasarela que identifica inequívocamente la conexión dentro del contexto del punto extremo en cuestión.

LocalConnectionDescriptor (descriptor de conexión local) es un parámetro devuelto por la pasarela que describe la sesión y contiene información acerca de, por ejemplo, direcciones y puertos RTP para conexiones de "RI" como se definen en SDP. Es similar al RemoteConnectionDescriptor, salvo que especifica este lado de la conexión. En la cláusula 7.4 se dan detalles sobre la utilización admitida de SDP en el perfil NCS.

Cuando se recibe una instrucción "CreateConnection" que no incluye un parámetro RemoteConnectionDescriptor, una pasarela se encuentra en una situación ambigua en cuanto a la conexión en cuestión. Como la pasarela ha exportado un parámetro LocalConnectionDescriptor, puede recibir paquetes por dicha conexión. Como no ha recibido todavía el otro parámetro de pasarela RemoteConnectionDescriptor, no sabe si los paquetes que recibe han sido autorizados por el agente de llamada. Por ello debe navegar entre dos riesgos, a saber, recortando algunos avisos importantes o escuchando datos perturbados. El comportamiento de la pasarela está determinado por el valor del parámetro modo (sujeto a seguridad):

- Si se ha fijado el modo a "receive only", la pasarela DEBE aceptar las señales de voz recibidas por la conexión y transmitir las a través del punto extremo.
- Si el modo se ha puesto a "inactive", la pasarela DEBE (como siempre) descartar las señales vocales recibidas por la conexión.
- Si el modo se ha fijado a "network loopback" o "network continuity test" la pasarela DEBE configurar la respuesta o eco esperado. El tren de medios generado o devuelto en eco DEBE entonces ser enviado a la fuente del tren de medios recibido.
- Obsérvese que cuando el punto extremo no tiene un RemoteConnectionDescriptor para la conexión, ésta no puede por definición encontrarse en ninguno de los modos "send only", "send/receive", "replicate" o "conference".

Los parámetros **RequestedEvents**, **RequestIdentifier**, **DigitMap**, **SignalRequests**, **QuarantineHandling** y **DetectEvents** pueden ser todos opcionales. Pueden ser utilizados por el agente de llamada para incluir efectivamente una petición de notificación que sea ejecutada 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 ser reconocida por la presencia de un RequestIdentifier. El resto de los parámetros puede o no estar presente. Si no está presente uno de los parámetros, la petición de notificación DEBE ser tratada como si fuese una NotificationRequest normal con el parámetro en cuestión omitido. Esto puede producir la cancelación de señales y la detención de la espera de eventos.

A título de ejemplo de utilización, consideremos que un agente de llamada desea realizar una llamada a un cliente incorporado. El agente de llamada debe:

- pedir al cliente incorporado que cree una conexión para garantizar asegurar que el usuario pueda comenzar a hablar tan pronto como el teléfono descuelga,

- pedir al cliente incorporado que inicie el tono de llamada,
- pedir al cliente incorporado que notifique al agente de llamada cuándo el teléfono descuelga.

Todo lo anterior puede configurarse en una sola instrucción `CreateConnection` mediante la inclusión de una petición de notificación con los parámetros `RequestedEvents` para el evento descolar y el parámetro `SignalRequests` para la señal de llamada.

Cuando estos parámetros están presentes, la creación de la conexión y la petición de notificación DEBEN estar sincronizadas, lo cual significa que ambas sean o aceptadas o rechazadas. En nuestro ejemplo, la instrucción `CreateConnection` debe ser rechazada si la pasarela no dispone de suficientes recursos o no puede conseguir los recursos adecuados del acceso de red local. La petición de notificación de descolar debe rechazarse en la condición de doble toma si el usuario está ya descolar. En este ejemplo, el teléfono no debe sonar si no se puede establecer la conexión, y la conexión no se debe establecer si el usuario está ya descolar. En su lugar debería devolverse un error, (código de error 401 –teléfono descolar), que informa al agente de llamada de la condición de doble toma.

ReturnCode es un parámetro devuelto por la pasarela. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

ResourceID es un parámetro D-QoS que puede ser devuelto por la pasarela. Cuando se realiza con éxito una reserva de recursos D-QoS, el `ResourceID` proporciona un título para los recursos reservados.

6.3.4 **ModifyConnection (modificar conexión)**

Esta instrucción se utiliza para modificar las características de la "visión" que tiene la pasarela de una conexión. Esta "visión" de la llamada incluye tanto el descriptor de la conexión local como el descriptor de la conexión distante.

```

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

```

Los parámetros utilizados son los mismos que en la instrucción `CreateConnection`, con el añadido de un **ConnectionId** que identifica inequívocamente la conexión dentro del punto extremo. Este parámetro es devuelto por la instrucción `CreateConnection` junto con el descriptor de conexión local. Con ello se identifica inequívocamente la conexión dentro del contexto del punto extremo.

El **EndpointId** DEBE ser un nombre de punto extremo totalmente calificado. El nombre local NO DEBE utilizar el convenio de comodín.

La instrucción `ModifyConnection` se puede utilizar para incidir en los parámetros de la conexión, de acuerdo con las mismas reglas y constricciones especificadas para `CreateConnection`:

- Proporcionar información en el otro extremo de la conexión a través del **RemoteConnectionDescriptor**.

- Activar o desactivar la conexión cambiando el valor del parámetro **modo**. Esto puede ocurrir en cualquier momento de la conexión con valores de parámetro arbitrarios. Una activación puede, por ejemplo, fijar la conexión al modo "receive only".
- Cambiar los parámetros de la conexión a través de **LocalConnectionOptions**, por ejemplo conmutando a un esquema de codificación diferente, cambiando el periodo de paquetización o modificando el tratamiento de la cancelación de eco.

Los detalles de la operación D-QoS se especificaron en la instrucción CreateConnection, y aquí se utilizan por lo general las mismas reglas, salvo las que se señalan a continuación:

- **D-QoS GateID**: Un GateID D-QoS es obligatorio cuando se necesita la operación D-QoS, a menos que se haya efectuado anteriormente la operación D-QoS para la conexión en cuestión. En el último caso se utilizará el GateID D-QoS suministrado anteriormente.
- **D-QoS Resource Reservation**: Permite el control explícito acerca de si debe realizarse o no la reserva y/o el compromiso de recursos D-QoS en el sentido de emisión y/o recepción. El parámetro es opcional y se pueden especificar varios valores. Cuando se ha omitido el parámetro y ha de realizarse la reserva D-QoS, la solución por defecto es reservar en ambos sentidos, de emisión y de recepción, salvo que se haya efectuado ya una reserva adecuada para la conexión (véase el anexo B). En tal caso, no se realizará una nueva reserva. Los recursos se comprometen del mismo modo que en CreateConnection, excepto cuando se cambia al modo "inactive". En ese caso, los recursos comprometidos DEBEN ser reducidos a cero. No obstante, todavía se mantiene una reserva de recursos existente.
- **ResourceID**: El parámetro es opcional. Cuando se suministra sustituye al ResourceID mantenido por el cliente incorporado para la conexión.
- **ReserveDestination**: El parámetro es opcional. Cuando se suministra sustituye al ReserveDestination mantenido por el cliente incorporado para la conexión. Si se ha suministrado un RemoteConnectionDescriptor para la conexión, el parámetro es ignorado.

La instrucción sólo devolverá un **LocalConnectionDescriptor** si se modifican los parámetros de la conexión local, tales como, por ejemplo, los puertos RTP, etc. De este modo, si sólo se modifica, por ejemplo, el modo de la conexión, no se devolverá un LocalConnectionDescriptor. Si se omite un parámetro de la conexión, como por ejemplo el modo o la supresión de silencio, se retendrá, si es posible, el valor antiguo de ese parámetro. Si se necesita cambiar uno o más parámetros *no especificados*, la pasarela tiene libertad para elegir los valores adecuados de los parámetros no especificados que se deben cambiar¹⁹.

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 agente de llamada²⁰. Cuando se da información de dirección RTP a un cliente incorporado para una conexión, el cliente DEBERÍA aceptar solamente trenes de medios (y RTCP) procedentes también de la dirección RTP especificada. Cualquier tren de medios procedente de otra dirección DEBERÍA ser descartado. Para conocer los requisitos de seguridad adicionales deben consultarse la Especificación UIT-T J.170 de seguridad IPCablecom (en desarrollo).

¹⁹ Esto puede ocurrir, por ejemplo, si se especifica un cambio de códec y el antiguo códec utilizaba supresión de silencio mientras que el nuevo códec no la soporta. Si, por ejemplo, no se ha especificado además el periodo de paquetización, y el nuevo códec soporta el antiguo periodo de paquetización, el valor de este parámetro no cambiaría puesto que tal cambio no sería necesario.

²⁰ Por ejemplo si los medios necesitan atravesar un cortafuegos.

Los parámetros **RequestedEvents**, **RequestIdentifier**, **DigitMap**, **SignalRequests**, **QuarantineHandling** y **DetectEvents** son opcionales. Los parámetros pueden ser utilizados por el agente de llamada para incluir una petición de notificación que esté ligada a la modificación de conexión y sea ejecutada simultáneamente con ésta. Si se suministran uno o más de estos parámetros, el RequestIdentifier DEBE entonces ser uno de ellos. Por ejemplo, cuando se acepta una llamada, la pasarela llamante debe recibir instrucciones para que ponga la conexión en el modo "send/receive" y detenga la provisión de tonos de llamada. Esto se puede llevar a cabo en una sola instrucción ModifyConnection incluyendo una petición de notificación con los parámetros RequestedEvents para el evento colgado y un parámetro SignalRequests vacío para detener la provisión de tonos de llamada de retorno.

Cuando están presentes estos parámetros, la modificación de la conexión y la petición de notificación DEBEN estar sincronizadas, lo que implica que ambas son aceptadas o rechazadas.

ReturnCode es un parámetro devuelto por la pasarela. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

ResourceID es un parámetro D-QoS que es devuelto por la pasarela si ésta realiza una reserva de recurso y obtiene un nuevo ResourceID del encaminador en borde. Cuando se efectúa una reserva de recursos D-QoS, el ResourceID proporciona un título para los recursos reservados.

6.3.5 DeleteConnection (suprimir conexión) (desde el agente de llamada)

Esta instrucción se utiliza para terminar una conexión. Como efecto colateral, recoge datos estadísticos sobre la ejecución de la conexión.

```

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

```

El identificador de punto extremo, en esta forma de la instrucción DeleteConnection, DEBE estar totalmente calificado. NO SE DEBEN utilizar convenios de comodines.

En el caso general de una conexión con dos extremos, esta instrucción ha de ser enviada a las dos pasarelas involucradas en la conexión. Después de que se haya suprimido la conexión, los trenes de medios de la red de paquetes soportados anteriormente por la conexión no estarán disponibles en adelante. Cualquier paquete de medios recibido para la conexión antigua es descartado sencillamente y no se envía ningún paquete de medios nuevo para el tren de medios. Cuando se han realizado una o más reservas y/o compromisos D-QoS para la conexión, la instrucción DeleteConnection liberará los recursos reservados.

En respuesta a la instrucción DeleteConnection, la pasarela devuelve una lista de parámetros que describe la situación de la conexión. Estos parámetros son:

- **Number of packets sent (número de paquetes enviados):** Número total de paquetes de datos RTP enviados por el emisor desde el inicio de la transmisión a través de la conexión. El cómputo no se reinicia si el emisor cambia su identificador de fuente de sincronización (SSRC, *synchronization source identifier*, definido en RTP) –por ejemplo, como resultado de una instrucción (modificar). El valor es cero si, por ejemplo, la conexión ha estado siempre fijada en el modo "receive only".

- **Number of octets sent (número de octetos enviados):** Número total de octetos de cabida útil (es decir, sin incluir el encabezamiento y el relleno) enviados en paquetes de datos RTP por el emisor desde el inicio de la transmisión a través de la conexión. El cómputo no se reinicia si el emisor cambia su identificador de fuente de sincronización (SSRC, definido en RTP) – por ejemplo, como resultado de una instrucción ModifyConnection (modificar conexión). El valor es cero si, por ejemplo, la conexión ha estado siempre fijada en el modo "receive only".
- **Number of packets received (número de paquetes recibidos):** Número total de paquetes de datos RTP recibidos por el emisor desde el inicio de la recepción a través de la conexión. El cómputo incluye paquetes recibidos de diferentes SSRC si el emisor ha utilizado varios valores. El valor es cero si, por ejemplo, la conexión ha estado siempre fijada en el modo "send only".
- **Number of octets received (número de octetos recibidos):** Número total de octetos de cabida útil (es decir, sin incluir el encabezamiento y el relleno) transmitidos en paquetes de datos RTP por el emisor desde el inicio de la transmisión a través de la conexión. El cómputo incluye paquetes recibidos de diferentes SSRC si el emisor ha utilizado varios valores. El valor es cero si, por ejemplo, la conexión ha estado siempre fijada en el modo "send only".
- **Number of packets lost (número de paquetes perdidos):** Número total de paquetes de datos RTP que se han perdido desde el comienzo de la recepción. Este número se define como el número de paquetes esperados menos el número de paquetes realmente recibidos, donde el número de paquetes recibidos incluye los paquetes retrasados y los duplicados. El cómputo incluye paquetes recibidos de diferentes SSRC si el emisor ha utilizado varios valores. Por ello, los paquetes que llegan retrasados no se cuentan como paquetes perdidos, y la pérdida puede ser negativa si los paquetes son duplicados. El cómputo incluye paquetes recibidos de diferentes SSRC si el emisor ha utilizado varios valores. El número de paquetes esperados se define como la diferencia entre el último número secuencial recibido y el número secuencial inicial recibido. El cómputo incluye paquetes recibidos de diferentes SSRC, si el emisor ha utilizado varios valores. El valor es cero si, por ejemplo, la conexión ha estado siempre fijada en el modo "send only".
- **Interarrival jitter (fluctuación entre llegadas):** Estimación de la varianza estadística del tiempo entre llegadas de paquetes de datos RTP medido en milisegundos y expresado como un número entero sin signo. La fluctuación entre llegadas "J" se define como la desviación media (valor absoluto suavizado) de la diferencia "D" de la separación de paquetes en el receptor en comparación con la separación en el emisor para una pareja de paquetes. En RFC 1889 se pueden ver los algoritmos de cálculo detallados al respecto. El cómputo incluye paquetes recibidos de diferentes SSRC si el emisor ha utilizado varios valores. El valor es cero si, por ejemplo, la conexión ha estado siempre fijada en el modo "send only".
- **Average transmission delay (retardo medio de transmisión):** Estimación de la latencia de la red, expresada en milisegundos. Es el valor medio de la diferencia entre la indicación de tiempo NTP de los emisores de los mensajes RTCP y la indicación de tiempo NTP de los receptores, medida cuando se reciben los mensajes. El valor medio se obtiene sumando todas las estimaciones y dividiendo la suma por el número de mensajes RTCP que se han recibido. Debe señalarse que el cálculo correcto de este parámetro depende de que los relojes estén sincronizados. Los dispositivos de los clientes incorporados PUEDEN alternativamente estimar el retardo medio de transmisión dividiendo por dos el tiempo de propagación de ida y retorno.

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

Los parámetros **NotifiedEntity**, **RequestedEvents**, **RequestIdentifier**, **DigitMap**, **SignalRequests**, **QuarantineHandling**, y **DetectEvents** son opcionales. Pueden ser utilizados por el agente de llamada para transmitir una petición de notificación que está ligada a, y es ejecutada simultáneamente con, la supresión de la conexión. Sin embargo, si están presentes uno más de estos parámetros, **RequestIdentifier** DEBE ser uno de ellos. Por ejemplo, cuando un usuario cuelga el teléfono, la pasarela puede recibir instrucciones para que suprima la conexión e inicie la espera de un evento descolgar. Esto se puede realizar también en una sola instrucción **DeleteConnection** mediante la transmisión del parámetro **RequestedEvents** para el evento colgar y un parámetro **SignalRequests** vacío.

Cuando están presentes estos parámetros, la supresión de conexión y la petición de notificación DEBEN estar sincronizadas, lo que implica que ambas serán aceptadas o rechazadas.

ReturnCode es un parámetro devuelto por la pasarela. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

6.3.6 **DeleteConnection (suprimir conexión) (desde el cliente incorporado)**

En determinadas circunstancias, una pasarela puede tener que liberar una conexión debido, por ejemplo, a que ha perdido los recursos asociados con dicha conexión. La pasarela puede terminar la conexión utilizando una variante de la instrucción **DeleteConnection**:

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

El **EndpointId** en esta forma de la instrucción **DeleteConnection**, DEBE estar totalmente calificado. NO SE DEBEN utilizar convenios de comodines.

El **Reason-code** es una cadena de texto que comienza con un código de motivo numérico que va seguido opcionalmente por una cadena de texto descriptiva. En 6.6 se recoge una lista de códigos de motivo.

Además de los **CallId**, **EndpointId** y **ConnectionId** el cliente incorporado deberá enviar también los parámetros de la conexión, los cuales habrían sido devueltos al agente de llamada en respuesta a una instrucción **DeleteConnection** procedente del agente de llamada. El código de motivo indica la causa de la instrucción **DeleteConnection**. Cuando se han realizado una o más reservas y/o compromisos D-QoS para la conexión, el cliente incorporado liberará los recursos reservados.

ReturnCode (código de retorno) es un parámetro devuelto por el agente de llamada. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

6.3.7 **DeleteConnection (suprimir conexión) (suprimir múltiples conexiones desde el agente de llamada)**

El agente de llamada puede utilizar una variación de la función **DeleteConnection** para suprimir varias conexiones al mismo tiempo. La instrucción puede utilizarse para suprimir todas las conexiones relativas a una llamada para un punto extremo:

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

El **EndpointId**, en la forma de la instrucción DeleteConnection, NO DEBE utilizar el comodín "any of" ("cualquiera de"). Se suprimirán todas las conexiones para el punto o puntos extremos con el CallId especificado. La instrucción no devuelve ningún parámetro estadístico o parámetro de llamada individual.

DeleteConnection puede ser también utilizada por el agente de llamada para suprimir todas las conexiones que terminan en un punto extremo determinado:

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

En esta forma de la instrucción DeleteConnection, los agentes de llamada pueden aprovechar la estructura de denominación jerárquica de los puntos extremos para suprimir todas las conexiones que pertenecen a un grupo de puntos extremos. En este caso, se puede especificar parte de la componente "nombre de punto extremo local" del punto extremo utilizando el convenio de comodines "all" ("todos"), como se especifica en 6.1.1. NO SE DEBE utilizar el convenio de comodines "any of" ("cualquiera de"). La instrucción no devuelve ningún parámetro estadístico o parámetro de llamada individual.

Después de que se haya suprimido la conexión, los trenes de medios de la red de paquetes soportados anteriormente por la misma no se encuentran en adelante disponibles. Cualquier paquete de medios recibido para la conexión antigua es sencillamente descartado y no se envía ningún nuevo paquete de medios para el tren. Cuando se han efectuado una o más reservas y/o compromisos D-QoS para la conexión, el cliente incorporado liberará los recursos reservados.

ReturnCode es un parámetro devuelto por la pasarela. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

6.3.8 Auditing (auditoría)

El MGCP se basa en una arquitectura de control de llamada centralizado en la cual el agente de llamada actúa como controlador distante de los dispositivos de cliente que proporcionan interfaces de voz a los usuarios y las redes. Con el fin de alcanzar niveles de disponibilidad iguales o superiores a los actuales de la RTPC, algunos protocolos han implementado mecanismos destinados a "interrogar" periódicamente a los abonados para hacer mínimo el tiempo hasta que se detecta una interrupción individual. Con este objeto, se proporciona un mecanismo de auditoría específico del MGCP entre los clientes incorporados y los agentes de llamada de un sistema IPCablecom para permitir al agente de llamada que realice la auditoría del estado de la conexión y del punto extremo y recupere las capacidades específicas del protocolo de un punto extremo.

Para los clientes incorporados se definen dos instrucciones de auditoría:

- **AuditEndPoint (auditoría de punto extremo):** Utilizada por el agente de llamada para determinar la situación de un punto extremo.
- **AuditConnection (auditoría de conexión):** Utilizada por el agente de llamada para obtener información acerca de una conexión.

Es generalmente deseable que la gestión de la red abarque mas capacidades que las proporcionadas por estas instrucciones, por ejemplo la información relativa a la situación del cliente incorporado en comparación con los puntos extremos individuales. Se espera que tales capacidades sean soportadas por la utilización del protocolo de gestión de red simple (SNMP) y por la definición de una MIB (base de información de gestión) para el cliente incorporado, las cuales caen ambas fuera del alcance de esta Recomendación.

6.3.8.1 AuditEndPoint (auditar punto extremo)

La instrucción AuditEndPoint puede ser utilizada por el agente de llamada para averiguar la situación de un punto extremo determinado.

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

El **EndPointId** identifica el punto extremo que está siendo auditado. NO SE DEBE utilizar el convenio de comodines "cualquiera de" ("any of").

El convenio de comodines "todos de" ("all of") se puede utilizar para auditar un grupo de puntos extremos. Si se utiliza este convenio, la pasarela DEBE devolver la lista de identificadores de puntos extremos que se ajustan al comodín en el parámetro **EndPointIdList**, que es sencillamente una lista de SpecificEndPointId – RequestedInfo NO SE DEBE incluir en este caso. **MaxEndPointID** es un valor numérico que indica el número máximo de EndpointIds que se han de devolver. Si existen puntos extremos adicionales, DEBE estar presente el parámetro de retorno **NumEndPoints** e indicar el número total de puntos extremos que corresponden al EndpointID especificado. A fin de recuperar el siguiente bloque de EndpointID, el **SpecificEndPointID** se fija al valor del último punto extremo devuelto en la anterior EndPointIDList, y la instrucción es expedida.

Cuando no se utiliza el convenio de comodines, el **RequestedInfo** (que puede estar vacío) describe la información solicitada sobre el EndpointId especificado – los parámetros SpecificEndPointID y MaxEndPointID NO SE DEBEN entonces utilizar. Con esta instrucción se puede auditar la siguiente información específica del punto extremo:

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

La respuesta, a su vez, incluirá información acerca de cada uno de los elementos sobre los cuales se ha solicitado información de auditoría:

- **RequestedEvents** – Valor actual de la lista RequestedEvents que está utilizando el punto extremo incluida la acción asociada con cada evento. Los eventos persistentes están incluidos en la lista.
- **DigitMap** – Mapa de dígitos que está utilizando actualmente el punto extremo.

- **SignalRequests** – Lista de las señales de temporización actualmente activas, las señales On/Off que están actualmente "activas" ("on") para el punto extremo (con o sin parámetro) y cualquier señal breve ("brief") pendiente²¹. No se incluyen las señales de temporización que han expirado y las señales breves actualmente en ejecución. Las señales parametrizadas son comunicadas con los parámetros con que han sido aplicadas.
- **RequestIdentifier** – RequestIdentifier (identificador de petición) para la última NotificationRequest recibida por el punto extremo (incluida la petición de notificación incluida en las primitivas de tratamiento de la conexión). Si no se ha recibido ninguna petición de notificación, se devolverá el valor cero.
- **NotifiedEntity** – "Entidad notificada" actual para el punto extremo.
- **ConnectionIdentifiers** – Lista de ConnectionIdentifiers, separados por comas, para todas las conexiones que existen actualmente para el punto extremo especificado.
- **DetectEvents** – Valor actual de la lista DetectEvents que está utilizando el punto extremo. Los eventos persistentes están incluidos en la lista.
- **ObservedEvents** – Lista vigente de los eventos observados para el punto extremo.
- **EventStates** – Para los eventos que tienen estados auditables asociados, el evento que corresponde al estado en que se encuentra el punto extremo, por ejemplo, descolgado en el ejemplo de paquete de línea si el punto extremo está descolgado. La definición de cada evento establecerá si el evento en cuestión tiene un estado auditable asociado con él.
- **VersionSupported** – Lista de versiones de protocolo soportadas por el punto extremo.
- **Capabilities** – Capacidades del punto extremo similares al parámetro LocalConnectionOptions y que incluyen lotes de eventos y modos de conexión. Si se necesita especificar que algunos parámetros, como por ejemplo la supresión de silencio, solamente son compatibles con algunos códecs, la pasarela devolverá varios conjuntos de capacidades. Si un punto extremo es preguntado sobre una capacidad que no se comprende, el punto extremo NO DEBE generar un error; en su lugar el parámetro DEBE omitirse en la respuesta:
- **Compression Algorithm (algoritmo de compresión)** – Lista de códecs soportados. El resto de los parámetros se aplicarán a todos los códecs especificados en esta lista.
- **Packetization Period (periodo de paquetización)** – Se puede especificar un valor único o una gama de valores.
- **Bandwidth (anchura de banda)** – Se puede especificar un valor único o una gama de valores correspondiente a la gama de periodos de paquetización (suponiendo que no hay supresión de silencio).
- **Echo Cancellation (cancelación de eco)** – Especifica si es soportada o no la cancelación de eco.
- **Silence Suppression (supresión de silencio)** – Especifica si es soportada o no la supresión de silencio.
- **Type of Service (tipo de servicio)** – Especifica si es soportado o no el tipo de servicio.
- **Event Packages (lotes de eventos)** – Lista de lotes de eventos soportados. El primer lote de eventos de la lista será el lote por defecto.
- **Modes (modos)** – Lista de modos de conexión soportados.
- **Dynamic Quality of Service (calidad de servicio dinámica)** – Especifica si es soportada o no la calidad de servicio dinámica.

²¹ Normalmente, no debería haber ninguna señal breve pendiente.

- **Security (seguridad)** – Especifica si son soportados o no los servicios de seguridad de IPCablecom. Si son soportados, puede estar presentes también los siguientes parámetros.
- **RTP Ciphersuites** – Lista de algoritmos de autenticación y criptación soportados por RTP.
- **RTCP Ciphersuites** – Lista de algoritmos de autenticación y criptación soportados por RTCP.

El agente de llamada puede entonces decidir si utiliza la instrucción `AuditConnection` para obtener más información acerca de las conexiones.

ReturnCode es un parámetro devuelto por la pasarela. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

Si no se solicita ninguna información y el `EndpointId` es un `EndpointId` válido y completamente especificado, la pasarela devolverá simplemente una respuesta de éxito (código de retorno 200 – transacción ejecutada normalmente).

Debe señalarse que toda la información devuelta es meramente una "instantánea". La recepción de nuevas instrucciones, la actividad local, etc., pueden alterar la mayor parte de las informaciones citadas anteriormente. Por ejemplo, el estado de colgado/descolgado puede cambiar antes de que el agente de llamada reciba la información antes descrita.

6.3.8.2 `AuditConnection` (auditar conexión)

La auditoría de conexiones individuales en un punto extremo se pueden realizar utilizando la instrucción `AuditConnection`.

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

El **EndpointId** identifica el punto extremo que se está auditando – NO SE DEBEN utilizar comodines. El parámetro **RequestedInfo** (que puede estar vacío) contiene la información que se solicita para el **ConnectionId** dentro del `EndpointId` especificado. Con esta instrucción se puede auditar la siguiente información de conexión:

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

La respuesta, a su vez, deberá incluir información acerca de cada uno de los elementos sobre los cuales se pidió la información de auditoría:

- **CallId** – `CallId` de la llamada a la que pertenece la conexión.
- **NotifiedEntity** – "Entidad notificada" actual para el punto extremo.
- **LocalConnectionOptions** – `LocalConnectionOptions` suministrada para la conexión.
- **Mode** – Modo de conexión vigente.
- **ConnectionParameters** – Parámetros de conexión vigentes para la conexión.

- **LocalConnectionDescriptor** – LocalConnectionDescriptor que ha suministrado la pasarela para la conexión.
- **RemoteConnectionDescriptor** – RemoteConnectionDescriptor que ha suministrado la pasarela para la conexión.

ReturnCode parámetro devuelto por la pasarela. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

Si no se ha solicitado ninguna información y el EndpointId se refiere a un punto extremo válido, la pasarela simplemente comprobará que la conexión especificada existe y, en caso afirmativo, devolverá un respuesta positiva (código de retorno 200 – transacción ejecutada).

6.3.9 Restart in Progress (rearranque en curso)

La instrucción RestartInProgress es utilizada por la pasarela para indicar que un punto extremo, o un grupo de puntos extremos, está retirado del servicio o está pasando a la situación de reserva.

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

El **EndpointId** identifica los puntos extremos que entran o se retiran del servicio. Se puede utilizar el convenio de comodines "todos de" ("all of") para aplicar la instrucción a un grupo de puntos extremos, por ejemplo, a todos los puntos extremos que están asociados con una interfaz específica, o incluso a todos los puntos extremos asociados con una pasarela determinada. NO SE DEBE utilizar el convenio de comodines "any of".

El parámetro RestartMethod especifica el tipo de rearranque:

- Un método de rearranque "graceful" ("cuidadoso") indica que el punto, o puntos, extremos especificados se retirarán del servicio después del "retardo de rearranque" ("restart delay") especificado. Las conexiones establecidas no se ven todavía afectadas, pero el agente de llamada deberá abstenerse de establecer nuevas conexiones y debe tratar de liberar con cuidado cualquier conexión existente.
- Un método de rearranque "forced" ("obligado") indica que los puntos extremos especificados son retirados del servicio de forma abrupta. Las conexiones establecidas, si las hay, se pierden.
- Un método de "restart" ("rearranque") indica que el servicio será restaurado en los puntos extremos después de "retardo de rearranque" especificado. No hay ninguna conexión establecida actualmente en los puntos extremos.
- Un método "disconnected" ("desconectado") indica que el punto extremo ha pasado a la situación de desconectado y está ahora intentando establecer la conectividad. El "retardo de rearranque" especifica el número de segundos durante los cuales ha estado desconectado el punto extremo. Las conexiones establecidas no se ven afectadas.

El parámetro opcional "retardo de rearranque" se expresa como un número de segundos. Si este número está ausente, el valor del retardo se debe considerar nulo. En el caso del método "cuidadoso", un retardo nulo indica que el agente de llamada debe sencillamente esperar la terminación natural de las conexiones existentes, sin establecer nuevas conexiones. El "retardo de rearranque" se considera siempre nulo en el caso del método "obligado". Un "retardo de rearranque" nulo para el método de "rearranque" indica que el servicio ya ha sido restablecido. Esto ocurrirá típicamente después del arranque/reinicio de la pasarela. Para mitigar los efectos de un cambio de dirección IP del cliente, el agente de llamada PUEDE desear resolver el nombre de dominio de

cliente incorporado consultando al DNS con independencia del TTL (tiempo de vida) de un registro de recursos vigente para el cliente incorporado restablecido.

Los clientes incorporados DEBERÍAN enviar un mensaje RestartInProgress "cuidadoso" u "obligado" como cortesía con el agente de llamada cuando son puestos fuera de servicio debido, por ejemplo, a una interrupción, o por un sistema de gestión de red, aunque el agente de llamada no puede confiar siempre en que recibirá tales mensajes. Los clientes incorporados DEBEN enviar un mensaje RestartInProgress "rearranque" con un retardo nulo a sus respectivos agentes de llamada cuando son puestos de nuevo en servicio de conformidad con el procedimiento especificado en 6.4.3.5 – Los agentes de llamada pueden confiar en recibir este mensaje. Asimismo, los clientes incorporados DEBEN enviar un mensaje RestartInProgress "desconectado" a su respectiva "entidad notificada" actual de conformidad con el procedimiento de "desconectado" especificado en 6.4.3.6. El parámetro "retardo de rearranque" NO SE DEBE utilizar con el método de rearranque "obligado".

El mensaje RestartInProgress se enviará a la "entidad notificada" actual para el EndpointId en cuestión. Se espera que se haya proporcionado un agente de llamada por defecto, es decir, "entidad notificada", para cada punto extremo, de modo que después de un reinicio el agente de llamada por defecto será la "entidad notificada" para cada punto extremo. Los clientes incorporados DEBEN aprovechar totalmente la utilización de comodines para hacer mínimo el número de mensajes RestartInProgress generados cuando rearrancan múltiples puntos extremos en un pasarela y los puntos extremos son gestionados por el mismo agente de llamada.

ReturnCode es un parámetro devuelto por el agente de llamada. Indica el resultado de la instrucción y está formado por un número entero (véase 6.5) seguido opcionalmente de un comentario.

Puede devolverse además una **NotifiedEntity** con la respuesta del agente de llamada:

- Si la respuesta indica éxito (código de retorno 200 – transacción ejecutada), el procedimiento de rearranque se ha completado y la NotifiedEntity devuelta es la nueva "entidad notificada" para el punto o puntos extremos.
- Si la respuesta del agente de llamada indica un error, el procedimiento de rearranque no está todavía completado, y debe por consiguiente iniciarse de nuevo. Si se ha devuelto un parámetro NotifiedEntity, éste especifica entonces la nueva "entidad notificada" para el punto o puntos extremos, la cual debe en consecuencia utilizarse cuando se reintenta el procedimiento de rearranque.

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

6.4 Estados, condiciones de cambio-por-fallo y condiciones de carrera

Para implementar la señalización de llamada apropiada, el agente de llamada debe hacer un seguimiento del estado del punto extremo, y la pasarela debe garantizar que los eventos son notificados adecuadamente al agente de llamada. Se pueden presentar condiciones especiales cuando rearrancan la pasarela o el agente de llamada: la pasarela puede tener necesidad de ser redirigida a un nuevo agente de llamada durante los procedimientos de cambio-por-fallo. De manera análoga, el agente de llamada puede tener necesidad de emprender acciones especiales cuando la pasarela es retirada del servicio, o rearrancada.

6.4.1 Recapitulaciones y puntos notables

Como se ha mencionado en la cláusula 6.1.4, los agentes de llamada se identifican mediante su nombre de dominio, y cada punto extremo tiene una, y sólo una, "entidad notificada" asociada con él en cualquier momento determinado. En esta cláusula recapitulamos y subrayamos las áreas que tienen una importancia especial en relación con la fiabilidad y el cambio-por-fallo en el MGCP:

- Un agente de llamada se identifica por su nombre de dominio, no por sus direcciones de red, y se pueden asociar varias direcciones de red con un nombre de dominio.

- Un punto extremo tiene uno, y sólo uno, agente de llamada asociado con él en cualquier momento determinado. El agente de llamada asociado con el punto extremo es el valor vigente de la "entidad notificada".
- La "entidad notificada" se fija inicialmente a un valor suministrado. Cuando se reciben instrucciones con un parámetro NotifiedEntity para el punto extremo, incluidos nombres de punto extremo con comodín, la "entidad notificada" se fija al valor especificado. Si la "entidad notificada" para un punto extremo está vacía o no ha sido fijada explícitamente²², la "entidad notificada" no se presenta a la dirección de fuente de la última instrucción de tratamiento de la conexión o petición de notificación recibida para el punto extremo. En este caso, el agente de llamada, será por tanto identificado por su dirección de red, lo que DEBE efectuarse solamente en casos excepcionales.
- Las respuestas a instrucciones se envían siempre a la dirección de fuente de la instrucción, con independencia de la "entidad notificada" actual. Cuando se necesita transportar con la respuesta un mensaje Notify, el datagrama es todavía enviado a la dirección de fuente de la nueva instrucción recibida, con independencia de la NotifiedEntity para cualquiera de las instrucciones.
- Cuando la "entidad notificada" se refiere a un nombre de dominio que se resuelve convirtiéndose en múltiples direcciones IP, los puntos extremos tienen capacidad para conmutar entre cada una de estas direcciones, si bien no pueden no pueden cambiar la "entidad notificada" a otro nombre de dominio por sí mismos. Un agente de llamada puede sin embargo darles instrucciones para que efectúen la conmutación proporcionándoles una nueva "entidad notificada".
- Si un agente de llamada pasa a estar indisponible, los puntos extremos gestionados por el mismo pasarán finalmente a la situación de "desconectados". El único medio de que estos puntos extremos se conecten de nuevo es que el agente de llamada que ha sufrido fallo esté de nuevo disponible, o que otro agente de llamada (reserva) ponga en contacto los puntos extremos afectados con una nueva "entidad notificada".
- Cuando otro agente de llamada (reserva) se ha hecho cargo del control de un grupo de puntos extremos, se supone que el agente de llamada que ha sufrido fallo comunicará y sincronizará con el agente de llamada de reserva para transferir el control de los puntos extremos afectados al agente de llamada original, si así se desea. Alternativamente, el agente de llamada que ha sufrido fallo podría sencillamente pasar a ser ahora el agente de llamada de reserva.

Debemos señalar que no se proporciona la resolución de conflictos de transferencia entre agentes de llamada – confiamos totalmente en que los agentes de llamada conocen lo que hacen y lo que se comunican entre sí (aunque se puede utilizar la AuditEndpoint para conocer la "entidad notificada" actual).

6.4.2 Retransmisión y detección de asociaciones perdidas

El protocolo MGCP se organiza como un conjunto de transacciones, cada una de ellas formada por una instrucción y una respuesta. Los mensajes MGCP, cursados por el UDP, pueden estar sujetos a pérdidas. En ausencia de una respuesta oportuna (véase 7.5), las instrucciones se repiten. Las pasarelas DEBEN guardar en memoria una lista de las respuestas que han enviado a las transacciones recientes y una lista de las transacciones que se están ejecutando actualmente. El adjetivo reciente se define aquí por el valor $T_{t_{hist}}$ que especifica el número de segundos que deben ser guardadas en memoria las respuestas a las transacciones antiguas. El valor por defecto de $T_{t_{hist}}$ es 30 segundos.

²² Esto puede ocurrir, por ejemplo, debido a la especificación de un parámetro NotifiedEntity vacío.

Los identificadores de transacción de las instrucciones entrantes se comparan en primer lugar con los identificadores de transacción de las respuestas recientes. Si existe concordancia, la pasarela no ejecuta la transacción, sino que sencillamente repite la respuesta antigua. Si no se encuentra concordancia con una respuesta anterior a una transacción, el identificador de transacción de la instrucción entrante se compara con la lista de transacciones que aún no se han terminado de ejecutar. Si se encuentra concordancia la pasarela no ejecuta la transacción, que es sencillamente ignorada –se proporcionará una respuesta cuando se complete la ejecución de la instrucción.

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

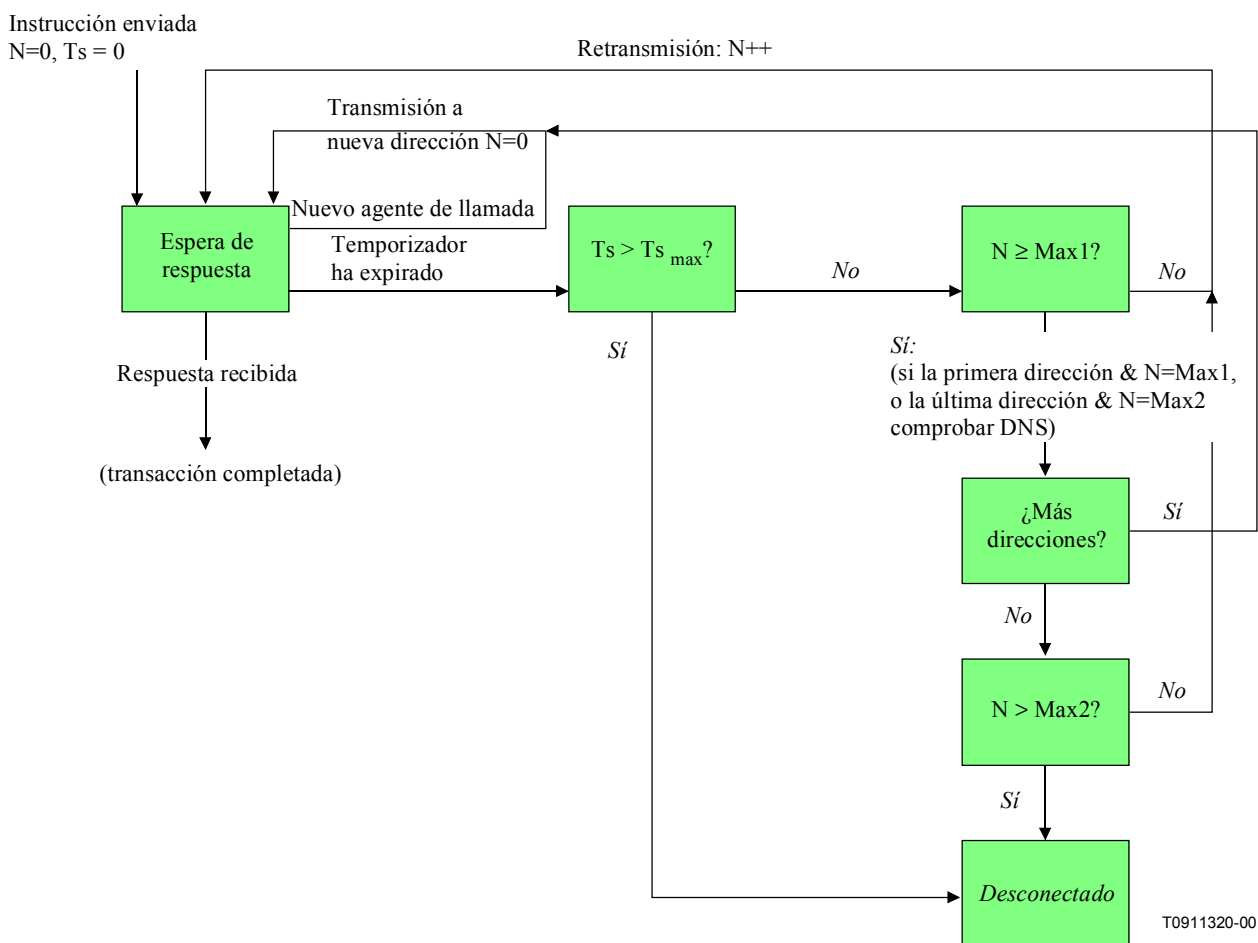
- errores de transmisión, como cuando, por ejemplo, se pierde un paquete debido al ruido en una línea o la congestión en una cola,
- fallo de un componente, como cuando, por ejemplo, una interfaz para un agente de llamada pasa a estar indisponible,
- fallo del agente de llamada, como cuando, por ejemplo, todas las interfaces de un agente de llamada pasan a estar indisponibles,
- cambio-por-fallo, cuando un nuevo agente de llamada se "hace cargo" transparentemente.

Los elementos deben ser capaces de obtener, a partir de la historia pasada, una estimación de la tasa de pérdida de paquetes. En un sistema configurado adecuadamente, esta tasa de pérdida debe ser muy baja, típicamente inferior al 1% como valor medio. Si un agente de llamada o una pasarela ha de repetir un mensaje varias veces, se puede suponer legítimamente que se está produciendo algo más que un error de transmisión. Por ejemplo, dada una tasa de pérdida de 1% uniformemente distribuida, la probabilidad de que fallen 5 tentativas de transmisión consecutivas es de 1 en 100 000 millones, lo que significa que un evento de fallo debe ocurrir menos de una vez cada 10 días para un agente de llamada que procese 1000 transacciones por segundo. (En efecto, el número de repeticiones que se considera excesivo debe ser una función de la tasa de pérdida de paquetes predominante.) Cuando los errores no están uniformemente distribuidos, la probabilidad de fallos consecutivos puede hacerse algo superior. Debemos señalar que el "umbral de desconfianza" ("suspicion threshold"), que llamaremos "Max1", es normalmente menor que el "umbral de desconexión", que denominaremos "Max2", y que debe fijarse a un valor más grande.

Un algoritmo de retransmisión clásico contaría sencillamente el número de repeticiones sucesivas y concluiría que la asociación está interrumpida después de que el paquete ha sido retransmitido un número excesivo de veces (típicamente entre 7 y 11 veces). Para tener en cuenta la posibilidad de que se produzca un cambio-por-fallo en curso o no detectado, modificamos el algoritmo clásico como sigue:

- La pasarela DEBE comprobar siempre la presencia de un nuevo agente de llamada. Esto puede ser señalado por:
 - la recepción de una instrucción donde NotifiedEntity apunte a un nuevo agente de llamada; o
 - la recepción de una respuesta de redirección que apunte a un nuevo agente de llamada.
- Si se detecta un nuevo agente de llamada, la pasarela DEBE ordenar que las retransmisiones de cualquier instrucción excepcional para los puntos extremos sea redirigida a tal agente de llamada nuevo. Las respuestas a las instrucciones nuevas o antiguas se envían sin embargo a la dirección de fuente de la instrucción.
- Antes de cualquier retransmisión, se comprueba que el tiempo transcurrido desde la emisión del datagrama inicial no excede de $T_{s_{max}}$. Si es superior a $T_{s_{max}}$, el punto extremo se desconecta.

- Si el número de retransmisiones a este agente de llamada es igual a "Max1", la pasarela PUEDE interrogar al servidor de nombres a fin de detectar el posible cambio de las interfaces de agente de llamada, independientemente del tiempo de vida (TTL, *time to live*) asociado con el registro DNS.
- La pasarela puede haber aprendido varias direcciones IP para el agente de llamada. Si el número de retransmisiones para esta dirección IP es superior a "Max1" e inferior a "Max2", y hay más direcciones IP que no han sido probadas, la pasarela DEBE dirigir las retransmisiones a las direcciones alternativas restantes de su lista local.
- Si no quedan más interfaces por probar, y el número de retransmisiones es Max2, la pasarela DEBERÍA entonces contactar una vez más con el DNS para ver si ha quedado disponible alguna otra interfaz. En caso negativo, los puntos extremos gestionados por este agente de llamada se desconectan ahora. Cuando un punto extremo se desconecta, DEBE entonces iniciar el procedimiento "desconectado" que se especifica en 6.4.3.6.



Para adaptarse automáticamente a la carga de la red, el MGCP especifica temporizadores de valores crecientes exponencialmente (véase 7.5.2). Si la temporización inicial se fija a 200 milisegundos, la pérdida de una quinta retransmisión será detectada después de 6 segundos aproximadamente. Esta cifra es probablemente un plazo de espera aceptable para detectar un cambio-por-fallo. Las retransmisiones deben continuar después de este plazo, no sólo para superar un problema de conectividad transitorio, sino también para conceder algún tiempo más para la ejecución de un cambio-por-fallo – una espera de un retardo total de 30 segundos es probablemente aceptable.

Debe señalarse que hay una relación estrecha entre $T_{s_{max}}$, T_{hist} y el tiempo de tránsito máximo, $T_{p_{max}}$. Específicamente, DEBE cumplirse la siguiente relación para impedir que las instrucciones retransmitidas sean ejecutadas más de una vez:

$$T_{\text{hist}} \geq T_{\text{Smax}} + T_{\text{Pmax}}$$

El valor por defecto de T_{Smax} es 20 segundos. Con ello, si se supone que el retardo de propagación máximo es de 10 segundos, las respuestas a las transacciones antiguas deben mantenerse durante un periodo de 30 segundos como mínimo. La importancia de que el emisor y el receptor se ajusten a estos valores no puede ser exagerada.

El valor por defecto de Max1 es de 5 retransmisiones y el valor por defecto de Max2 es de 7 retransmisiones. Estos dos valores pueden ser alterados por el proceso de aprovisionamiento.

Además, el proceso de aprovisionamiento DEBE poder desactivar una o ambas consultas DNS Max1 y Max2.

6.4.3 Condiciones de carrera

En esta cláusula describimos cómo el MGCP trata con las condiciones de carrera (condiciones de pugna por llegar primero).

En primer lugar, el MGCP trata con las condiciones de carrera a través de la noción de "lista de cuarentena" ("quarantine list"), que pone en situación de cuarentena los eventos, y a través de la detección explícita de la falta de sincronización, por ejemplo, en caso de estado de colgado-descolgado discordante debido a que se da la condición de doble toma de un punto extremo.

En segundo lugar, el MGCP no supone que el mecanismo de transporte mantendrá el orden de las instrucciones y las respuestas. Esto puede dar lugar a condiciones de carrera que pueden ser obviadas mediante un comportamiento apropiado del agente de llamada que ordene convenientemente las instrucciones.

Finalmente, en algunos casos, muchas pasarelas pueden decidir reiniciar a la vez el funcionamiento. Esto puede ocurrir, por ejemplo, si un área pierde potencia o capacidad de transmisión durante un terremoto o una tormenta de hielo. Cuando se restablecen la potencia y la capacidad de transmisión, muchas pasarelas pueden decidir enviar simultáneamente instrucciones RestartInProgress, lo que podría llevar a un funcionamiento muy inestable si no se controla cuidadosamente.

6.4.3.1 Lista de cuarentena (Quarantine list)

Las pasarelas controladas por MGCP recibirán peticiones de notificación en las que se les pida esperar una lista de eventos. Los elementos de protocolo que determinan el tratamiento de estos eventos son la lista de eventos solicitados ("Requested Events"), el mapa de dígitos ("Digit Map") y la lista de eventos detectados ("Detect Events").

Cuando se inicializa el punto extremo, la lista de eventos solicitados contiene solamente eventos persistentes para el punto extremo, y el mapa de dígitos está vacío. Después de la recepción de una instrucción, la pasarela arranca observando el punto extremo en cuanto a las ocurrencias de los eventos mencionados en la lista, incluidos los eventos persistentes.

Los eventos se examinan a medida que ocurren. La acción que sigue es determinada por el parámetro "action" (acción) asociado con el evento en la lista de eventos solicitados, y también por el mapa de dígitos. Los eventos definidos como "accumulate" o "accumulate according to digit map" se acumulan en la lista de eventos observados. Los eventos señalados como "accumulate according to the digit map" se acumularán además en la "cadena de marcación vigente" ("current dial string"). Esta actuación continuará hasta que se encuentre un evento que dispare una instrucción Notify que se enviará a la "entidad notificada".

La pasarela, en este punto, transmitirá la instrucción Notify y pondrá al punto extremo en un "estado notificación". Mientras el punto extremo se encuentre en este "estado notificación", los eventos detectados en el mismo son almacenados en un memoria intermedia de cuarentena para su posterior procesamiento. Los eventos son, en este sentido, "puestos en cuarentena". Los eventos detectados

son los eventos especificados por la unión del parámetro RequestedEvents y el parámetro DetectEvents recibido más recientemente o, en caso de que no se haya recibido ningún parámetro DetectEvents, los eventos a que se hace referencia en el parámetro RequestedEvents. Se detectan también los eventos persistentes.

El punto extremo sale del "estado notificación" cuando se recibe la respuesta a la instrucción Notify²³. La instrucción Notify se puede retransmitir en el "estado notificación", como se especifica en 6.4.2.

Cuando el punto extremo sale del "estado notificación" se repone la lista de eventos observados y la "cadena de marcación vigente" del punto extremo a un valor nulo.

El perfil NCS ordena la utilización del "modo lockstep", que implica que la pasarela DEBE recibir una nueva instrucción NotificationRequest después de que haya enviado una instrucción Notify. Hasta que esto suceda, el punto extremo se encuentra en un "estado lockstep", y los eventos que ocurran y deban ser detectados son simplemente almacenados en la memoria intermedia de cuarentena. Los eventos que se han de almacenar en cuarentena son los mismos que en el "estado notificación". Una vez que se ha recibido y ejecutado con éxito la nueva NotificationRequest, el punto extremo abandona el "estado lockstep".

Una pasarela puede recibir en cualquier momento una nueva instrucción NotificationRequest para el punto extremo, lo que surtirá también el efecto de sacar al punto extremo del "estado notificación" suponiendo que la NotificationRequest se ejecuta con éxito.

Cuando se recibe una nueva NotificationRequest en el "estado notificación", la pasarela garantizará que la Notify pendiente será recibida por el agente de llamada antes de una respuesta exitosa a la nueva NotificationRequest. Esto se realiza utilizando la funcionalidad de "transporte en remolque" del protocolo y colocando los mensajes (instrucciones y respuestas) que se han de enviar ordenados de modo que el mensaje más antiguo sea el primero. Los mensajes se envían a continuación en un solo paquete a la fuente de la nueva NotificationRequest, independientemente de la fuente y de la "entidad notificada" para la instrucción antigua y nueva. Los pasos involucrados son los siguientes:

- 1) La pasarela prepara un mensaje que cursa en un solo paquete una repetición de la antigua instrucción Notify excepcional y la respuesta a la nueva instrucción NotificationRequest.
- 2) El punto extremo es retirado a continuación del "estado notificación" sin esperar a la respuesta a la instrucción Notify.
- 3) Se conserva una copia de la instrucción Notify excepcional hasta que se reciba una respuesta. Si expira un plazo estipulado, se repetirá la instrucción Notify en un paquete que cursará también una repetición de la respuesta a la NotificationRequest.
 - Si se pierde el paquete que transporta la respuesta a la NotificationRequest, el agente de llamada retransmitirá la NotificationRequest. La pasarela responderá a esta repetición retransmitiendo en un solo paquete la instrucción Notify excepcional y la respuesta a la NotificationRequest – este datagrama se enviará a la fuente de la NotificationRequest.
 - Si la pasarela tiene que transmitir una nueva Notify antes de que se reciba una respuesta a la Notify anterior, prepara un paquete que transporte una repetición de la Notify antigua, una repetición de la respuesta a la última NotificationRequest y la nueva Notify – este datagrama será enviado a la "entidad notificada" vigente.

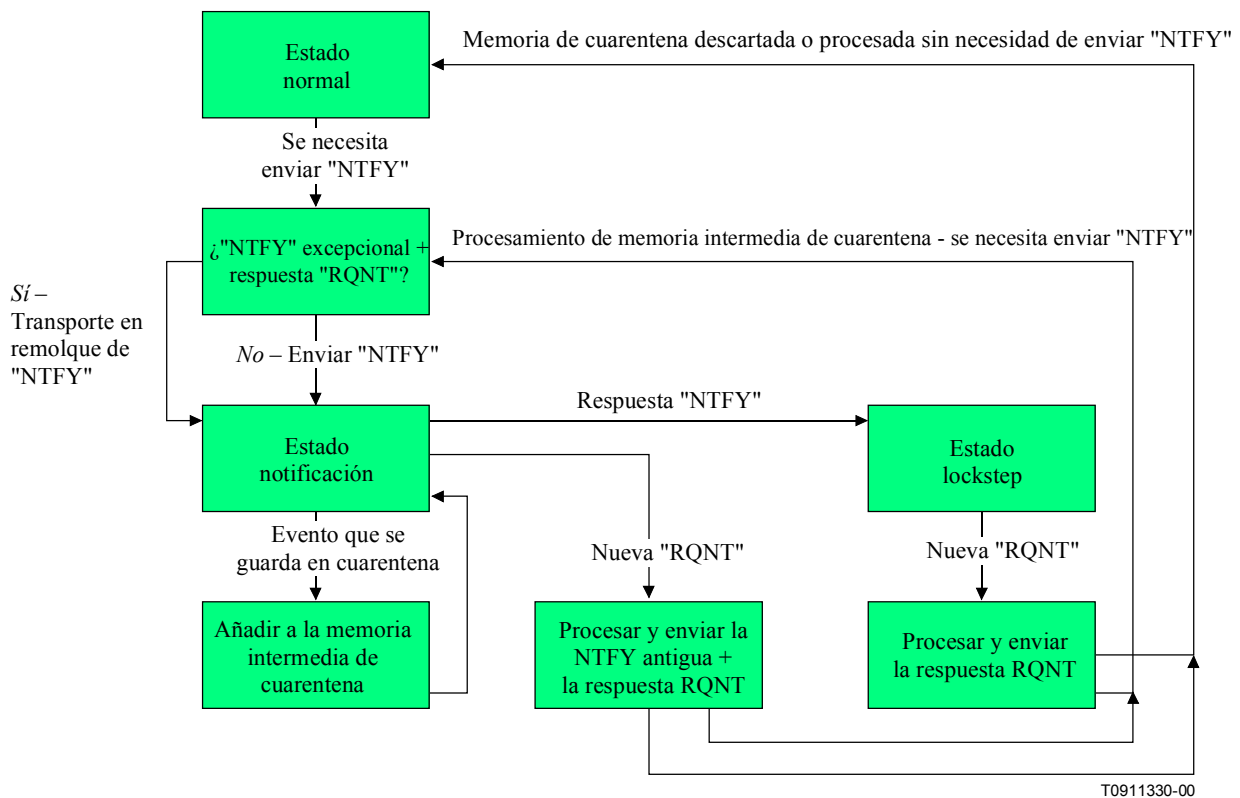
Después de recibir una instrucción NotificationRequest, la lista de "eventos solicitados" y el mapa de dígitos (si se ha proporcionado uno nuevo) son sustituidos por los parámetros recibidos más recientes, y la lista de "eventos observados" y la "cadena de marcación vigente" se reponen a un valor nulo. El comportamiento posterior estará entonces condicionado por el parámetro QuarantineHandling. El parámetro puede especificar que los eventos en cuarentena han de ser

²³ Debe señalarse que la acción Notify no se puede combinar con una NotificationRequest incorporada.

descartados, en cuyo caso se descartarán todos los eventos en cuarentena. Si el parámetro especifica que se deben procesar los eventos en cuarentena, la pasarela iniciará el procesamiento de la lista de eventos en cuarentena, utilizando la lista recientemente recibida de "eventos solicitados" y el "mapa de dígitos" si ha sido proporcionado. Al procesar estos eventos, la pasarela puede encontrar un evento que desencadene el envío de una instrucción Notify. Si éste es el caso, la pasarela transmitirá inmediatamente una instrucción Notify que comunicará todos los eventos que estaban acumulados en la lista de "eventos observados" incluido el evento desencadenante, dejando los eventos no procesados en la memoria de cuarentena. El punto extremo entra de nuevo a continuación en el "estado notificación".

El procedimiento anterior se aplica a todas las formas de peticiones de notificación, con independencia de si ellas forman o no parte de una instrucción de tratamiento de la conexión o de si son proporcionadas como una instrucción NotificationRequest. Las instrucciones de tratamiento de la conexión que no incluyen una petición de notificación, ni son afectadas por el procedimiento anterior ni repercuten en el mismo.

El diagrama a continuación ilustra el procedimiento especificado anteriormente suponiendo que todas las transacciones se ejecutan con éxito:



Los agentes de llamada DEBERÍAN suministrar la respuesta a un mensaje Notify con éxito y la nueva NotificationRequest en el mismo datagrama utilizando el mecanismo de transporte en remolque²⁴.

6.4.3.2 Detección explícita

Un elemento clave relativo al estado de varios puntos extremos es la posición colgado-descolgado. Se pueden presentar condiciones de carrera (condiciones de pugna por llegar primero) y discordancia sobre el estado cuando, por ejemplo, el usuario decide descolgar el teléfono mientras que el agente

²⁴ Los vendedores que decidan no seguir esta Recomendación deben examinar cuidadosamente los escenarios de fallo del agente de llamada.

de llamada está solicitando a la pasarela que espere eventos de descolgado y posiblemente aplique una señal de timbre (la condición de "doble toma" muy conocida en las capacidades basadas en la voz).

Para evitar que se produzca esta situación de carrera, la pasarela DEBE comprobar la condición del punto extremo antes de responder a una NotificationRequest. En particular, DEBE devolver un error:

- 1) Si se solicita a la pasarela que notifique una transición a "descolgado" cuando el teléfono está ya descolgado (código de error 401 – teléfono descolgado),
- 2) Si se solicita a la pasarela que notifique una condición de colgado o de "colgado instantáneo" ("flash hook") cuando el teléfono está ya colgado (código de error 402 – teléfono colgado).

Adicionalmente, la definición de cada una de las señales puede especificar que una señal operará solamente en determinadas condiciones, por ejemplo, la señal de respuesta solamente se producirá si el teléfono ya está descolgado. Si existen tales requisitos previos para una señal dada, la pasarela DEBE devolver el error especificado en la definición de la señal en el caso de que estos prerequisites no se cumplan.

Hay que señalar que la comprobación de la condición se efectúa al mismo tiempo que se recibe la petición de notificación, siendo así que el evento real que ha causado la condición vigente puede haber sido comunicado, o ignorado anteriormente, o puede encontrarse actualmente en cuarentena.

Las otras variables de estado de la pasarela, tales como la lista de eventos solicitados o lista de señales solicitadas, son sustituidas enteramente después de cada NotificationRequest con éxito, lo que evita cualquier discrepancia a largo plazo entre el agente de llamada y la pasarela.

Cuando la NotificationRequest no tiene éxito, esté o no incluida en una instrucción de tratamiento de la conexión, la pasarela continuará sencillamente como si la instrucción nunca se hubiera recibido, aunque se devolverá un error. Al igual que las demás transacciones, la NotificationRequest DEBE operar como una transacción atómica. Por ello, ningún cambio iniciado como resultado de la instrucción DEBE ser deshecho.

Se puede presentar otra condición de carrera cuando se envía brevemente una instrucción Notify antes de la recepción por la pasarela de una NotificationRequest. El RequestIdentifier se utiliza para relacionar las instrucciones Notify con las instrucciones NotificationRequest, con lo cual se capacita al agente de llamada para que determine si la instrucción Notify fue generada antes o después de que la pasarela haya recibido la nueva NotificationRequest.

6.4.3.3 Semántica transaccional

A medida que aumentan los tiempos potenciales de compleción de las transacciones, debido, por ejemplo, a las reservas de recursos externos, se hace cada vez más importante la definición cuidadosa de la semántica transaccional. En particular, el tema de las condiciones de carrera, específicamente en lo que se refiere al estado de colgado-descolgado debe definirse precisamente.

Es importante considerar que el estado colgado-descolgado puede de hecho cambiar entre el momento en que se inicia una transacción y el momento en que se completa. De manera más general, podemos decir que la compleción exitosa de una transacción depende de una o más precondiciones, las cuales pueden cambiar durante la ejecución de la transacción.

La semántica más sencilla para este fin consiste simplemente en exigir que todas las precondiciones SE DEBEN cumplir desde el momento en que se inicia la transacción hasta el momento en que se completa. De este modo, si se modifica alguna de las precondiciones durante la ejecución de la transacción, la transacción DEBE fallar. Además, tan pronto como se inicia la transacción todos los nuevos eventos son puestos en cuarentena. Cuando se conoce el resultado de la transacción, todos los eventos en cuarentena son entonces procesados.

A título de ejemplo, consideremos una transacción que incluye una petición del evento "descolgar". Cuando se inicia la transacción el teléfono está "colgado" y se cumple por tanto esta precondition. Si el estado del gancho conmutador cambia a "descolgado" antes de que se complete la transacción, la precondition no se satisface por más tiempo, y la transacción falla por tanto inmediatamente. El evento "descolgado" deberá ser ahora almacenado en la memoria de cuarentena y será después procesado.

6.4.3.4 Ordenación de las instrucciones y tratamiento de la falta de ordenación

El MGCP no ordena que el protocolo de transporte subyacente garantice la secuenciación de las instrucciones enviadas a una pasarela o a un punto extremo. Esta propiedad tiende a maximizar la oportunidad de las acciones pero presenta algunos inconvenientes. Por ejemplo:

- Las instrucciones Notify pueden retrasarse y llegar al agente de llamada después de la transmisión de una nueva instrucción NotificationRequest.
- Si se transmite una nueva NotificationRequest antes de que se reciba la respuesta a una petición anterior, no está garantizado que la respuesta a esta petición anterior no se reciba en segunda posición.

Los agentes de llamada y las pasarelas que deseen garantizar el funcionamiento coherente de los puntos extremos pueden utilizar las reglas especificadas siguientes:

- 1) Cuando una pasarela se ocupa de varios puntos extremos, las instrucciones pertenecientes a los 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 camino.
- 2) Cuando se crean varias conexiones en el mismo punto extremo, las instrucciones pertenecientes a diferentes conexiones se pueden enviar en paralelo.
- 3) En una conexión concreta, debería haber normalmente una sola instrucción excepcional (crear o modificar). No obstante, la instrucción DeleteConnection puede enviarse en cualquier momento. En consecuencia, una pasarela puede a veces recibir una instrucción ModifyConnection que se aplica a una conexión anteriormente suprimida. Tales instrucciones DEBEN ignorarse, y se devolverá un error (código de error 515 – Id de conexión incorrecto).
- 4) En un punto extremo determinado, debería haber normalmente una sola instrucción NotificationRequest excepcional en un momento dado. El parámetro RequestId se utiliza para relacionar las instrucciones Notify con la NotificationRequest desencadenante.
- 5) En algunos casos, una instrucción DeleteConnection confeccionada implícita o explícitamente con comodines que se aplica a un grupo de puntos extremos puede enfrentarse a una instrucción CreateConnection pendiente. El agente de llamada debe eliminar una a una todas las conexiones cuya completación estaba pendiente en el momento en que se produjo la instrucción DeleteConnection global. Asimismo, no se deberían enviar nuevas instrucciones CreateConnection para puntos extremos denominados por los comodines hasta que se reciba una respuesta a la instrucción DeleteConnection confeccionada con comodines.
- 6) Cuando unas instrucciones están incorporadas en otras, DEBEN adherirse a los requisitos de secuenciación para el conjunto de todas las instrucciones. Por ejemplo, una instrucción CreateConnection con una petición de notificación incorporada debe adherirse a los requisitos de secuenciación para CreateConnection y NotificationRequest al mismo tiempo.
- 7) AuditEndpoint y AuditConnection no están sujetas a ninguna secuenciación.

- 8) RestartInProgress debe ser siempre la primera instrucción enviada por un punto extremo, tal como se define en procedimiento de re arranque (véase 6.4.3.5). Cualquier otra instrucción o respuesta debe ser entregada después de esta instrucción RestartInProgress (se permite el transporte en remolque).
- 9) Cuando se transportan múltiples mensajes en un solo paquete, los mensajes se procesarán siempre por orden.

Aquellas de las reglas anteriores que especifican el comportamiento de las pasarelas DEBEN gozar de la adhesión de los clientes incorporados, si bien estos clientes incorporados NO DEBEN hacer ninguna suposición acerca de si los agentes de llamada siguen las reglas o no. En consecuencia, las pasarelas DEBEN siempre responder a las instrucciones, sin tener en cuenta si ellas se adhieren o no a las reglas señaladas anteriormente.

6.4.3.5 Contienda en una avalancha de re arranques

Supongamos que se activan simultáneamente un número elevado de pasarelas. Si todas ellas han iniciado una transacción RestartInProgress, el agente de llamada se vería probablemente inundado, lo que podría conducir a pérdidas de mensajes y congestión de la red durante el periodo crítico de restauración del servicio. Para evitar tales avalanchas, SE DEBEN seguir las siguientes directrices:

- 1) Cuando se activa una pasarela, ésta inicia un temporizador de re arranque con un valor aleatorio uniformemente distribuido entre 0 y un retardo de espera máximo (MWD, *maximum waiting delay*) provisionable, por ejemplo 360 milisegundos (véase más adelante). SE DEBE tener cuidado de evitar la sincronización de la generación de números aleatorios entre múltiples pasarelas que utilicen el mismo algoritmo.
- 2) La pasarela espera entonces la finalización de este temporizador, la recepción de una instrucción del agente de llamada o la detección de una actividad de usuario local, como por ejemplo, una transición a descolgado en una pasarela residencial. La condición de descolgado preexistente da como resultado la generación de un evento de descolgado.
- 3) Cuando expira el temporizador de re arranque, se recibe una instrucción o se detecta una condición de descolgado preexistente, la pasarela inicia el procedimiento de re arranque.

El procedimiento de re arranque establece sencillamente que el punto extremo DEBE enviar una instrucción RestartInProgress al agente de llamada informándole acerca del re arranque y garantizar además que el primer mensaje (instrucción o respuesta) que vea el agente de llamada procedente de este punto extremo sea esta instrucción RestartInProgress. El punto extremo DEBE aprovechar enteramente el transporte en remolque para realizar esta operación. Por ejemplo, si tiene lugar una actividad de descolgar antes de que expire el temporizador de re arranque, se generará un paquete conteniendo la instrucción RestartInProgress con una instrucción Notify transportada en remolque para el evento descolgar. En el caso de que expire el temporizador de re arranque sin ninguna otra actividad, la pasarela envía sencillamente un mensaje RestartInProgress.

Si la pasarela entra en el estado "desconectado" mientras transporta el procedimiento de re arranque, SE DEBE cursar el procedimiento desconectado especificado en 6.4.3.6, salvo que durante el procedimiento se envíe un mensaje de "re arranque" ("restart") en lugar de un mensaje de "desconectado" ("disconnected").

Se espera que cada punto extremo en una pasarela tenga un agente de llamada suministrable, por ejemplo, una "entidad notificada", hacia el cual dirigir el mensaje de re arranque inicial. Cuando el conjunto de puntos extremos en una pasarela es gestionado por más de un agente de llamada, se debe ejecutar el procedimiento anterior para cada conjunto de puntos extremos gestionado por un agente de llamada determinado. La pasarela DEBE aprovechar completamente la aplicación de comodines para hacer mínimo el número de mensajes RestartInProgress generados cuando re arrancan múltiples puntos extremos en una pasarela y estos puntos extremos son gestionados por el mismo agente de llamada.

El valor del retardo MWD es un parámetro de la configuración que depende del tipo de la pasarela. Para determinar el valor de este parámetro en pasarelas residenciales se puede utilizar el siguiente razonamiento.

Los agentes de llamada se dimensionan típicamente para gestionar el tráfico en la hora cargada, durante la cual se ocupará el 10% de las líneas como valor medio y la duración media de las llamadas será típicamente de 3 minutos. El procesamiento de una llamada requiere generalmente la ejecución de 5 ó 6 transacciones entre cada punto extremo y el agente de llamada. Según este sencillo cálculo se espera que el agente de llamada trate de 5 a 6 transacciones para cada punto extremo, cada 30 minutos por término medio, o, dicho de otro modo, una transacción aproximadamente cada 5 ó 6 minutos como media. Esto sugiere que un valor razonable del MWD para una pasarela residencial estaría comprendido entre 10 y 12 minutos. Si no existe una configuración explícita, los clientes incorporados DEBEN utilizar un valor por defecto de 600 segundos para el MWD.

6.4.3.6 Puntos extremos desconectados

Además del procedimiento de re arranque, los clientes incorporados tienen también un procedimiento "desconectado", el cual se inicia cuando el punto extremo pasa a "desconectado" como se describe en 6.4.2. Debe señalarse aquí que los puntos extremos solamente se pueden desconectar cuando intentan comunicar con el agente de llamada. Un punto extremo que pasa a "desconectado" sigue los pasos a continuación:

- 1) Se inicializa un temporizador de "desconectado" a un valor aleatorio uniformemente distribuido entre 0 y un retardo "de espera" inicial ($T_{d_{init}}$, *initial waiting delay*) provisionable, por ejemplo, de 15 segundos. SE DEBE tener cuidado de evitar la sincronización de la generación de números aleatorios entre múltiples pasarelas y puntos extremos que utilicen el mismo algoritmo.
- 2) La pasarela espera entonces el final de este temporizador, la recepción de una instrucción procedente del agente de llamada o la detección de una actividad del usuario local para el punto extremo, del mismo modo, por ejemplo, que en una transición a descolgado.
- 3) Cuando expira el temporizador de "desconectado", se recibe una instrucción o se detecta una actividad del usuario local, la pasarela inicia el procedimiento de "desconectado" para el punto extremo. En el caso de una actividad del usuario local, debe además haber transcurrido un retardo de espera mínimo ($T_{d_{min}}$) de "desconectado" suministrable desde que la pasarela ha pasado al estado "desconectado" o desde la última vez que se inicio el procedimiento de "desconectado", al objeto de limitar la velocidad a la cual se ejecuta el procedimiento.
- 4) Si el procedimiento de "desconectado" todavía dejó el punto extremo desconectado, el valor del temporizador de "desconectado" se duplica entonces, sujeto a un retardo "de espera" máximo ($T_{d_{max}}$) de "desconectado" suministrable, por ejemplo, 600 segundos, y la pasarela continúa regresando al paso 2).

El procedimiento de "desconectado" es análogo al procedimiento de re arranque en cuanto éste ahora sencillamente establece que el punto extremo DEBE enviar una instrucción RestartInProgress al agente de llamada informándole de que el punto extremo ha sido desconectado y garantizar además que el primer mensaje (instrucción o respuesta) que ve ahora el agente de llamada procedente de este punto extremo DEBE ser esta instrucción RestartInProgress. El punto extremo DEBE aprovechar completamente el transporte en remolque para alcanzar este objetivo. El agente de llamada puede a continuación decidir si audita el punto extremo o sencillamente libera todas las conexiones de punto extremo.

Esta Recomendación deliberadamente no especifica ningún comportamiento adicional para un punto extremo desconectado. Los vendedores PUEDEN, por ejemplo, decidir proporcionar silencio, presentar el tono de volver a llamada (*reorder tone*) o incluso posibilitar que un fichero wav telecargado sea presentado en los puntos extremos afectados.

El valor por defecto de Td_{init} es de 15 segundos, el valor por defecto de Td_{min} es de 15 segundos y el valor por defecto de Td_{max} es de 600 segundos.

6.5 Códigos de retorno y códigos de error

Todas las instrucciones MGCP reciben una respuesta. La respuesta contiene un código de retorno que indica la situación de la instrucción. El código de retorno es un número entero para el cual se han especificado tres gamas de valores:

- el valor 000 indica un acuse de recibo de respuesta²⁵;
- los valores entre 100 y 199 indican una respuesta provisional;
- los valores entre 200 y 299 indican una compleción fructuosa;
- los valores entre 400 y 499 indican un error transitorio;
- los valores entre 500 y 599 indican un error permanente.

Los valores que se acaban de definir se listan en el siguiente cuadro:

Código	Significado
000	Acuse de recibo de respuesta.
100	La transacción se está ejecutando actualmente. Un mensaje de compleción real seguirá más tarde.
200	La transacción solicitada se ha ejecutado normalmente.
250	La conexión o conexiones han sido suprimidas.
400	La transacción no se pudo ejecutar debido a un error transitorio.
401	El teléfono ya está descolgado.
402	El teléfono ya esta colgado.
500	La transacción no se pudo ejecutar porque el punto extremo es desconocido.
501	La transacción no se pudo ejecutar porque el punto extremo no está preparado.
502	La transacción no se pudo ejecutar porque el punto extremo no dispone de recursos suficientes.
510	La transacción no se pudo ejecutar porque se ha detectado un protocolo de error.
511	La transacción no se pudo ejecutar porque la instrucción contenía una extensión no reconocida.
512	La transacción no se pudo ejecutar porque la pasarela no está equipada para detectar uno de los eventos solicitados.
513	La transacción no se pudo ejecutar porque la pasarela no esta equipada para generar una de las señales solicitadas.
514	La transacción no se pudo ejecutar porque la pasarela no puede enviar el aviso especificado.
515	La transacción se refiere a un id de conexión incorrecto (puede haber sido ya suprimido).

²⁵ El acuse de recibo de respuesta se utiliza para las respuestas provisionales (véase 7.8).

Código	Significado
516	La transacción se refiere a un id 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 se pudo ejecutar porque el punto extremo se encontraba en "rearranque".
521	Redirección del punto extremo a otro agente de llamada.
522	Ninguno de tales eventos o señales.
523	Acción desconocida o combinación ilegal de acciones.
524	Incoherencia interna en LocalConnectionOptions.
525	Extensión desconocida en LocalConnectionOptions.
526	Anchura de banda insuficiente.
527	Falta RemoteConnectionDescriptor.
528	Versión de protocolo incompatible.
529	Avería interna de soporte físico.
532	Valor(es) no soportado(s) en LocalConnectionOptions.
533	Respuesta demasiado extensa.

6.6 Códigos de motivo

Los códigos de motivo son utilizados por la pasarela cuando se suprime una conexión para informar al agente de llamada acerca el motivo de supresión de la conexión. El código de motivo es un número entero, y se definen los siguientes valores:

Código	Significado
900	Funcionamiento deficiente del punto extremo.
901	Punto extremo fuera de servicio.
902	Pérdida de conectividad de capa inferior (por ejemplo, sincronización en sentido descendente).
903	Se ha perdido la reserva de recursos QoS.

7 Protocolo de control de pasarela de medios

El MGCP implementa la interfaz de control de pasarela de medios como un conjunto de transacciones. Cada transacción está constituida por una instrucción y una respuesta obligatoria. Hay ocho tipos de instrucciones:

- CreateConnection (crear conexión).
- ModifyConnection (modificar conexión).
- DeleteConnection (suprimir conexión).
- NotificationRequest (petición de notificación).
- Notify (notificar).
- AuditEndpoint (auditar punto extremo).

- AuditConnection (auditar conexión).
- RestartInProgress (rearranque en curso).

Las cuatro primeras instrucciones son enviadas por el agente de llamada a una pasarela. La instrucción Notify es enviada por la pasarela al agente de llamada. La pasarela puede también enviar una instrucción DeleteConnection como se define en 6.3.6. El agente de llamada puede enviar a la pasarela cualquiera de las dos instrucciones auditar y, finalmente, la pasarela puede enviar una instrucción RestartInProgress al agente de llamada.

7.1 Descripción general

Todas las instrucciones están formadas por un encabezamiento de instrucción, el cual puede ir seguido en algunas instrucciones por una descripción de sesión.

Todas las respuestas están formadas por un encabezamiento de respuesta, el cual puede ir seguido en algunas instrucciones por una descripción de sesión.

Los encabezamientos y las descripciones de sesión se codifican como un conjunto de líneas de texto separadas por un carácter retroceso de carro y cambio de renglón (u, opcionalmente, por un carácter único de cambio de renglón). Los encabezamientos están separados de la descripción de sesión por una línea vacía.

El MGCP utiliza un identificador de transacción cuyo valor oscila entre 1 y 999999999 para relacionar las instrucciones y las respuestas. El identificador de transacción se codifica como un componente del encabezamiento de instrucción y se repite como un componente del encabezamiento de respuesta.

7.2 Encabezamiento de instrucción

El encabezamiento de instrucción está compuesto por:

- una línea de instrucción que identifica la acción o verbo solicitado, el identificador de transacción, el punto extremo hacia el que se solicita que se dirija la acción y la versión de protocolo MGCP;
- un conjunto de líneas de parámetro compuesto de un nombre de parámetro seguido de un valor de parámetro.

Salvo que otras normas referenciadas indiquen u ordenen lo contrario, cada componente del encabezamiento de instrucción es insensible a la posición mayúsculas/minúsculas. Esto atañe a los verbos y a los parámetros y valores, y todas las comparaciones DEBEN tratar por igual los caracteres en mayúsculas y en minúsculas así como las combinaciones de estos.

7.2.1 Línea de instrucción

La línea de instrucción está compuesta por:

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

Estos cuatro elementos se codifican como cadenas de caracteres ASCII imprimibles separadas por espacios en blanco, es decir, los caracteres ASCII espacio (0x20) o tabulación (0x09). Los clientes incorporados DEBERÍAN utilizar solamente un separador de espacio ASCII, si bien DEBEN ser capaces de analizar gramaticalmente los mensajes con caracteres de espacio en blanco adicionales.

7.2.1.1 Codificación de los verbos solicitados

Los verbos solicitados se codifican como cuatro caracteres ASCII de letras mayúsculas y/o minúsculas (las comparaciones DEBEN ser insensibles a la posición mayúsculas/minúsculas) tal como se muestra en el cuadro siguiente:

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

En las futuras versiones de esta Recomendación se pueden definir nuevos verbos. Puede ser necesario, para fines experimentales, utilizar verbos nuevos antes de sancionarlos en una versión publicada de esta Recomendación. Los verbos experimentales deberían identificarse por un código de cuatro letras que comience por la letra X (por ejemplo, XPER).

Un cliente incorporado que reciba una instrucción con un verbo experimental que no soporta DEBE devolver un error (código de error 511 – extensión no reconocida).

7.2.1.2 Identificadores de transacción

Los identificadores de transacción se utilizan para relacionar las instrucciones y las respuestas.

Un cliente incorporado soporta dos espacios de nombre de identificador de transacción separados:

- un espacio de nombre de identificador de transacción para la emisión de transacciones; y
- un espacio de nombre de identificador de transacción para la recepción de transacciones.

Como mínimo, los identificadores de transacción para instrucciones enviadas a un cliente incorporado determinado DEBEN ser exclusivos durante el tiempo de vida máximo de las transacciones dentro del conjunto de agentes de llamada que controla dicho cliente incorporado (véase 7.5). De este modo, con independencia del agente de llamada emisor, los clientes incorporados pueden siempre detectar las transacciones duplicadas mediante el examen sencillo del identificador de transacción. La coordinación de estos identificadores de transacción entre agentes de llamada cae sin embargo fuera del alcance de esta Recomendación.

Los identificadores de transacción para todas las instrucciones enviadas desde un cliente incorporado DEBEN ser exclusivos durante el tiempo de vida máximo de las transacciones (véase 7.5) con independencia del agente de llamada al que es enviada la instrucción. Con ello, el agente de llamada puede siempre detectar una transacción duplicada procedente de un cliente incorporado mediante la combinación del nombre de dominio del punto extremo y el identificador de transacción. El cliente incorporado a su vez puede detectar siempre un acuse de recibo de respuesta duplicado examinando el identificador o identificadores de transacción.

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

Los identificadores de transacción tienen valores comprendidos entre 1 y 999999999. Una entidad MGCP NO DEBE reutilizar un identificador de transacción antes de que transcurran tres minutos desde la compleción de la instrucción anterior en la cual se utilizó dicho identificador.

7.2.1.3 Codificación de los nombres de puntos extremos, de agentes de llamada y de NotifiedEntity

Los nombres de los puntos extremos y los nombres de los agentes de llamada se codifican como direcciones de e-electrónico, tal como se define en RFC 821. En estas direcciones, el nombre de dominio identifica el sistema al cual está asociado el punto extremo, mientras que el lado izquierdo identifica un punto extremo específico en ese sistema. Ambos componentes DEBEN ser insensibles a la posición mayúsculas/minúsculas.

Ejemplos de tales nombres son:

aaln/1@ncs2.whatever.net	Línea de acceso analógica 1 en el cliente incorporado ncs2 en la red "Whatever".
Call-agent@ca.whatever.net	Agente de llamada para la red "whatever".

El nombre de las entidades notificadas se expresa con la misma sintaxis, añadiendo posiblemente un número de puerto, como en:

```
Call-agent@ca.whatever.net:5234
```

cuando se omite el número de puerto, se aplicará el puerto MGCP por defecto (2427). En 6.1.1 se recogen más detalles sobre los nombres de puntos extremos.

7.2.1.4 Codificación de la versión de protocolo

La versión de protocolo se codifica con la palabra clave "MGCP" seguida de un espacio en blanco y el número de versión, que a su vez va seguido por el nombre de perfil "NCS" y un número de versión de perfil. Los números de versión están formados por un número de versión principal, un punto y un número de versión secundario. Los números de versión principal y secundario 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 NCS 1.0
```

La porción "NCS 1.0" indica que se trata del perfil NCS 1.0 de MGCP 1.0.

Una entidad que recibe una instrucción con una versión de protocolo que la entidad no soporta, DEBE responder con un error (código de error 528 – Versión de protocolo incompatible).

7.2.2 Líneas de parámetro

Las líneas de parámetro están formadas por un nombre de parámetro, que en la mayoría de los casos está compuesto por un solo carácter en mayúsculas, seguido de un carácter dos puntos, un espacio en blanco y el valor de parámetro. Los nombres y valores de parámetro son sin embargo todavía insensibles a la posición mayúsculas/minúsculas. Los parámetros que pueden estar presentes en las instrucciones se definen en el siguiente cuadro:

Nombre de parámetro	Código	Valor de parámetro
ResponseAck ²⁶	K	Véase descripción.
CallId	C	Cadena hexadecimal; la longitud NO DEBE exceder de 32 caracteres.
ConnectionId	I	Cadena hexadecimal; la longitud NO DEBE exceder de 32 caracteres.
NotifiedEntity	N	Identificador, en formato RFC 821, formado por una cadena arbitraria y el nombre de dominio de la entidad solicitante, que puede estar completado por un número de puerto, como en: Call-agent@ca.whatever.net:5234
RequestIdentifier	X	Cadena hexadecimal; la longitud NO DEBE exceder de 32 caracteres.
LocalConnectionOptions	L	Véase descripción.
Connection Mode	M	Véase descripción.
RequestedEvents	R	Véase descripción.
SignalRequests	S	Véase descripción.
DigitMap	D	Texto codificado de un mapa de dígitos.
ObservedEvents	O	Véase descripción.
ConnectionParameters	P	Véase descripción.
ReasonCode	E	Véase descripción.
SpecificEndPointId	Z	Identificador, en formato RFC 821, formado por una cadena arbitraria, seguido opcionalmente por un carácter "@" seguido a su vez por el nombre de dominio del cliente incorporado al que está asociado este punto extremo.
MaxEndPointIds	ZM	Cadena decimal; la longitud NO DEBE exceder de 16 caracteres.
NumEndPoints	ZN	Cadena decimal; la longitud NO DEBE exceder de 16 caracteres.
RequestedInfo	F	Véase descripción.
QuarantineHandling	Q	Véase descripción.
DetectEvents	T	Véase descripción.
EventStates	ES	Véase descripción.
ResourceID	DQ-RI	Véase descripción.
RestartMethod	RM	Véase descripción.
RestartDelay	RD	Número de segundos codificado como un número decimal.
Capabilities	A	Véase descripción.
VersionSupported	VS	Véase descripción.

²⁶ El parámetro ResponseAck no se ha mostrado en 6.3 ya que los identificadores de transacción no son visibles en nuestro ejemplo de API. Los implementadores pueden elegir un enfoque diferente.

Los parámetros no están necesariamente presentes en todas las instrucciones. En el siguiente cuadro se presenta la asociación entre parámetros e instrucciones. M indica obligatorio (mandatory), O opcional (optional) y F prohibido (forbidden):

Nombre de parámetro	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
ResponseAck ²⁶	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	M	O	F	F	F	F	F	F
Connection Mode (modo de conexión)	M	O	F	F	F	F	F	F
RequestedEvents	O*	O*	O*	O*	F	F	F	F
SignalRequests	O*	O*	O*	O*	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
DigitMap	O	O	O	O	F	F	F	F
Connection parameters (parámetros de la conexión)	F	F	O	F	F	F	F	F
Specific Endpoint Id (Id de punto extremo específico)	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
ResourceID	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
RemoteConnectionDescriptor	O	O	F	F	F	F	F	F
* Los parámetros RequestedEvents y SignalRequests son opcionales en la NotificationRequest. Si se omiten estos parámetros, las listas correspondientes se considerarán vacías. Para las instrucciones de tratamiento de la conexión, se aplica también este criterio cuando está incluido un RequestIdentifier.								

Los clientes incorporados y los agentes de llamada DEBERÍAN proporcionar siempre los parámetros obligatorios antes que los opcionales, si bien los clientes incorporados NO DEBEN fallar aunque no se cumpla esta recomendación.

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

X-FlowerOfTheDay: Daisy

Los nombres de parámetros que comiencen con "X+" son extensiones de parámetro obligatorias. Una pasarela que recibe una extensión de parámetro obligatoria que no puede comprender DEBE responder con un error (código de error 511 – extensión no reconocida).

Los nombres de parámetro que comienzan con "X-" son extensiones de parámetro no críticas. Una pasarela que recibe una extensión de parámetro no crítica que no puede comprender puede ignorar sin temor este parámetro.

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

Si se recibe una línea de parámetro con un parámetro prohibido, o con cualquier otro error de formato, la entidad receptora debe responder con el código de error más específico del error en cuestión. El código de error menos específico es 510 – error de protocolo. Se pueden proporcionar siempre comentarios en forma de texto.

7.2.2.1 ResponseAcknowledgement (Acuse de recibo de respuesta)

El parámetro acuse de recibo de respuesta²⁶ se utiliza para sustentar la toma de contacto de tres caminos descrita en 7.7. Contiene una lista de "gamas de identificadores de transacción confirmadas" separadas por comas.

Cada "gama de identificadores de transacción confirmada" está formada, bien por un número decimal, cuando la gama incluye solamente una transacción, o bien por dos números decimales separados por un guión que describen los identificadores de transacción inferior y superior incluidos en la gama.

Un ejemplo de acuse de recibo de respuesta es:

K: 6234-6255, 6257, 19030-19044

7.2.2.2 RequestIdentifier (identificador de petición)

El identificador de petición relaciona una instrucción Notify con la NotificationRequest que la desencadena. Un RequestIdentifier es una cadena hexadecimal; la longitud NO DEBE exceder de 32 caracteres. La cadena "0" se reserva para informar de los eventos persistentes en el caso en que no se haya recibido todavía ninguna NotificationRequest (véase 6.3.2)

7.2.2.3 Local Connection Options (Opciones de la conexión local)

Las opciones de la conexión local describen los parámetros operacionales que los agentes de llamada ordenan a la pasarela que utilice para una conexión. Estos parámetros son:

- El periodo de paquetización en milisegundos, codificado como la palabra clave "p" seguida por un carácter dos puntos y un número decimal.
- El nombre literal del algoritmo de compresión, codificado como la palabra clave "a" seguida por un carácter dos puntos y una cadena de caracteres.
- El parámetro cancelación de eco, codificado como la palabra clave "e" seguida por un carácter dos puntos y el valor "on" (activado) u "off" (desactivado).
- El parámetro tipo de servicio, codificado como la palabra clave "t" seguida por un carácter dos puntos y el valor codificado como dos dígitos hexadecimales.
- El parámetro supresión de silencio, codificado como la palabra clave "s" seguida por un carácter dos puntos y el valor "on" (activado) u "off" (desactivado).

Los parámetros LocalConnectionOptions utilizados para la calidad de servicio dinámica son:

- El parámetro identificador de puerta (GateID) D-QoS, codificado como la palabra clave "dq-gi" seguida por un carácter dos puntos y una cadena de hasta 8 caracteres hexadecimal correspondiente a un identificador de 32 bits para el GateID.

- El parámetro reserva de recursos D-QoS, codificado como la palabra clave "dq-rr" seguida por un carácter dos puntos y una cadena de caracteres. Puede especificarse una lista de valores, en cuyo caso estos valores irán separados por un carácter punto y coma. Los valores posibles son:

Modo	Significado
sendresv	Reserva en el sentido de emisión solamente
recvresv	Reserva en el sentido de recepción solamente
snrcresv	Reserven en los sentidos de emisión y de recepción
sendcomt	Compromiso en el sentido de emisión solamente
recvcomt	Compromiso en el sentido de recepción solamente
snrccomt	Compromiso en los sentidos de emisión y de recepción

- El ResourceID, codificado como la palabra clave "dq-ri" seguida por un carácter dos puntos y una cadena de hasta 8 caracteres hexadecimales correspondientes a un identificador de 32 bits para el ResourceID.
- El ReserveDestination, codificado como la palabra clave "dq-rd" seguida por un carácter dos puntos y una dirección IP codificada de manera similar a una dirección IP para la porción nombre de dominio de un nombre de punto extremo. El parámetro ReserveDestination puede opcionalmente ir seguido por un carácter dos puntos y hasta 5 caracteres decimales para un número de puerto UDP que se ha de utilizar.

Los parámetros LocalConnectionOptions utilizados para seguridad se codifican del modo siguiente:

- El secret se codifica como la palabra clave "sc-st" seguida por un carácter dos puntos, un método, una carácter dos puntos y el secret actual. El método está formado, bien por la cadena "clear" si el secreto está codificado en texto claro, o bien por la cadena "base64" si el secreto está codificado en base64.
- La ciphersuite RTP se codifica como la palabra clave "sc-rtp" seguida por un carácter dos puntos y una cadena de ciphersuite RTP como se define más adelante. Se puede especificar una lista de valores, en cuyo caso los valores estarán separados por caracteres punto y coma.
- La ciphersuite RTCP se codifica como la palabra clave "sc-rtcp" seguida por un carácter dos puntos y una cadena de ciphersuite RTCP como se define más adelante. Se puede especificar una lista de valores, en cuyo caso estos valores estarán separados por caracteres punto y coma.

Las cadenas de ciphersuite RTP y RTCP se ajustan a la gramática siguiente:

```

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

```

donde ALPHA y DIGIT son los definidos en RFC 2234. Dentro de una ciphersuite no se permiten los espacios en blanco. El siguiente ejemplo ilustra el empleo de la ciphersuite:

62/51

La lista real de ciphersuites soportadas por IPCablecom se proporcionará en la Especificación UIT-T J.170 de seguridad de IPCablecom.

Cuando están presentes varios parámetros, los valores se separan por una coma. SE DEBE considerar un error la inclusión de un parámetro sin valor (código de error 524 – incoherencia de LocalConnectionOptions).

Son ejemplo de opciones de conexión local los siguientes:

```
L: p:10, a:PCMU  
L: p:10, a:PCMU, e:off, t:20, s:on  
L: p:30, a:G729A, e:on, t:A0, s:off
```

Un tipo de servicio de valor hex "20" implica una precedencia IP de 1, y un tipo de servicio de valor hex "A0" implica una precedencia IP de 5.

Este conjunto de atributos puede ampliarse mediante atributos extensión (*extension*). Los atributos extensión están formados por un nombre de atributo, seguido por un carácter de puntos y de una lista de valores de atributo separados por caracteres punto y coma. El nombre de atributo DEBE comenzar con los dos caracteres, "x+" para una extensión obligatoria, o "x-" para una extensión no obligatoria. Si una pasarela recibe un atributo de extensión obligatoria que no reconoce, DEBE rechazar la instrucción con un error (código de error 525 –Extensión desconocida en LocalConnectionOptions).

7.2.2.4 Capacidades (Capabilities)

Capacidades informa al agente de llamada sobre sus capacidades cuando es auditado. La codificación de las capacidades se basa en la codificación de las opciones de conexión local para los parámetros que son comunes a ambas. Además, las capacidades pueden también contener una lista de lotes soportados y una lista de modos soportados.

Los parámetros utilizados son:

- El periodo de paquetización en milisegundos, codificado como la palabra clave "p" seguida por un carácter dos puntos y un número decimal. Se puede especificar una gama como dos números decimales separados por un guión.
- El nombre literal del algoritmo de compresión, codificado como la palabra clave "a" seguida por un carácter dos puntos y una cadena de caracteres. Se puede especificar una lista de valores, en cuyo caso estos valores irán separados por caracteres punto y coma.
- La anchura de banda en kilobits por segundo (1000 bits por segundo), codificada como la palabra clave "b" seguida por un carácter dos puntos y un número decimal. Se puede especificar una gama como dos números decimales separados por un guión.
- El parámetro cancelación de eco, codificado como la palabra clave "e" seguida por un carácter dos puntos y por el valor "on" (activado) si se soporta la cancelación de eco, y por el valor "off" (desactivado) en el caso contrario.
- El parámetro tipo de servicio, codificado como la palabra "t" seguida por un carácter dos puntos y por el valor "0" si no se soporta el tipo de servicio, indicando todos los demás valores que se soporta el tipo de servicio.
- El parámetro supresión de silencio, codificado como la palabra clave "s" seguida por un carácter dos puntos y el valor "on" (activado) si se soporta la supresión de silencio, y por el valor "off" (desactivado) en caso contrario.
- Los lotes de eventos soportados por este punto extremo, codificados como la palabra clave "v" seguida por un carácter dos puntos y seguida por una lista de los nombres de los lotes soportados separados por caracteres 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 por un carácter dos puntos y una lista de modos de conexión soportados separados por caracteres punto y coma como se define en 7.2.2.7.

- La palabra clave "dq-gi" si se soporta la calidad de servicio dinámica.
- La palabra clave "sc-st" si se soporta la seguridad IPCablecom. En este caso las ciphersuites soportadas vienen indicadas por las siguientes palabras clave:
 - La palabra clave "sc-rtp" seguida por un carácter dos puntos y por una lista de AuthenticationAlgorithms (algoritmos de autenticación) RTP separados por caracteres punto y coma, una barra de fracción, y una lista de EncryptionAlgorithms (algoritmos de criptación) soportados separados por caracteres punto y coma.
 - La palabra clave "sc-rtcp" seguida de un carácter dos puntos y de una lista de AuthenticationAlgorithms (algoritmos de autenticación) RTCP separados por caracteres punto y coma, una barra fracción, y una lista de EncryptionAlgorithms (algoritmos de criptación) soportados separados por caracteres punto y coma.

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

Son ejemplos de capacidades los siguientes:

```
A: a:PCMU;G729A, p:10-100, e:on, s:off, v:L;S,
m:sendonly;recvonly;sendrecv;inactive
A: a:G729A; p:30-90, e:on, s:on, v:L;S,
m:sendonly;recvonly;sendrecv;inactive;confrnce,
dq-gi, sc-st, sc-rtp: 00/51;03
```

Obsérvese que los códecs y algoritmos de seguridad son meros ejemplos – especificaciones IPCablecom separadas detallan los códecs y algoritmos reales soportados, así como la codificación utilizada.

7.2.2.5 Parámetros de conexión

Los parámetros de conexión se codifican como una cadena de parejas de tipo y valor, donde el tipo es un identificador de dos letras del parámetro, y el valor un número entero decimal. Los tipos van separados de los valores por un signo "=". Los parámetros están separados entre sí por una coma.

Los tipos de parámetro de conexión se especifican en el cuadro siguiente:

Nombre del parámetro de conexión	Código	Valor del parámetro de conexión
Paquetes enviados	PS	El número de paquetes que se envían por la conexión.
Octetos enviados	OS	El número de octetos que se envían por la conexión.
Paquetes recibidos	PR	El número de paquetes que se reciben por la conexión.
Octetos recibidos	OR	El número de octetos que se reciben por la conexión.
Paquetes perdidos	PL	El número de paquetes que no se han recibido por la conexión, deducido de los saltos en el numero secuencial.
Fluctuación	JI	Valor medio de la fluctuación de llegada entre paquetes, en milisegundos, expresado como un número entero.
Latencia	LA	Valor medio de la latencia, en milisegundos, expresado como un número entero.

Los nombres de los parámetros de conexión de extensión están formados por la cadena "X-" seguida de un nombre de parámetro de extensión de dos letras. Los agentes de llamada que reciben extensiones no reconocidas DEBEN tranquilamente ignorar estas extensiones.

Un ejemplo de codificación de parámetro de conexión es:

```
P: PS=1245, OS=62345, PR=0, OR=0, PL=0, JI=0, LA=48
```


7.2.2.6 Códigos de motivo

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

```
900 Endpoint malfunctioning
```

En 6.6 se presenta una lista de códigos de motivo.

7.2.2.7 Modos de conexión

El modo de conexión describe el modo de explotación de la conexión. Los valores posibles son:

Modo	Significado
M: sendonly	La pasarela solamente enviaría paquetes
M: recvonly	La pasarela solamente recibiría paquetes.
M: sendrecv	La pasarela enviaría y recibiría paquetes.
M: confnrc	La pasarela enviaría y recibiría paquetes en el modo conferencia.
M: inactive	La pasarela no enviaría ni recibiría paquetes.
M: replcate	La pasarela solamente enviaría paquetes según el modo replicate.
M: netwloop	La pasarela pondría el punto extremo en el modo bucle de red.
M: netwtst	La pasarela pondría el punto extremo en el modo prueba de continuidad de red.

7.2.2.8 Codificación de los nombres de los eventos/señales

Los nombres de los eventos/señales se componen de un nombre de lote opcional, separado por una barra de fracción (/) del nombre del evento real. El nombre del evento puede ir seguido opcionalmente de un signo (@) y del identificador de la conexión en la cual se debe observar el evento. Los nombres de eventos son utilizados en los parámetros RequestedEvents, SignalRequests, DetectEvents, ObservedEvents y EventStates. Cada evento se identifica mediante un código de evento. Estas codificaciones ASCII no son sensibles a la posición mayúsculas/minúsculas. Valores como "hu", "Hu", "HU" o "hU" se deben considerar iguales.

Los siguientes son ejemplos de nombres de eventos:

X/hu	Transición a colgado, en el ejemplo de lote de línea
X/0	Dígito 0 en el ejemplo de lote de línea
hf	Cambio instantáneo, suponiendo que el ejemplo de lote de línea es el lote por defecto para el punto extremo.
X/rt@0A3F58	Llamada de retorno en la conexión "0A3F58"

Además, se pueden utilizar la gama y la notación con comodines de los eventos, en lugar de los nombres individuales, en los parámetros RequestedEvents y DetectEvents (pero no en los parámetros SignalRequests ObservedEvents o EventStates) como sigue:

X/[0-9]	Dígitos de 0 a 9 en el ejemplo de lote de línea
X/X	Dígitos de 0 a 9 en el ejemplo de lote de línea
[0-9*#A-D]	Todos los dígitos y letras en el ejemplo de lote de línea (valor por defecto para el punto extremo).
X/todos	Todos los eventos en el ejemplo de lote de línea.

Finalmente, se puede utilizar el signo asterisco (estrella) para indicar "todas las conexiones", y el signo de dólar para indicar la conexión "actual". Los siguientes son ejemplos de tales notaciones:

X/rt@*	Llamada de retorno en todas las conexiones para el punto extremo.
X/rt@\$	Llamada de retorno en la conexión actual.

En el anexo A se puede ver un conjunto inicial de lotes de eventos para clientes incorporados.

7.2.2.9 RequestedEvents

El parámetro RequestedEvents proporciona la lista de eventos que han sido solicitados. Los códigos de eventos definidos normalmente se describen en el anexo A.

Cada evento se puede calificar por una acción solicitada, o por una lista de acciones. No todas las acciones se pueden combinar – las combinaciones válidas pueden verse en 6.3.1. Las acciones, cuando se especifican, se codifican como una lista de palabras clave encerradas entre paréntesis y separadas por comas. Los códigos para las distintas acciones son:

Acción	Código
Notify immediately	N
Accumulate	A
Accumulate according to digit map	D
Ignore	I
Keep Signal(s) active	K
Embedded NotificationRequest	E
Embedded ModifyConnection	C

Si no se proporciona un mapa de dígitos cuando se especifica la acción "accumulate according to digit map" (acumular de acuerdo con mapa de dígitos), el punto extremo sencillamente utiliza su mapa de dígitos vigente. Si el punto extremo no tiene actualmente ningún mapa de dígitos, debe devolverse un error (código de error 519 – ningún mapa de dígitos).

Cuando no se especifica ninguna acción, la acción por defecto es notificar el evento. Ello implica que, por ejemplo, "oc" y "oc(N)" son equivalentes. Los eventos que no están en la lista son descartados, a excepción de los eventos persistentes.

La acción mapa de dígitos solamente se puede especificar para los dígitos, letras y temporizadores.

La lista de eventos solicitados se codifica en una sola línea, con los grupos de eventos/acciones separados por comas. Son ejemplos de codificaciones de RequestedEvents (utilizando el ejemplo de lote de línea) los siguientes:

```
R: hu (N) , hf (N)          Notify on-hook, notify hook-flash.
R: hu (N) , [0-9#T] (D)    Notify on-hook, accumulate digits according to digit
                             map.
```

La NotificationRequest incorporada sigue el formato:

```
E ( R( <RequestedEvents>), D( <Digit Map>), S( <SignalRequests> ) )
```

donde cada uno de los R, D y S son opcionales y se habrán suministrado probablemente en otro orden. El siguiente ejemplo ilustra la utilización de Embedded NotificationRequest con el ejemplo de lote de línea:

```
R: hd(A, E(S(d1), R( B/oc(N), [0-9#T](D) ) ), D((1xxxxxxxxxxx|9011x.T)) ) )
```

Tras descolgar, acumular el evento, proporcionar el tono de invitación a marcar e iniciar la acumulación de dígitos de acuerdo con el mapa de dígitos proporcionado. Detener el tono de invitación a marcar cuando se introduce el primer dígito, o, si no se introduce ningún dígito antes de expirar la temporización del tono de invitación a marcar, notificar la operación completada. En cualquier otro caso, notificar la operación de descolgar y los dígitos recopilados cuando se ha producido una concordancia, discordancia o temporización entre dígitos. Debe señalarse que, como colgar es un evento persistente, será no obstante detectado e identificado aunque no haya sido especificado aquí.

La acción ModifyConnection incorporada se ajusta al formato:

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

En el siguiente ejemplo se ilustra la utilización de la acción Embedded ModifyConnection con el ejemplo de lote de línea:

```
R: hf(A, C(M(inactive(X43DC)), M(sendrecv($)))) , B/oc(N) , B/of(N)
```

Tras el accionamiento del gancho conmutador, cambiar el modo de la conexión "X43DC" a "inactive" ("inactivo"), y cambiar después el modo de la "conexión actual" a "send receive" ("enviar y recibir"). Eventos Notify tras "operación completada" y "operación fracasada".

7.2.2.10 SignalRequests

El parámetro SignalRequests proporciona el nombre de las señales que se han solicitado. En el anexo A se pueden ver las señales definidas normalmente. Una señal determinada sólo puede aparecer una vez en la lista, y todas las señales serán, por definición, aplicadas al mismo tiempo.

Algunas señales se pueden calificar mediante parámetros de señal. Cuando se califica una señal por medio de varios parámetros de señal éstos se separan por comas. Cada parámetro de señal DEBE ajustarse al formato especificado a continuación (se permiten espacios en blanco):

```
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 el valor de parámetro de señal puede ser, bien una cadena o bien una cadena entrecomillada, es decir, una cadena encerrada por dos dobles comillas. Dos dobles comillas consecutivas en una cadena entrecomillada se convertirá en una doble comilla dentro de esa cadena entrecomillada. Por ejemplo, "ab" "c" producirá la cadena ab"c .

Cada señal tiene asociado uno de los siguientes tipos de señal (véase 6.3.1):

- On/Off (OO) (activado/desactivado);
- Time-out (TO) (temporización);
- Brief (BR) (breve).

Las señales On/Off (activado/desactivado) se pueden parametrizar con "+" para pasar la señal a activado, o con "-" para conmutar la señal a desactivado. Si una señal on/off no está parametrizada, la señal se conmuta a activado. Las dos siguientes conmutarán la señal vmwi del ejemplo de lote de línea a la situación de activado:

vmwi(+), vmwi

Las señales Time-out (temporización) se pueden parametrizar con el parámetro de señal "TO" y un valor de temporización que sustituye al valor de temporización por defecto. Si una señal de temporización no está parametrizada con un valor de temporización, se utilizará el valor de temporización por defecto. Las dos siguientes aplicarán la señal de timbre del ejemplo de lote de línea durante 6 segundos:

rg(to=6000)

rg(to(6000))

Señales individuales pueden definir parámetros de señal adicionales.

Los parámetros de señal se encerrarán entre paréntesis como se ha mostrado anteriormente

Cuando se solicitan varias señales, sus códigos se separan sencillamente por una coma.

7.2.2.11 ObservedEvents

Los parámetros de eventos observados proporcionan la lista de los eventos que se han observado. Los códigos de evento son los mismos que se han utilizado en NotificationRequest. Cuando se detecta un evento en una conexión, el evento observado identificará la conexión en la que el mismo ha sido detectado utilizando la sintaxis "@<connection>". Son ejemplo de eventos observados que utilizan el ejemplo de lote de línea los siguientes:

O: hu

O: 8,2,9,5,5,5,5,T

O: hf,hf,hu

Los eventos que se han acumulados de conformidad con el mapa de dígitos, se comunican como eventos individuales en el orden en que se han detectado. Otros eventos pueden mezclarse entre sí. Debe señalarse que si la "cadena de marcación vigente" no está vacía y tiene una concordancia parcial, y se produce otro evento que da como resultado la generación de un mensaje Notify, la "cadena de marcación vigente" parcialmente concordante se incluirá en la lista de eventos observados, y la "cadena de marcación vigente" será entonces liberada – consúltese 6.4.3.1 para más detalles.

7.2.2.12 RequestedInfo

El parámetro RequestedInfo contiene una lista de códigos de parámetro separados por comas, tal como se define en la cláusula "Líneas de parámetros" – En la cláusula 6.3.8 se listan los parámetros que pueden auditarse. Se soportan también los siguientes valores:

Parámetro RequestedInfo	Código
LocalConnectionDescriptor	LC
RemoteConnectionDescriptor	RC

Por ejemplo, si se desea auditar el valor de los parámetros NotifiedEntity, RequestIdentifier, RequestedEvents, SignalRequests, DigitMap, DetectEvents, EventStates, LocalConnectionDescriptor y RemoteConnectionDescriptor, el valor del parámetro RequestedInfo será:

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

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

F: A

7.2.2.13 QuarantineHandling

El parámetro de tratamiento en cuarentena contiene las palabras clave "procesar" o "descartar" para indicar el tratamiento de los eventos en cuarentena, por ejemplo:

Q: process

7.2.2.14 DetectEvents

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

T: hu, hd, hf, [0-9#*]

Debe observarse que no se puede asociar ninguna acción a los eventos.

7.2.2.15 EventStates

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

ES: hu

Debe observarse que no se puede asociar ninguna acción a los eventos.

7.2.2.16 ResourceID

El parámetro ResourceID es un parámetro de retorno utilizado en la calidad de servicio dinámica para señalar el ID de recursos asignado para la puerta en cuestión. El ResourceID se codifica como una cadena de hasta 8 caracteres hex, como por ejemplo:

DQ-RI: AB345DC

7.2.2.17 RestartMethod

El parámetro RestartMethod se codifica como una de las palabras clave "graceful" (cuidadoso), "forced" (obligado), "restart" (rearranque) o "disconnected" (desconectado), como por ejemplo:

RM: restart

7.2.2.18 VersionSupported

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

VS: MGCP 1.0, MGCP 1.0 NCS 1.0

7.3 Formatos del encabezamiento de respuesta

El encabezamiento de respuesta está formado por una línea de respuesta seguida opcionalmente por encabezamientos que codifican los parámetros de respuesta.

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

200 1201 OK

En el siguiente cuadro se resumen los parámetros de respuesta cuya presencia es obligatoria u opcional en un encabezamiento de respuesta, de acuerdo con la instrucción que desencadenó la respuesta suponiendo que esta instrucción haya tenido éxito. El lector debería no obstante examinar la definición de cada una de las instrucciones individuales ya que este cuadro solamente proporciona información resumida. La letra M significa obligatorio (mandatory), O significa opcional (optional) y F prohibido (forbidden).

Nombre de parámetro	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
ResponseAck ²⁶	O*	O*	O*	O*	O*	O*	O*	O*
CallId	F	F	F	F	F	F	O	F
ConnectionId	M	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
Connection Mode	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
DigitMap	F	F	F	F	F	O	F	F
ConnectionParameters	F	F	O	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
ResourceID	O	O	F	F	F	F	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	M	O	F	F	F	F	O	F
RemoteConnection Descriptor	F	F	F	F	F	F	O	F

* El parámetro ResponseAck NO SE DEBE utilizar con ninguna otra respuesta distinta de la respuesta definitiva enviada después de una respuesta provisional para la transacción en cuestión. En este caso, la presencia del parámetro ResponseAck DEBE activar un mensaje "Response Acknowledgement" (acuse-de-recibo-de-respuesta) – no deberá ignorarse ningún valor de ResponseAck proporcionado.

En lo que sigue se describen los parámetros de respuesta para cada una de las instrucciones.

7.3.1 CreateConnection

En el caso de un mensaje CreateConnection, la línea de respuesta va seguida de un parámetro Connection-Id con una respuesta fructuosa (código 200). Se transmite además un LocalConnectionDescriptor con una respuesta positiva. El LocalConnectionDescriptor se codifica como una "descripción de sesión", tal como se define en 7.4. 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 96
a=rtpmap:96 G726-32/8000
```

Cuando se ha emitido con anterioridad una respuesta provisional, la respuesta definitiva puede contener además el parámetro Response Acknowledgement (acuse-de-recibo-de-respuesta), y cuando se utiliza la calidad de servicio dinámica y la respuesta definitiva puede también contener un ResourceID, como en:

```
200 1204 OK
K:
I: FDE234C8
DQ-RI: 23DB4A43

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 96
a=rtpmap:96 G726-32/8000
```

Se acusa recibo de la respuesta definitiva mediante un Response Acknowledgement:

```
000 1204
```

7.3.2 ModifyConnection

En el caso de un mensaje ModifyConnection exitoso, la línea de respuesta va seguida por un LocalConnectionDescriptor si la modificación ha dado como resultado un cambio de los parámetros de sesión (por ejemplo, cambiando solamente el modo de una conexión no se alteran los parámetros de la sesión). El LocalConnectionDescriptor se codifica como una "descripción de sesión", tal como se define en 7.4. 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
```

La respuesta puede contener también un ResourceID cuando se utiliza calidad de servicio dinámica, como en:

```
200 1207 OK
DQ-RI: 12345
```

Cuando se ha enviado con anterioridad una respuesta provisional, la respuesta definitiva puede contener además el parámetro Response Acknowledgement como en:

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

Se acusa recibo de la respuesta definitiva mediante un Response Acknowledgement:

```
000 1207 OK
```

7.3.3 DeleteConnection

Dependiendo de la variante del mensaje DeleteConnection, la línea de respuesta puede ir seguida por una línea de parámetros de conexión, como se define en 7.2.2.5.

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

7.3.4 NotificationRequest

Una respuesta de NotificationRequest no incluye ningún parámetro de respuesta adicional.

7.3.5 Notify

Una respuesta de Notify no incluye ningún parámetro de respuesta adicional.

7.3.6 AuditEndpoint

En el caso de una instrucción AuditEndPoint la línea de respuesta puede ir seguida de información sobre cada uno de los parámetros solicitados – cada parámetro aparecerá en una línea separada. Deberán proporcionarse, no obstante, los parámetros para los cuales no existe ningún valor actualmente, por ejemplo, mapa de dígitos. Cada nombre de punto extremo local "ampliado" por medio de un carácter comodín aparecerá en una línea separada utilizando el código de parámetro "SpecificEndPointId", por ejemplo:

```
200 1200 OK
Z: aaln/1@rgw.whatever.net
Z: aaln/2@rgw.whatever.net
```

O:

```
200 1200 OK
A: a:PCMU;G728, p:10-100, e:on, s:off, t:1, v:X;B,
  m:sendonly;recvonly;sendrecv;inactive
A: a:G729A; p:30-90, e:on, s:on, t:1, v:X;B,
  m:sendonly;recvonly;sendrecv;inactive;confrnce
```

7.3.7 AuditConnection

En el caso de una instrucción AuditConnection, la respuesta puede ir seguida de información sobre cada uno de los parámetros solicitados. Deberán proporcionarse, no obstante, los parámetros para los que no existe actualmente ningún valor. Los descriptores de la conexión aparecerán siempre al final, y cada uno irá precedido por una línea vacía, como por ejemplo:

```
200 1203 OK
C: A3C47F21456789F0
N: [128.96.41.12]
L: p: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 G726-32/8000
```

Si se proporcionan 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 para la conexión auditada, este descriptor de conexión aparecerá solamente con el campo versión de protocolo SDP.

7.3.8 RestartInProgress

La respuesta a una instrucción RestartInProgress puede incluir el nombre de otro agente de llamada para contactar, por ejemplo, cuando el agente de llamada redirige el punto extremo a otro agente de llamada como en:

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

7.4 Codificación de la descripción de sesión

La descripción de sesión se codifica de conformidad con el protocolo de descripción de sesión (SDP, *session description protocol*), si bien los clientes incorporados pueden establecer ciertos supuestos simplificadores acerca de la descripción de sesión como se especifica seguidamente. Debe señalarse que las descripciones de sesión son sensibles a la posición mayúsculas/minúsculas según RFC 2327.

La utilización del SDP depende del tipo de sesión, especificado en el parámetro "medios":

- Si el parámetro medios se fija a "audio", la descripción de sesión se refiere a un servicio audio,
- Si el parámetro medios se fija a "vídeo", la descripción de sesión se refiere a un servicio vídeo,

En el caso de un servicio audio, la pasarela deberá considerar la información proporcionada en SDP para el parámetro medios "audio", y en caso de un servicio de vídeo la pasarela deberá considerar la información proporcionada en SDP para el parámetro medios "vídeo".

7.4.1 Utilización del servicio audio de SDP

En una pasarela de voz solamente, tendremos que describir solamente las sesiones que utilizan únicamente el parámetro medios "audio". Los parámetros de SDP que son pertinentes para la aplicación basada en la voz se especifican seguidamente. Los clientes incorporados DEBEN soportar las descripciones de sesión que se ajusten a estas reglas y en el siguiente orden:

- 1) El perfil SDP presentado a continuación.
- 2) RFC 2327 (*SDP: Session Description Protocol* –Protocolo de descripción de sesión).

El perfil SDP proporcionado describe la utilización del protocolo de descripción de sesión en la NCS. La descripción y explicación generales de cada uno de los parámetros se pueden ver en RFC 2327, si bien más adelante detallamos cuales son los valores de puntos extremos NCS que se han proporcionar para estos campos (emisión) y qué puntos extremos NCS deben realizarse con valores suministrados o no suministrados para estos campos (recepción).

Ningún parámetro no especificado más abajo NO DEBERÍA ser proporcionado por ningún punto extremo NCS, y en caso de que se recibiera tal parámetro DEBERÍA ser ignorado

7.4.1.1 Versión de protocolo (v=)

v= <version>
v= 0

Emisión: DEBE suministrarse de conformidad con RFC 2327 (es decir, v=0).

Recepción: DEBE suministrarse de conformidad con RFC 2327.

7.4.1.2 Origen (o=)

El campo origen (origen) se compone (o=) de 6 subcampos en RFC 2327:

o= <username> <session-ID> <version> <network-type> <address-type> <address>
o= - 2987933615 2987933615 IN IP4 A3C47F2146789F0

Username (nombre de usuario)

Emisión: DEBE utilizarse el guión como nombre de usuario cuando se requiere privacidad DEBERÍA utilizarse el guión en caso contrario²⁷.

Recepción: Este campo DEBERÍA ignorarse.

Session-ID (ID de sesión)

Emisión: DEBE ajustarse a RFC 2327 para la interoperabilidad con clientes distintos de los clientes IPCablecom.

Recepción: Este campo DEBERÍA ignorarse.

Version (versión)

Emisión: De conformidad con RFC 2327.

Recepción: Este campo DEBERÍA ignorarse.

Network Type (tipo de red)

Emisión: DEBE utilizarse el tipo "RI".

Recepción: Este campo DEBERÍA ignorarse.

Address Type (tipo de dirección)

Emisión: DEBE utilizarse el tipo "IP4".

Recepción: Este campo DEBERÍA ignorarse.

Address (dirección):

Emisión: DEBE ajustarse a RFC 2327 para la interoperabilidad con clientes distintos de los clientes IPCablecom.

Recepción: Este campo DEBERÍA ignorarse.

7.4.1.3 Nombre de sesión (s=)

s= <session-name>
s= -

Emisión: DEBE utilizarse el guión como nombre de sesión.

Recepción: Este campo DEBE ignorarse.

²⁷ Puesto que los puntos extremos de NCS no conocen cuándo se requiere privacidad DEBERÍAN utilizar siempre un guión.

7.4.1.4 Información de medios y de sesión (i=)

i= <session-description>

Emisión: Para la NCS, el campo NO DEBE utilizarse.

Recepción: Este campo DEBE ignorarse.

7.4.1.5 URI (u=)

u= <URI>

Emisión: Para la NCS, el campo NO DEBE utilizarse.

Recepción: Este campo DEBE ignorarse.

7.4.1.6 Dirección de e-electrónico y número de teléfono (e=, p=)

e= <e-mail-address>

p= <phone-number>

Emisión: Para la NCS, el campo NO DEBE utilizarse.

Recepción: Este campo DEBE ignorarse.

7.4.1.7 Datos de conexión (c=)

Los datos de conexión se componen de 3 subcampos:

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

c= IN IP4 10.10.111.11

Network Type (tipo de red):

Emisión: DEBE utilizarse el tipo "RI".

Recepción: DEBE estar presente el tipo "RI".

Address Type (tipo de dirección):

Emisión: DEBE utilizarse el tipo "IP4".

Recepción: DEBE estar presente el tipo "IP4".

Connection Address (dirección de conexión):

Emisión: Este campo DEBE rellenarse con una dirección IP de unidifusión en la cual la aplicación recibirá el tren de medios. Por ello un valor TTL (tiempo de vida) NO DEBE estar presente y un valor "número de direcciones" NO DEBE estar presente. El campo NO DEBE rellenarse con un nombre de dominio completamente calificado en lugar de con un dirección IP. Un dirección distinta de cero especifica **la dirección de emisión y la dirección de recepción para el tren o trenes de medios que cubre.**

Recepción: DEBE estar presente una dirección IP de unidifusión o un nombre de dominio completamente calificado. Un dirección distinta de cero especifica la dirección de emisión y la dirección de recepción para el tren o trenes de medios que cubre.

7.4.1.8 Anchura de banda (b=)

b= <modifier> : <bandwidth-value>

b= AS : 64

Emisión: La información de anchura de banda es opcional en SDP pero SE DEBERÍA incluir siempre²⁸. Cuando se utiliza un rtpmap (mapa RTP) o un códec no suficientemente conocido²⁹, SE DEBE utilizar la información de anchura de banda.

Recepción: SE DEBERÍA incluir la información de anchura de banda. Si no se incluye un modificador de anchura de banda, el receptor DEBE suponer valores por defecto de la anchura de banda que sean razonables los para códecos conocidos.

Modifier (Modificador):

Emisión: SE DEBE utilizar el tipo "AS".

Recepción: DEBE estar presente el tipo "AS".

Bandwidth value (Valor de anchura de banda):

Emisión: El campo SE DEBE llenar con el requisito de anchura de banda máxima del tren de medios, en kilobits por segundo.

Recepción: DEBE estar presente el requisito de anchura de banda máxima del tren de medios, en kilobits por segundo.

7.4.1.9 Tiempo (Time), Tiempos de repetición (Repeat Times) y zonas horarias (Time Zones) (t=, r=, z=)

t= <start-time> <stop-time>

t= 36124033 0

r= <repeat-interval> <active-duration> <list-of-offsets-from-start-time>

z= <adjustment-time> <offset>

Emisión: Time DEBE estar presente: start-time (tiempo de arranque) PUEDE ser cero, aunque DEBERÍA ser el tiempo actual, y stop-time (tiempo de parada) DEBERÍA ser cero. Repeat Times (tiempos de repetición) y Time zones (zonas horarias) NO DEBERÍAN utilizarse, y en caso de que se utilicen deben ajustarse a RFC 2327.

Recepción: Si no están presentes ninguno de estos campos, DEBERÍAN ignorarse.

7.4.1.10 Claves de criptación

k= <method>

k= <method> : <encryption-keys>

Los servicios de seguridad para IPCablecom se definirán mediante la Especificación UIT-T J.170 de seguridad de IPCablecom. Los servicios de seguridad especificados para RTP y RTCP no cumplen las normas RFC 1889, RFC 1890 y RFC 2327. En aras de la interoperabilidad con dispositivos distintos de los IPCablecom, no se utilizará por tanto el parámetro "k" para transportar los parámetros de seguridad.

Emisión: NO SE DEBE utilizar.

Recepción: Este campo DEBERÍA ignorarse.

²⁸ Si no se utiliza este campo, el controlador de puerta no puede autorizar la anchura de banda apropiada.

²⁹ Un códec no suficientemente conocido es un códec que no está definido en la especificación de códecos J.161.

7.4.1.11 Atributos (a=)

```
a= <attribute> : <value>
a= rtpmap : <payload type> <encoding name>/<clock rate> [/<encoding parameters>]
a= rtpmap : 0 PCMU / 8000
a= X-pc-codecs: <alternative 1> <alternative 2> ...
a= X-pc-secret: <method>:<encryption key>
a= X-pc-csuites-rtp: <alternative 1> <alternative 2> ...
a= X-pc-csuites-rtcp: <alternative 1> <alternative 2> ...
a= X-pc-spi-rtcp: <value>
a= X-pc-bridge: <number-ports>
a= <attribute>
a= recvonly
a= sendrecv
a= sendonly
a=ptime
```

Emisión: SE PUEDEN incluir una o más de las líneas de atributos "a" que se especifican más adelante. NO SE DEBERÍA utilizar una línea de atributos que no figure entre las especificadas más adelante.

Recepción: SE PUEDEN incluir una o más de las líneas de atributos "a" especificadas más adelante y SE DEBE actuar sobre ellas en consecuencia. Pueden estar presentes líneas de atributos "a" no especificadas más adelante pero DEBEN ser ignoradas.

rtpmap:

Emisión: El campo DEBE utilizarse de conformidad con RFC 2327. PUEDE utilizarse tanto con códecs muy conocidos como con códecs no muy conocidos. Los nombres de codificación utilizados se proporcionan en una especificación IPCablecom separada.

Recepción: El campo DEBE utilizarse de conformidad con RFC 2327.

X-pc-codecs:

Emisión: El campo contiene una lista de códecs alternativos que el punto extremo es capaz de utilizar para esta conexión. La lista está ordenada según el grado decreciente de preferencia, es decir el códec alternativo más preferido es el primero de la lista. Un códec se codifica de manera similar al "nombre de codificación" en rtpmap.

Recepción: Transporta una lista de códecs que el punto extremo distante es capaz de utilizar para esta conexión. Los códecs NO DEBEN utilizarse hasta que sean señalados a través de una línea de medios (m=).

X-pc-secret:

Emisión: El campo contiene un secreto extremo a extremo para su utilización con la seguridad RTP y RTCP. El secret (secreto) se codifica de manera similar al parámetro (k=) clave de criptación de RFC 2327 con las siguientes limitaciones:

- La clave de criptación NO DEBE contener una ciphersuite sino solamente una frase de paso.
- El <method> que especifica la codificación de la frase de paso DEBE estar, bien en texto claro ("clear") o bien en "base64" como se define en RFC 2045, salvo para la longitud de línea máxima, que no se especifica aquí. El método "clear" NO DEBE utilizarse si el secreto contiene algunos caracteres que están prohibidos en SDP.

Recepción: Transporta el secreto de extremo a extremo para su utilización con la seguridad RTP y RTCP.

X-pc-csutes-rtp:
X-pc-csutes-rtcp:

Emisión: El campo contiene una lista de ciphersuites que el punto extremo es capaz de utilizar para esta conexión (respectivamente RTP y RTCP). La primera ciphersuite listada es la que el punto extremo está actualmente esperando utilizar. Cualesquiera ciphersuites restantes de la lista representan alternativas ordenadas por grado decreciente de preferencia, es decir, la ciphersuite alternativa más preferida es la segunda de la lista. Una ciphersuite 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 RFC 2234. No se permiten espacios en blanco dentro de una ciphersuite. El ejemplo siguiente ilustra el uso de la ciphersuite:

62/51

La lista real de ciphersuites que se proporcionará en la Especificación UIT-T J.170 de seguridad de IPCablecom.

Recepción: Transporta una lista de ciphersuites que el punto extremo distante es capaz de utilizar para esta conexión. Cualquier otra ciphersuite distinta de la primera de la lista no se puede utilizar hasta que sea señalada por medio de una nueva línea de ciphersuites con la ciphersuite deseada listada en primer lugar.

X-pc-spi-rtcp:

Emisión: El campo contiene el índice de parámetro de seguridad (SPI, *security parameter index*) que se utilizará al enviar paquetes RTCP al punto extremo para el tren de medios en cuestión. El SPI es un identificador de 32 bits codificado como una cadena de hasta 8 caracteres hexadecimales. El campo DEBE suministrarse cuando se utiliza la seguridad RTCP.

Recepción: Transporta el SPI RTCP que ha de utilizarse cuando se envían paquetes RTCP por IPSEC. El campo DEBE estar presente cuando se utiliza la seguridad RTCP.

X-pc-bridge:

Emisión: Los puntos extremos NCS NO DEBEN utilizar este atributo.

Recepción: Los puntos extremos NCS DEBEN ignorar este atributo en caso de recibirlo.

recvonly:

Emisión: El campo DEBE utilizarse de conformidad con RFC 2543.

Recepción: El campo DEBE utilizarse de conformidad con RFC 2543.

sendrecv:

Emisión: El campo DEBE utilizarse de conformidad con RFC 2543.

Recepción: El campo DEBE utilizarse de conformidad con RFC 2543.

sendonly:

Emisión: El campo DEBE utilizarse de conformidad con RFC 2543, salvo que la dirección IP y el número de puerto NO DEBEN ser cero.

Recepción: El campo DEBE utilizarse de conformidad con RFC 2543.

ptime:

Emisión: El campo ptime SE DEBERÍA proporcionar siempre, y en el caso de que se utilice DEBE hacerse de conformidad con RFC 2327. Cuando se utiliza un campo rtpmap o un códec no bien conocido, SE DEBE proporcionar el campo ptime.

Recepción: El campo SE DEBE utilizar de conformidad con RFC 2327. Cuando está presente el campo ptime, el MTA DEBE utilizarlo en el cálculo de las reservas QoS. Si no está presente "ptime", el MTA DEBE suponer valores por defecto razonables para los códecs bien conocidos.

7.4.1.12 Avisos de medios (m=)

Los avisos de medios (m=) están formados por 3 subcampos:

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

Media (medios):

Emisión: SE DEBE utilizar el tipo de medios "audio".

Recepción: El tipo de medios recibido DEBE ser "audio".

Port (puerto):

Emisión: DEBE llenarse de conformidad con RFC 2327. El puerto especificado es el puerto de recepción, con independencia de si el tren es unidireccional o bidireccional. El puerto de emisión puede ser diferente.

Recepción: DEBE utilizarse de conformidad con RFC 2327. El puerto especificado es el puerto de recepción. El puerto de emisión puede ser diferente.

Transport (transporte):

Emisión: SE DEBE utilizar el protocolo de transporte "RTP/AVP".

Recepción: El protocolo de transporte DEBE ser "RTP/AVP".

Media Formats (Formatos de medios):

Emisión: SE DEBE utilizar el tipo de medios apropiado definido en RFC 2327.

Recepción: De conformidad con RFC 2327.

7.4.2 Utilización del servicio vídeo SDP

La información detallada relativa a la utilización del SDP para el servicio vídeo será objeto de ulterior estudio.

7.5 Transmisión por UDP

7.5.1 Entrega de mensajes fiable

Los mensajes MGCP se transmiten por el protocolo de datagramas de usuario (UDP). Las instrucciones se envían a una de las direcciones IP definidas en el sistema de nombres de dominio (DNS) para el agente de llamada o punto extremo especificado. Las respuestas se envían hacia atrás a la dirección de fuente de la instrucción. Sin embargo, debe señalarse que la respuesta puede, de hecho, proceder de otra dirección IP distinta de la dirección a la que se envió la instrucción.

Cuando no se proporciona ningún puerto para el punto extremo³⁰, la instrucción debe enviarse al puerto MGCP por defecto, 2427.

³⁰ Cada punto extremo se puede proporcionar con un puerto y dirección de agente de llamada separado.

Los mensajes MGCP cursados por UDP pueden experimentar pérdidas. En ausencia de una respuesta oportuna en el tiempo se repiten las instrucciones. Se espera que las entidades MGCP conserven en memoria una lista de las respuestas enviadas a transacciones recientes, a saber, una lista de todas las respuestas enviadas en los últimos T_{hist} segundos, así como una lista de las transacciones que se están ejecutando actualmente. Los identificadores de transacción de las instrucciones entrantes se comparan con los identificadores de transacción de las respuestas recientes. Si existe concordancia, la entidad MGCP no ejecuta la transacción sino que sencillamente repite la respuesta. Si no se encuentra concordancia, la entidad MGCP examina la lista de transacciones actualmente en ejecución. Si se encuentra concordancia, la entidad MGCP no ejecutará la transacción, la cual se ignora sencillamente.

Compete a la entidad solicitante proporcionar las temporizaciones adecuadas para todas las instrucciones excepcionales y repetir las instrucciones cuando las temporizaciones se han sobrepasado. En 7.5.2 se especifica una estrategia de retransmisión.

Además, cuando las instrucciones repetidas fracasan en la consecución de una respuesta, se supone que la entidad de destino se encuentra indisponible. Corresponde a la entidad solicitante la búsqueda de servicios redundantes y/o la liberación de conexiones existentes o pendientes como se especifica en 6.4.

7.5.2 Estrategia de retransmisión

Esta Recomendación evita especificar alguno de los valores estáticos para los temporizadores de retransmisión puesto que estos valores dependen típicamente de la red. Normalmente, los temporizadores de retransmisión deben calcular la temporización mediante la medición del tiempo transcurrido entre la emisión de una instrucción y el retorno de una respuesta. Los clientes incorporados DEBEN implementar una estrategia de retransmisión que utilice una reducción exponencial con valores de las temporizaciones de retransmisión máximo e inicial configurables.

Los clientes incorporados DEBERÍAN utilizar el algoritmo implementado en TCP-IP, el cual hace uso de estas dos variables:

- El retardo de acuse de recibo medio (AAD, *average acknowledgement delay*), estimado como un valor medio suavizado exponencialmente de los retardos observados.
- La desviación media (ADEV, *average deviation*), estimada como un valor medio suavizado exponencialmente del valor absoluto de la diferencia entre el retardo observado y el valor medio actual.

El temporizador de retransmisión (RTO) en TCP, se fija a la suma del retardo medio más N veces la desviación media, siendo N una constante.

Después de cualquier retransmisión, la entidad MGCP debe realizar lo siguiente:

- doblar el valor estimado del retardo medio, AAD;
- calcular un valor aleatorio, uniformemente distribuido entre 0,5 AAD y AAD;
- fijar el temporizador de retransmisión (RTO, *retransmission timer*) al valor mínimo de:
 - la suma de dicho valor aleatorio y N veces la desviación media.
- RTO_{max} , donde el valor por defecto de RTO_{max} es 4 segundos.

Este procedimiento tiene dos efectos: Como incluye una componente exponencialmente creciente, disminuirá automáticamente la velocidad del tren de mensajes en caso de congestión de acuerdo con las necesidades de comunicación en tiempo real. Como incluye una componente aleatoria, interrumpirá la posible sincronización entre notificaciones activadas por el mismo evento externo.

El valor inicial utilizado por defecto para el temporizador de retransmisión es de 200 milisegundos, y el valor máximo por defecto para el temporizador de retransmisión es de 4 segundos por defecto. El proceso de aprovisionamiento puede alterar estos valores por defecto.

7.6 Transporte en remolque

En algunos casos el agente de llamada deseará enviar varios mensajes al mismo tiempo a uno o más puntos extremos de una pasarela, y viceversa. Cuando hay que enviar varios mensajes en los mismos paquetes UDP, los mensajes se separan por una línea de texto que contiene un punto aislado, como por ejemplo:

```
200 2005 OK
```

```
DLCX 1244 aaln/2@rgw.whatever.net MGCP 1.0 NCS 1.0  
C: A3C47F21456789F0  
I: FDE234C8
```

Los mensajes remolcados SE DEBEN procesar como si se hubieran recibido en datagramas separados, si bien cuando se ha de retransmitir un mensaje (instrucción o respuesta), SE DEBE retransmitir el datagrama entero, y no solamente el mensaje perdido. Los mensajes individuales en el datagrama SE DEBEN procesar comenzando por el primer mensaje.

Los errores encontrados en un mensaje que ha sido transportado en remolque, NO DEBE afectar a ninguno de los demás mensajes recibidos en tal paquete – cada mensaje es procesado solo.

7.7 Identificadores de transacción y toma de contacto de tres caminos

Los identificadores de transacción son números enteros comprendidos en la gama de 1 a 999 999 999. Los agentes de llamada puede decidir el uso de un espacio numérico específico para cada una de las pasarelas que ellos gestionan, o utilizar el mismo espacio numérico para todas las pasarelas que pertenecen a algún grupo arbitrario. Los agentes de llamada pueden decidir compartir la carga de gestión de una pasarela grande entre varios procesos independientes. Estos procesos compartirán el mismo espacio numérico de transacción. Existen múltiples implementaciones posibles de esta compartición, como la implementación que dispone de una asignación centralizada de los identificadores de transacción o la que preasigna gamas no superpuestas de identificadores a procesos diferentes. Las implementaciones DEBEN garantizar que se asignan identificadores de transacción exclusivos a todas las transacciones que se originan en cualquier agente de llamada y son enviadas a una pasarela concreta dentro de un periodo de $T_{t_{hist}}$ segundos. Las pasarelas pueden detectar sencillamente las transacciones duplicadas con la mera observación del identificador de transacción.

El parámetro acuse-de-recibo-de-respuesta se puede encontrar en cualquier instrucción. El parámetro transporta un conjunto de "gamas de identificadores de transacción confirmadas" para las respuestas definitivas recibidas – NO SE DEBEN confirmar las respuestas provisionales.

Las pasarelas MGCP pueden decidir suprimir las copias de las respuestas a transacciones cuyo identificador esté incluido en las "gamas de identificadores de transacción confirmadas" recibidas en un mensaje, si bien por el hecho de que la transacción haya sido ejecutada DEBEN no obstante ser retenidas durante $T_{t_{hist}}$ segundos. Asimismo, cuando se recibe un mensaje de acuse de recibo de respuesta³¹, la respuesta de la que el mensaje está acusando recibo puede suprimirse. Las pasarelas deben descartar sin más otras instrucciones procedentes de ese agente de llamada cuando el id de transacción se encuentra dentro de estas gamas y la respuesta ha sido emitida hace menos de $T_{t_{hist}}$ segundos.

Sean $term_{new}$ y $term_{old}$ el nombre de punto extremo, respectivamente en una instrucción nueva, cmd_{new} , y en alguna instrucción antigua, cmd_{old} . Los ids de transacción que se han de confirmar en la cmd_{new} DEBERÍAN entonces determinarse como sigue:

³¹ En vez de una instrucción con un parámetro Response Acknowledgement.

- 1) Si $term_{new}$ no contiene ningún comodín:
 - a) Respuestas no confirmadas a instrucciones antiguas donde $term_{old}$ es igual a $term_{new}$.
 - b) Opcionalmente, una o más respuestas no confirmadas, donde $term_{old}$ contiene el comodín "any-of" ("cualquiera de"), y el nombre de punto extremo devuelto en la respuesta ha sido $term_{new}$.
 - c) Opcionalmente, una o más respuestas no confirmadas donde $term_{old}$ contiene el comodín "all" ("todos") y $term_{new}$ está cubierto por el comodín en $term_{old}$.
 - d) Opcionalmente, una o más respuestas no confirmadas donde $term_{old}$ contiene el comodín "any-of" ("cualquiera de"), no se ha devuelto ningún nombre de punto extremo y $term_{new}$ está cubierto por el comodín en $term_{old}$.
- 2) Si $term_{new}$ contiene el comodín "all" ("todos"):
 - a) Opcionalmente, una o más respuestas no confirmadas donde $term_{old}$ contiene el comodín "all" ("todos") y $term_{new}$ está cubierto por el comodín en $term_{old}$.
- 3) Si $term_{new}$ contiene el comodín "any of" ("cualquiera de"):
 - a) Opcionalmente, una o más respuestas no confirmadas donde $term_{old}$ contiene el comodín "all" ("todos") y $term_{new}$ está cubierto por el comodín en $term_{old}$ si el comodín "any of" ("cualquiera de") en $term_{new}$ ha sido sustituido por el comodín "all" ("todos").

Una respuesta determinada NO SE DEBERÍA confirmar en dos mensajes separados.

Los ejemplos siguientes ilustran la aplicación de estas reglas:

- Si $term_{new}$ es "aaln/1" y $term_{old}$ es "aaln/1" la respuesta antigua se puede confirmar por la regla 1a.
- Si $term_{new}$ es "aaln/1" y $term_{old}$ es "*" la respuesta antigua se puede confirmar por la regla 1c.
- Si $term_{new}$ es "aaln/*" y $term_{old}$ es "*" la respuesta antigua se puede confirmar por la regla 2a.
- Si $term_{new}$ es "aaln/\$" y $term_{old}$ es "aaln/*" la respuesta antigua se puede confirmar por la regla 3a.

Los valores de las "gamas de identificadores de transacción confirmadas" NO DEBERÍA utilizarse si han transcurrido más de T_{hist} segundos desde que la pasarela emitió su última respuesta hacia ese agente de llamada, o cuando una pasarela reanuda la operación. En esta situación, se deben aceptar y procesar las instrucciones sin efectuar ninguna prueba del identificador de transacción.

Además, NO SE DEBE confirmar un respuesta si se ha recibido después de transcurridos más de T_{hist} segundos.

Los mensajes que confirman respuestas se pueden transmitir y recibir sin orden. La pasarela deberá retener la unión de los ids de transacción confirmados que se han recibido en instrucciones recientes.

7.8 Respuestas provisionales

En algunos casos los tiempos de compleción de las transacciones pueden ser significativamente más grandes que en otros³². La NCS utiliza el UDP como protocolo de transporte, y la fiabilidad se consigue mediante retransmisiones basadas en temporizaciones selectivas estimadas como la suma

³² Por ejemplo cuando los recursos se reservan y comprometen externamente como parte de una transacción.

del tiempo de ida y retorno de la red y el tiempo de compleción de la transacción. Una variación importante en el tiempo de compleción de la transacción resulta por consiguiente problemática cuando se desea la detección rápida de la pérdida de mensajes sin una tara excesiva.

Para resolver este problema, SE DEBE por tanto enviar una respuesta provisional si, y sólo si, el tiempo de compleción de la transacción excede de un periodo de tiempo pequeño. La respuesta provisional acusa recibo de la recepción de la instrucción aunque el resultado de la instrucción posiblemente no se pueda conocer todavía a causa, por ejemplo, de que está pendiente una reserva de recursos. A título orientativo, una transacción que requiere una comunicación externa para completarse, por ejemplo una reserva de recursos de red, debe emitir una respuesta provisional. Además, si se recibe una instrucción CreateConnection o ModifyConnection duplicada, y la transacción no ha finalizado todavía su ejecución, DEBE enviarse hacia atrás un respuesta provisional.

Una semántica transaccional pura implicaría que las respuestas provisionales no deberían devolver ninguna otra información diferente del hecho de que la transacción se está actualmente ejecutando, si bien un enfoque optimista que permita devolver alguna información posibilita una reducción del retardo en que de otro modo incurriría el sistema.

Las respuestas provisionales sólo SE DEBEN enviar en contestación a una instrucción CreateConnection o ModifyConnection. Para reducir el retardo en el sistema, SE DEBE incluir en la respuesta provisional a la instrucción CreateConnection un identificador de conexión y una descripción de sesión. Si una descripción de sesión es devuelta por la instrucción ModifyConnection, la descripción de sesión SE DEBE incluir también aquí en la respuesta provisional. Si la transacción se completa con éxito, la información devuelta en la respuesta provisional SE DEBE repetir en la respuesta definitiva. Se considera un error de protocolo no repetir esta información o cambiar alguna de las informaciones suministradas con anterioridad en una respuesta definitiva. Si la transacción falla, se devuelve un código de error – la información devuelta anteriormente no es válida en lo sucesivo.

Una transacción CreateConnection o ModifyConnection que se esté ejecutando actualmente DEBE ser cancelada si se recibe una instrucción DeleteConnection para el punto extremo. En ese caso, SE DEBERÍA sin embargo devolver automáticamente una respuesta para la transacción cancelada y SE DEBE devolver una respuesta para la transacción cancelada en el caso de que se detecte una retransmisión de la transacción cancelada.

Cuando se recibe una respuesta provisional, el periodo de temporización para la transacción en cuestión DEBE ser fijado a un valor significativamente superior ($T_{t_{longtran}}$). El propósito de este temporizador es en primer lugar detectar el fallo del punto extremo. El valor por defecto de $T_{t_{longtran}}$ es de 5 segundos, si bien el proceso de aprovisionamiento puede alterar este valor.

Cuando la transacción finaliza su ejecución, se envía la respuesta definitiva y se elimina la respuesta provisional ahora obsoleta. Con el fin de asegurar la detección rápida de una respuesta definitiva perdida, SE DEBE acusar recibo de las respuestas definitivas emitidas después de las respuestas provisionales para una transacción. El punto extremo DEBE incluir por tanto un parámetro "ResponseAck" vacío en aquellas, y solamente aquellas, respuestas definitivas. La presencia del parámetro "ResponseAck" en la respuesta definitiva desencadenará el envío hacia atrás al punto extremo de la respuesta "acuse de recibo de respuesta". La respuesta "acuse de recibo de respuesta" incluirá el id de transacción de la respuesta de la que se acusa recibo en el encabezamiento de respuesta. La recepción de esta respuesta de "acuse de recibo de respuesta" está sujeta a las mismas estrategias y procedimientos de temporización y retransmisión que las respuestas a instrucciones (véase 6.4), es decir, el emisor de la respuesta definitiva retransmitirá ésta si no se recibe en tiempo el "acuse de recibo de respuesta". Nunca se acusa recibo de la respuesta "acuse de recibo de respuesta".

8 Seguridad

Si entidades no autorizadas pudieran utilizar el MGCP, ellas podrían establecer llamadas no autorizadas o interferir con llamadas autorizadas. La seguridad no se proporciona como parte integrante de MGCP. En su lugar, el MGCP supone la existencia de una capa inferior que proporciona la seguridad real.

Los requisitos y soluciones de seguridad para la NCS se proporcionarán en la Especificación UIT-T J.170 de seguridad de IPCablecom, la cual se debe consultar para más información.

ANEXO A

Lotes de eventos

En este anexo se define un conjunto inicial de lotes de eventos para varios tipos de puntos extremos definidos actualmente por IPCablecom para clientes incorporados. Se definen los siguientes lotes para los tipos de punto extremo de clientes incorporados que se señalan:

Tipo de punto extremo	Lote	Nombre de lote	Lote por defecto
Línea de acceso analógica	Base	B	Ninguno
Vídeo	<i>En estudio</i>	<i>En estudio</i>	<i>En estudio</i>
BRI RDSI	<i>En estudio</i>	<i>En estudio</i>	<i>En estudio</i>

Cada lote define un nombre de lote para el lote y definiciones y códigos de eventos para cada uno de los eventos del lote. Las tablas de eventos/señales de cada lote tienen cinco columnas:

Código	Código de evento exclusivo del lote utilizado para el evento/señal.
Descripción	Descripción breve del evento/señal.
Evento	En esta columna aparece una señal de comprobación si el evento puede ser solicitado por el controlador de pasarela de medios. Alternativamente, pueden aparecer uno más de los símbolos siguientes: "P" que indica que el evento es persistente, "S" que indica que el evento/señal se encuentra en un estado de evento que puede ser auditado, "C" que indica que el evento/señal puede ser detectado/aplicado en una conexión.
Señal	Si no aparece nada en esta columna para un evento determinado, el evento no puede ser señalizado a petición por el controlador de pasarela de medios. En caso contrario, se identificará el tipo de evento por los siguientes símbolos: "OO" Señal On/Off (activado/desactivado). La señal está en la posición de activado ("on") hasta que el controlador de pasarela de medios la conmuta a la posición de desactivado ("off"), y viceversa. "TO" Señal Time-Out (de temporización). La señal permanece durante un tiempo dado a menos que sea reemplazada por una nueva señal. Se suministran valores por defecto de la temporización. Un valor cero indica que el periodo de temporización es infinito. El proceso de aprovisionamiento puede alterar estos valores por defecto.

"BR" Señal Brief (breve). El evento tiene una duración corta y conocida.

Información adicional Proporciona información acerca del evento/señal, por ejemplo, la duración por defecto de las señales TO.

A menos que se establezca otra cosa, todos los eventos/señales son detectados/aplicados en puntos extremos y la información audio generada por ellos no es reenviada en ninguna conexión que el punto extremo pueda tener. La información audio generada por eventos/señales que son detectados/aplicados en una conexión será no obstante reenviada en la conexión asociada con independencia del modo de conexión.

Lotes del protocolo de base

En el protocolo de base se definen por lo general los siguientes lotes. Estos lotes se aplican a todos los puntos extremos:

- Base

Lote base

Nombre del lote: B

Para identificar los eventos y señales del lote "base" para todos los tipos de punto extremo se utilizan los siguientes códigos:

Código	Descripción	Evento	Señal	Información adicional
oc	Operación completada	√	–	
of	Operación fracasada	√	–	

Operación completada (oc, *operation complete*): El evento operación completada se genera cuando se ha pedido a la pasarela que aplique una o varias señales del tipo TO en el punto extremo, y una o más de esas señales se han completado sin que sean detenidas por la detección de un evento solicitado tal como la transición a descolgado o la marcación de dígitos. El informe de compleción puede transportar como parámetro el nombre de la señal que llegó al final de su vida, como en:

O: B/oc(mypackage/mysignal)

Cuando la señal informada se ha aplicado en una conexión, el parámetro proporcionado incluirá también el nombre de la conexión, como en:

O: B/oc(mypackage/mysignal@0A3F58)

Cuando se pide el evento operación completada, éste puede ser parametrizado con cualesquiera parámetros de evento. Cuando se omite el nombre de lote, se adopta el nombre de lote por defecto

El evento operación completada puede generarse además del modo definido en el protocolo de base, por ejemplo, cuando se completa con éxito una instrucción ModifyConnection incorporada, como en:

O: B/oc(B/C)

Operación fracasada (of, *operation failure*): Por lo general, el evento operación fracasada se puede generar cuando se ha pedido al punto extremo que aplique una o varias señales de tipo TO en el punto extremo, y una o más de esas señales han fracasado antes de expirar la temporización. El informe de compleción puede transportar como parámetro el nombre de la señal que ha fallado, como en:

O: B/of(mypackage/mysignal)

Cuando se ha aplicado en una conexión la señal informada, el parámetro suministrado incluirá también el nombre de la conexión, como en:

```
O: B/of(mypackage/mysignal@0A3F58)
```

Cuando se ha solicitado el evento operación fracasada, los parámetros del evento no se pueden especificar. Cuando se omite el nombre del lote, se adopta el nombre de lote por defecto.

El evento operación fracasada se puede generar además en el protocolo de base, por ejemplo, cuando falla una instrucción ModifyConnection incorporada, como en:

```
O: B/of(B/C(M(sendrecv(AB2354))))
```

Audio

Los lotes de eventos para audio quedan en estudio.

Vídeo

Los lotes de eventos para vídeo quedan en estudio.

RDSI

Los lotes de eventos para el acceso básico RDSI quedan en estudio.

ANEXO B

Calidad de servicio dinámica

En este anexo, se proporcionan más detalles sobre la utilización de la calidad de servicio dinámica (D-QoS, *dynamic quality-of-service*) en NCS. Describimos más detalladamente el comportamiento MTA esperado e incluimos una máquina de estado que el MTA puede implementar para soportar el comportamiento D-QoS descrito. La especificación de la calidad de servicio dinámica de IPCablecom (UIT-T J.163) debe consultarse para detalles adicionales.

Introducción

La implementación del MTA para soportar la calidad de servicio dinámica debe almacenar y mantener el estado de la D-QoS para cada conexión. Cada vez que se utilice la D-QoS para una conexión, el punto extremo guardará la siguiente información D-QoS asociada con la conexión hasta que sea eliminada:

- **GateID (identificador de puerta)** – El GateID actual utilizado para la conexión.
- **ResourceID (identificador de recursos)** – El ResourceID actual utilizado para la conexión.
- **Last reservation (última reserva)** – Los parámetros de la reserva más reciente para la conexión. Incluye clasificadores y parámetros de medios en ambos sentidos, de emisión y de recepción.
- **Last commit (ultimo compromiso)** – Los parámetros del compromiso más reciente para la conexión. Incluye clasificadores y parámetros de medios en ambos sentidos, de emisión y de recepción.
- **Reserve Destination (destino de reservas)** – Un puerto y dirección IP que puede utilizarse para posibilitar las reservas de recursos en el caso en que la información de dirección distante no es todavía conocida como se explica más adelante.
- **Gate Location (ubicación de puerta)** – El puerto o dirección IP a donde debe enviarse el mensaje compromiso de D-QoS cuando se utiliza RSVP. El MTA aprende esta dirección a través de los mensajes QoS RSVP.

El GateID es la clave para la reserva de recursos. Una vez que se ha proporcionado el GateID para una conexión, se crea una máquina de estado D-QoS para la misma, y toda la información anterior se mantendrá para la conexión hasta que ésta sea suprimida, o se proporciona una nueva GateID. En el último caso, se reinicia la máquina de estado D-QoS y la información anterior, y se elimina la reserva antigua³³.

El MTA puede comprometer y reservar los recursos para los sentidos de emisión y de recepción por separado. El puerto y dirección IP de destino en emisión así como la dirección IP de la fuente se toman del RemoteConnectionDescriptor, cuando éste ha sido proporcionado. En tal caso, el MTA DEBE utilizar los siguientes clasificadores para la reserva y compromiso de recursos:

	MTA-o (J.112/RSVP)
Descendente/recepción	
IP de la fuente	IP(SDP-t)
Puerto de la fuente	*
IP del destino	IP(SDP-o)
Puerto del destino	Port(SDP-o)
Ascendente/emisión	
IP de la fuente	IP(SDP-o)
Puerto de la fuente	Port(o)
IP del destino	IP(SDP-t)
Puerto del destino	Port(SDP-t)

donde:

- **IP(SDP-o)** indica la dirección IP de medios en MTA-o's LocalConnectionDescriptor.
- **IP(SDP-t)** indica la dirección IP de medios en MTA-o's RemoteConnectionDescriptor.
- **Port(SDP-o)** indica el puerto de medios en MTA-o's LocalConnectionDescriptor.
- **Port(o)** indica el puerto de fuente MTA-o que se utilizará cuando se envíen medios por esta conexión. Obsérvese que este puerto puede ser el mismo o distinto que el Port(SDP-o).

Cuando no se ha proporcionado todavía un RemoteConnectionDescriptor, el puerto y dirección IP de destino en emisión real es desconocido y se utiliza por tanto en su lugar la dirección ReserveDestination. Para el sentido de recepción, el puerto y dirección IP de fuente será sustituido por un comodín. Esto permite una reserva y un compromiso del recurso de recepción en el enlace de acceso. DEBEN utilizarse los siguientes clasificadores:

³³ Obsérvese que si está incluido un ResourceID y este ResourceID concuerda con el ResourceID antiguo, no debe suprimirse la reserva antigua antes de que se realice la nueva.

	MTA-o (J.112/RSVP)
Descendente/recepción	
IP de la fuente	*
Puerto de la fuente	*
IP del destino	IP(SDP-o)
Puerto del destino	Port(SDP-o)
Ascendente/emisión	
IP de la fuente	IP(SDP-o)
Puerto de la fuente	Port(o)
IP del destino	IP(RD-o)
Puerto del destino	Port(RD-o)

donde:

- **IP(RD-o)** indica la dirección IP en el ReserveDestination suministrado.
- **IP(Port-o)** indica el número de puerto en el ReserveDestination suministrado. Si no se especifica ningún número de puerto se aplica un valor por defecto de 9.
- Una vez que se conocen el puerto y las direcciones de medios de destino en emisión y de fuente en recepción, se actualizarán las reservas con los clasificadores adecuados.
- Cuando se utiliza RSVP como protocolo de reserva de recursos, la dirección de destino utilizada para el mensaje PATH RSVP será la dirección IP ReserveDestination suministrada hasta que se facilita un RemoteConnectionDescriptor.

Máquina de estado NCS/D-QoS

Como se ha expuesto anteriormente, el MTA mantiene el estado de la calidad de servicio dinámica utilizada en una conexión. El estado se obtiene de una máquina de estado que está gobernada por lo siguiente:

- **Current state (estado vigente)** que consiste en la pareja de estados (SendQoSState, ReceiveQoSState), donde cada uno de los estados QoS puede ser uno de los siguientes:
 - **N** – No existe ninguna reserva de recursos para el sentido en cuestión.
 - **R** – Existe una reserva de recursos para el sentido en cuestión, pero no hay ningún recurso actualmente comprometido.
 - **C** – Existe una reserva de recursos para el sentido en cuestión, estando algunos recursos actualmente comprometidos.
 - **Connection mode (modo de conexión)** que es el modo de conexión de NCS. Los modos de conexión "Conference", "Network Loopback", y "Network Continuity Test" no se muestran explícitamente en la máquina de estado puesto que son similares a "SendReceive". El modo de conexión "Replicate" tampoco se muestra por ser similar a "SendOnly".
- **Resource Change** (cambio de recurso), que es uno o más de los siguientes:
 - Cambia el puerto o dirección IP RemoteConnectionDescriptor (se necesita actualizar el clasificador). Incluye el caso en que esto ocurre por primera vez.
 - Cambia el códec.
 - Cambia el Ptime.
 - etc.
- Las **D-QoS rules (reglas D-QoS)** proporcionadas en 6.3.3.

Como se ha explicado anteriormente, la máquina de estado se reinicializará cuando se reciba un nuevo GateID. Si se suministra también un ResourceID y este identificador es el mismo que el ResourceID antiguo, la(s) reserva(s) para la nueva máquina de estado DEBE(N) realizarse antes de que la(s) reserva(s) para la máquina de estado antigua se haya(n) liberado.

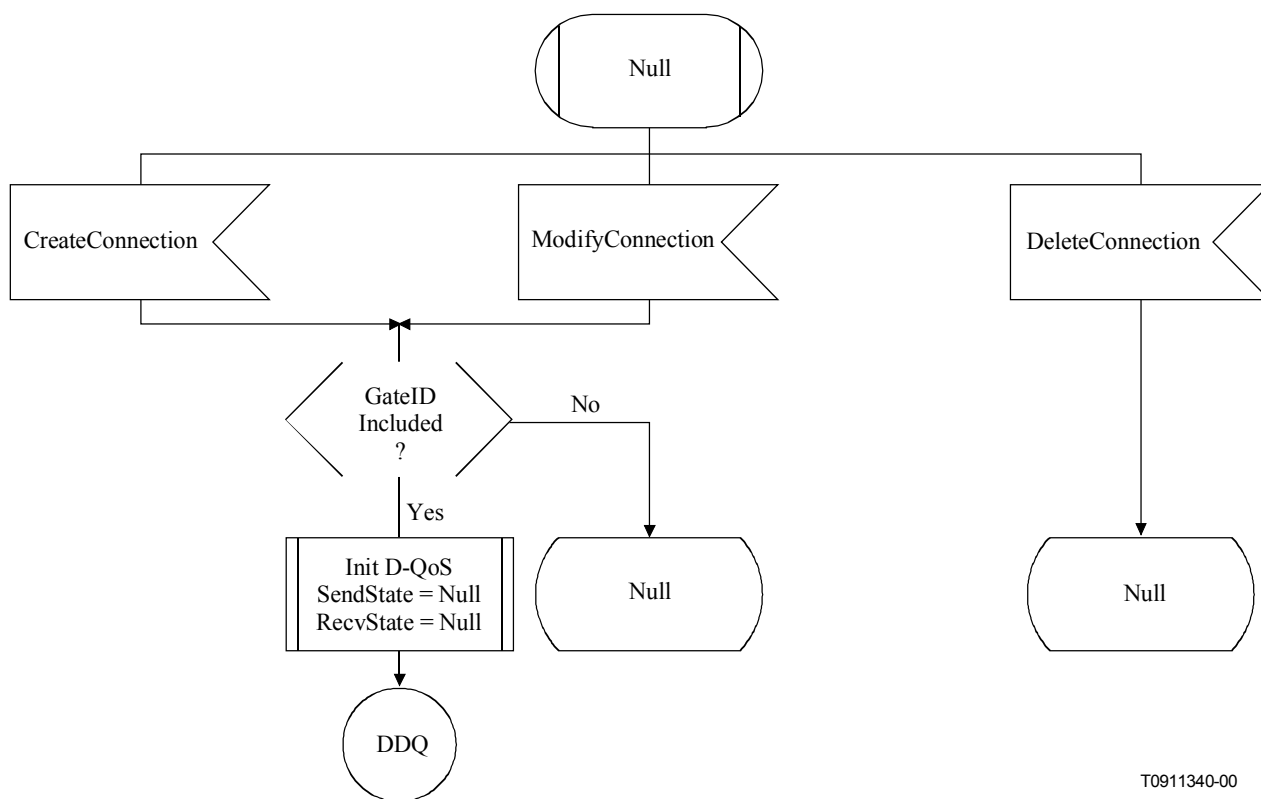
El conjunto de *estados* posibles es:

- (N, N) Emisión de recursos no reservados, recepción de recursos no reservados.
- (R, R) Emisión de recursos reservados, recepción de recursos reservados.
- (C, R) Emisión de recursos reservados y comprometidos, recepción de recursos reservados.
- (R, C) Emisión de recursos reservados, recepción de recursos reservados y comprometidos.
- (C, C) Emisión de recursos reservados y comprometidos, recepción de recursos reservados y comprometidos.
- (R, N) Emisión de recursos no reservados, recepción de recursos no reservados.
- (C, N) Emisión de recursos reservados y comprometidos, recepción de recursos no reservados.
- (N, R) Emisión de recursos no reservados, recepción de recursos reservados.
- (N, C) Emisión de recursos no reservados, recepción de recursos reservados y comprometidos.

Una vez que se han reservado y/o comprometido recursos para una dirección, la reserva para esta dirección existirá durante el tiempo de vida de la conexión. La relación entre estados y modo de conexión o parámetros de reserva D-QoS se muestra en el cuadro a continuación:

	SendState	RecvState
Ningún parámetro reservado/comprometido suministrado – modo de conexión:		
inactive	R	R
sendonly, replcate	C	R
recvonly	R	C
sendrecv, confrnce, netwloop, netwtest	C	C
Parámetro reservado/comprometido suministrado:		
sendresv	R	N, R*
recvresv	N, R*	R
snrcresv	R	R
sendcomt	C	N, R*
recvcomt	N, R*	C
snrccomt	C	C
* Si se han reservado o comprometido recursos anteriormente para la dirección, el estado será R; en caso contrario el estado será N.		

En la figura B.1 se representa el diagrama de transición de estados real:



T0911340-00

Figura B.1/J.162 – Diagrama de estados NCS/D-QoS (hoja 1 de 2)

Cuando se ejecuta la máquina de estado, se fijarán variables booleanas para indicar si se han de realizar operaciones de reserva, anulación de reserva, compromiso y anulación de compromiso. El pseudocódigo que se da a continuación proporciona entonces detalles sobre los procedimientos D-QoS individuales que han de ejecutar estas variables booleanas. Las *acciones* siguientes especifican las acciones D-QoS que han de emprenderse en cada uno de estos procedimientos:

- **SR** indica que se efectuará una reserva en emisión de D-QoS.
- **RR** indica que se efectuará una reserva en recepción de D-QoS.
- **SC** indica que se establecerá un compromiso en emisión de D-QoS.
- **RC** indica que se establecerá un compromiso en recepción de D-QoS.
- **SD** indica que se anulará una reserva en emisión de D-QoS.
- **RD** indica que se anulará una reserva en recepción de D-QoS.
- **SU** indica que se anulará un compromiso en emisión de D-QoS, es decir, se bajarán a cero los recursos en emisión comprometidos.
- **RU** indica que se anulará un compromiso en recepción de D-QoS, es decir, se bajarán a cero los recursos en recepción comprometidos.

SendReserve()

```
If <current resources reserved ≠ resources to reserve> then {
    -- skip reservation if existing reservation OK
    If <RemoteConnectionDescriptor provided> then
        SR(RemoteConnectionDescriptor)
        -- Use RemoteConnectionDescriptor classifier
    else if <ReserveDestination provided> then
        SR(ReserveDestination)
        -- Use ReserveDestination classifier, send to
        -- ReserveDestination if RSVP
    else ERROR
}
```

ReceiveReserve()

```
If <current resources reserved ≠ resources to reserve> then {
    -- skip reservation if existing reservation OK
    If <RemoteConnectionDescriptor provided> then
        RR(RemoteConnectionDescriptor)
        -- Use RemoteConnectionDescriptor classifier
    else if <(J.112 QoS) or (RSVP and ReserveDestination provided)>
        then RR(*)
        -- Use wildcard classifier, send to
        -- ReserveDestination if RSVP
    else ERROR
}
```

SendCommit()

```
If <current resources committed ≠ resources to commit> then {
    -- skip commit if existing OK
    If <RemoteConnectionDescriptor provided> then {
        If not <resources to commit ⊂ resources reserved > then {
            -- old reservation does not satisfy what is about to be
            -- committed, so update reservation
            SR(RemoteConnectionDescriptor)
        }
    }
    if <(J.112 QoS) or (RSVP and ReserveDestination provided)> then {
        SC(RemoteConnectionDescriptor)
        -- send to ReserveDestination if RSVP
    }
}
```

```

    } else ERROR
  } else ERROR. -- Cannot commit send direction without
                -- RemoteConnectionDescriptor
}

```

ReceiveCommit()

```

If <current resources committed ≠ resources to commit> then {
    -- skip commit if existing OK
  If not <resources to commit ≤ resources reserved> then {
    If <RemoteConnectionDescriptor provided> then
      RR(RemoteConnectionDescriptor)
    else if <(J.112 QoS) or (RSVP and ReserveDestination provided )> then
      RR(*) -- Use wildcard classifier, send to
            -- ReserveDestination if RSVP
    else ERROR
  }
  If <RemoteConnectionDescriptor provided> then
    RC(RemoteConnectionDescriptor)
  else if <(J.112 QoS) or (RSVP and ReserveDestination provided )> then
    RC(*) -- Use wildcard classifier, send to
          -- ReserveDestination if RSVP
  else ERROR
}

```

SendReserveDelete()

```

If <send resources reserved> then
  SD() -- delete the reservation

```

ReceiveReserveDelete()

```

If <receive resources reserved> then
  RD() -- delete the reservation

```

SendUnCommit()

```

If <send resources commit> then
  SU() -- uncommit committed resources

```

ReceiveUnCommit()

```

If <receive resources committed> then
  RU() -- uncommit committed resources

```

State UpdateState(DoCommit, DoReserve, OldState)

```

If <DoCommit = true> then
  return Commit
else if <DoReserve = true> then
  return Reserve
else
  return OldState

```

APÉNDICE I

Ejemplo de lote de eventos

Este apéndice proporciona un ejemplo de lote de eventos para líneas de acceso analógicas. El lote se incluye aquí únicamente a fines ilustrativos y para facilitar la inclusión de ejemplos informativos en la parte principal de la Recomendación. De ningún modo constituye una de definición de lote completa, ni debe considerarse asignado el nombre del lote que se muestra. Como el lote es meramente un ejemplo, se omiten también aquí los detalles de los eventos y señales individuales y se proporcionan solamente como descripciones de alto nivel con fines ilustrativos.

Ejemplo de lote de línea

Nombre del lote: X

Para identificar los eventos y señales del "ejemplo de lote de línea" para "líneas de acceso analógicas" se utilizan los siguientes códigos:

Código	Descripción	Evento	Señal	Información adicional
0-9,*,#,A, B,C,D	Tonos DTMF	√	BR	
bz	Tono de ocupado	–	TO	
dl	Tono de invitación a marcar	–	TO	
hd	Transición a descolgado	P, S	–	
hf	Colgado instantáneo	P	–	
hu	Transición a colgado	P, S	–	
rg	Señal de llamada	–	TO	
rt	Señal de llamada de retorno	–	C, TO	
t	Temporizador	√	–	
vmwi	Indicador visual de mensaje en espera	–	OO	
X	Comodín de tonos DTMF	√	–	Corresponde a cualquiera de los dígitos "0-9"

Como el lote anterior es meramente un ejemplo, la definición de los eventos y señales individuales que siguen se proporcionan como una descripción de alto nivel solamente. Un lote real e implementable debería especificar los detalles de cada evento y señal. Estos detalles pueden diferir entre los distintos proveedores de servicios RTPC analógicos:

Tonos DTMF (DTMF tones) (0-9,*,#,A, B,C,D): Define todos los tonos DTMF.

Tono de ocupado (bz, busy tone): El tono de ocupado indica a la parte llamante que la parte llamada está ya comprometida en una llamada.

Tono de invitación a marcar (dl, dial-tone): El tono de invitación a marcar indica a la parte llamante que puede realizar una llamada.

Transición a descolgado (hd, off-hook transition): El evento descolgado indica que el teléfono asociado con el punto extremo ha sido descolgado.

Colgado instantáneo (hf, flash hook): El evento colgado instantáneo indica que en el teléfono asociado con el punto extremo se ha producido un colgado instantáneo.

Transición a descolgado (hu, *on-hook transition*): El evento colgado indica que el teléfono asociado con el punto extremo ha sido colgado.

Tono de llamada (rg, *ringing*): La señal de llamada indica que el teléfono de la parte llamada esta sonando.

Tono de llamada de retorno (rt, *ring back tone*): La señal tono de llamada informa a la parte llamante que la parte llamada está siendo avisada.

Temporizador (t, *timer*): Como se describe en 6.1.5, el temporizador T es un temporizador suministrable que sólo puede ser cancelado por una entrada DTMF.

Indicador visual de mensaje en espera (vmwi, *visual message waiting indicator*): La señal indicador visual de mensaje en espera activa o desactiva una indicación visual de mensaje en el buzón vocal.

DTMF tones wildcard (X): El comodín de tonos DTMF corresponde a cualquier dígito DMTF entre 0 y 9.

APÉNDICE II

Ejemplos de codificaciones de instrucciones

Este apéndice presenta ejemplos de instrucciones y respuestas con la codificación real utilizada suponiendo que se aplica el ejemplo de lote de línea. Se dan ejemplos para cada instrucción. Todos los comentarios mostrados en las instrucciones y respuestas son opcionales.

NotificationRequest (petición de notificación)

El primer ejemplo ilustra una NotificationRequest que hará sonar el timbre de un teléfono y esperará un evento de descolgado:

```
RQNT 1201 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
N: ca@ca1.whatever.net:5678
X: 0123456789AC
R: hd(N)
S: rg
```

La respuesta indica que la transacción ha tenido éxito:

```
200 1201 OK
```

El segundo ejemplo ilustra una NotificationRequest que esperará y acumulará un evento de descolgado, y que proveerá después un tono de invitación a marcar y acumulará dígitos de conformidad con el mapa de dígitos proporcionado. La "entidad notificada" se fija a "ca@ca1.whatever.net:5678", y puesto que el parámetro SignalRequests está vacío³⁴, todas las señales TO actualmente activas serán detenidas. Se procesarán todos los eventos almacenados en la memoria de cuarentena, y la lista de eventos que hay que detectar en el estado "notificación" y "lockstep" incluirá tonos de fax además de los "eventos requeridos" y eventos persistentes:

```
RQNT 1202 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
N: ca@ca1.whatever.net:5678
X: 0123456789AC
R: hd(A, E(S(dl), R(B/oc, hu, [0-9#*T] (D))))
D: (0T|00T|#xxxxxxx|*xx|91xxxxxxxxxxxx|9011x.T)
S:
Q: process
T: ft
```

³⁴ Puede haber sido omitido también.

La respuesta indica que la transacción ha tenido éxito:

```
200 1202 OK
```

Notify (notificar)

El ejemplo que sigue ilustra un mensaje Notify (notificar), el cual notifica un evento de descolgado seguido de un número de 12 dígitos que comienza con "91". Se incluye un identificador de transacción que relaciona el mensaje Notify con la NotificationRequest de la cual es resultado. La instrucción se envía a la "entidad notificada" actual, que será normalmente el valor real suministrado en el parámetro NotifiedEntity (entidad notificada), es decir, "ca@ca1.whatever.net:5678" –una situación de cambio-por-fallo puede haber cambiado esta situación:

```
NTFY 2002 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
N: ca@ca1.whatever.net:5678
X: 0123456789AC
O: hd,9,1,2,0,1,8,2,9,4,2,6,6
```

La respuesta Notify indica que la transacción ha sido un éxito:

```
200 2002 OK
```

CreateConnection (crear conexión)

El primer ejemplo ilustra una instrucción CreateConnection (crear conexión) para crear una conexión en el punto extremo especificado. La conexión formará parte del CallId (identificador de llamada) especificado. LocalConnectionOptions (opciones de conexión local) especifica que se utilizará el códec de ley μ G.711 y que el periodo de paquetización será de 10 ms. El modo de conexión será "receive only" ("recibir solamente"):

```
CRCX 1204 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: recvonly
```

La respuesta indica que la transacción ha tenido éxito, y en consecuencia se ha incluido un identificador de conexión para la conexión creada recientemente. Se incluye también una descripción de sesión para la nueva conexión – obsérvese que está precedida por 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
```

El segundo ejemplo ilustra una instrucción CreateConnection que contiene una petición de notificación y un RemoteConnectionDescriptor (descriptor de conexión distante):

```
CRCX 1205 aaln/1@rgw-2569.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: sendrecv
X: 0123456789AD
R: hd
S: rg
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
```

La respuesta indica que la transacción ha fracasado porque el teléfono estaba ya descolgado. Consecuentemente, no se devuelven ni un identificador de conexión (connection-id) ni una descripción de sesión:

```
401 1205 Phone off-hook
```

Nuestro tercer ejemplo ilustra la utilización de la respuesta provisional y la toma de contacto de tres caminos. Creamos otra conexión utilizando esta vez la calidad de servicio dinámica y acusando recibo de la respuesta recibida anterior:

```
CRCX 1206 aaln/1@rgw-2569.whatever.net MGCP 1.0 NCS 1.0
K: 1205
C: A3C47F21456789F0
L: p:10, a:PCMU, dq-gi:A735C2
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
```

Inicialmente se devuelve 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
```

Un poco más tarde se recibe la respuesta definitiva.

```
200 1206 OK
K:
DQ-RI: A12D5F1
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
```

El agente de llamada acusa recibo de la respuesta definitiva como corresponde:

```
000 1206
```

y se completa la transacción.

ModifyConnection (modificar conexión)

El primer ejemplo muestra una instrucción ModifyConnection (modificar conexión) que simplemente fija el modo de conexión de una conexión a "send/receive" ("enviar/recibir") – se fija también la "entidad notificada":

```
MDCX 1209 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
I: FDE234C8
N: ca@ca1.whatever.net
M: sendrecv
```

La respuesta indica que la transacción ha tenido éxito:

```
200 1209 OK
```

En el segundo ejemplo pasamos una descripción de sesión e incluimos una petición de notificación con la instrucción ModifyConnection. El punto extremo iniciará el envío de tonos de llamada al usuario:

```
MDCX 1210 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
I: FDE234C8
M: recvonly
X: 0123456789AE
R: hu
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
```

La respuesta indica que la transacción ha tenido éxito:

```
200 1206 OK
```

DeleteConnection (suprimir conexión) (desde el agente de llamada)

En este ejemplo, el agente de llamada sencillamente ordena al cliente incorporado que suprima la conexión FDE234C8 en el punto extremo especificado.

```
DLCX 1210 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
I: FDE234C8
```

La respuesta indica que la instrucción se ha ejecutado con éxito y que la conexión ha sido suprimida. Se incluyen por tanto también los parámetros de conexión para la conexión:

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

DeleteConnection (suprimir conexión) (desde el cliente incorporado)

En este ejemplo, el cliente incorporado envía una instrucción DeleteConnection (suprimir conexión) al agente de llamada para instruirle sobre la supresión de una conexión en el punto extremo especificado. El ReasonCode (código de motivo) especifica el motivo de la supresión, y se proporcionan también los parámetros de conexión para la conexión:

```
DLCX 1210 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 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 agente de llamada envía a la pasarela una respuesta de que la instrucción se ha ejecutado con éxito:

```
200 1210 OK
```

DeleteConnection (suprimir conexión) (múltiples conexiones desde el agente de llamada)

En el primer ejemplo, el agente de llamada ordena al cliente incorporado que suprima todas las conexiones relativas a la llamada "A3C47F21456789F0" en el punto extremo especificado:

```
DLCX 1210 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0  
C: A3C47F21456789F0
```

La respuesta indica que la instrucción se ha ejecutado con éxito y que las conexiones han sido suprimidas:

```
250 1210 OK
```

En el segundo ejemplo, el agente de llamada ordena al cliente incorporado que suprima todas las conexiones relativas a todos los puntos extremos especificados:

```
DLCX 1210 aaln/*@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
```

La respuesta indica que la instrucción se ha ejecutado con éxito:

```
250 1210 OK
```

AuditEndpoint (auditar punto extremo)

En el primer ejemplo, el agente de llamada desea conocer qué puntos extremos están presentes en el cliente incorporado, y por tanto la utilización del comodín "todos" ("all") para la porción local del nombre de punto extremo:

```
AUEP 1200 *@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
```

El cliente incorporado indica que la instrucción se ha ejecutado con éxito e incluye una lista de nombres de puntos extremos:

```
200 1200 OK  
Z: aaln/1@rgw-2567.whatever.net  
Z: aaln/2@rgw-2567.whatever.net
```

En el segundo ejemplo, se solicitan las capacidades de uno de los puntos extremos:

```
AUEP 1201 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0 F: A
```

La respuesta indica que la instrucción se ha ejecutado con éxito y comunica las capacidades. Se soportan dos códecs, si bien con capacidades diferentes. Consecuentemente se devuelven dos conjuntos de capacidades separados:

```
200 1201 OK  
  
A: a:PCMU, p:10-100, e:on, s:off, v:X;B, m:sendonly;  
      recvonly;sendrecv;inactive;netwloop;netwtest  
A: a:G729A, p:30-90, e:on, s:on, v:X;B, m:sendonly;  
      recvonly;sendrecv;inactive;confrnce;netwloop
```

En el tercer ejemplo, el agente de llamada audita toda la información posible para el punto extremo:

```
AUEP 2002 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0  
F: R,D,S,X,N,I,T,O,ES
```

La respuesta indica que la instrucción se ha ejecutado con éxito:

```
200 2002 OK
R: X/hu,oc(N),[0-9](N)
D:
S: vmwi(+)
X: 0123456789B1
N: [128.96.41.12]
I: 32F345E2
T:
O: hd,9,1,2
ES: hd
```

La lista de eventos solicitados contiene tres eventos. Cuando no se especifica ningún nombre de lote, se adopta el lote por defecto. Lo mismo sucede con las acciones, de modo que debe, por tanto, adoptarse la acción por defecto – Notify – para el evento "X/hu". La omisión de un valor para el "mapa de dígitos" significa que el punto extremo no tiene actualmente un mapa de dígitos. No hay actualmente ninguna señal de temporización activa, aunque la señal OO "vmvi" está actualmente activada y se incluye en consecuencia – en este caso se ha parametrizado, si bien el parámetro podría haber sido excluido. La "entidad notificada" actual hace referencia a una dirección IP y solamente existe una conexión simple para el punto extremo. El valor actual de DetectEvents está vacío, y la lista de ObservedEvents contiene los cuatro eventos especificados. Por último, los estados de eventos auditados revelan que el teléfono ha sido descolgado en el momento en que se procesaba la transacción.

AuditConnection (auditar conexión)

El primer ejemplo muestra una instrucción AuditConnection (auditar conexión) donde auditamos el CallId (identificador de llamada), la NotifiedEntity (entidad notificada), las LocalConnectionOptions (opciones de conexión local), el ConnectionMode (modo de conexión), el LocalConnectionDescriptor (descriptor de conexión local) y los parámetros de la conexión:

```
AUCX 2003 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
I: 32F345E2
F: C,N,L,M,LC,P
```

La respuesta indica que la instrucción se ha ejecutado con éxito e incluye información para la RequestedInfo (información solicitada):

```
200 2003 OK
C: A3C47F21456789F0
N: ca@cal.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
```

En el segundo ejemplo, pedimos auditar el RemoteConnectionDescriptor (descriptor de conexión distante) y el LocalConnectionDescriptor:

```
AUCX 1203 aaln/2@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
I: FDE234C8
F: RC,LC
```

La respuesta indica que la instrucción se ha ejecutado con éxito e incluye información para la RequestedInfo. En este caso no existe ningún 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

v=0
```

RestartInProgress (rearranque en curso)

El primer ejemplo ilustra un mensaje RestartInProgress (rearranque en curso) enviado por un cliente incorporado para comunicar al agente de llamada que el punto extremo especificado quedará fuera de servicio al cabo de 300 segundos.

```
RSIP 1200 aaln/1@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
RM: graceful
RD: 300
```

Las respuestas del agente de llamada indica que la transacción ha tenido éxito:

```
200 1200 OK
```

En el segundo ejemplo, el mensaje RestartInProgress enviado por el cliente incorporado informa al agente de llamada de que todos los puntos extremos del cliente incorporado entrarán en servicio al cabo de 0 segundos, es decir, vuelven al servicio. El retardo pudo también haber sido omitido:

```
RSIP 1204 *@rgw-2567.whatever.net MGCP 1.0 NCS 1.0
RM: restart
RD: 0
```

La respuesta del agente de llamada indica que la instrucción ha tenido éxito, y proporciona además a los puntos extremos en cuestión una nueva "entidad notificada":

```
200 1204 OK
N: CA-1@whatever.net
```

Alternativamente, la instrucción pudo haber fallado con una "entidad notificada" nueva, como en:

```
521 1204 OK
N: CA-1@whatever.net
```

En este caso, la instrucción debería entonces reintentarse con el fin de satisfacer el "procedimiento de rearranque" (véase 6.4.3.5), enviándola esta vez al agente de llamada "CA-1@whatever.net".

APÉNDICE III

Ejemplo de flujo de llamada

En este apéndice se presenta un ejemplo de flujo de llamada entre dos clientes incorporados, EC-1 y EC-2. Debe señalarse que este flujo de llamada, aunque es válido, constituye sólo un ejemplo que puede o no utilizarse en la práctica. Asimismo, que el flujo de llamada utiliza el ejemplo de lote de línea.

En el flujo de llamada a continuación, CA indica el agente de llamada, CDB una base de datos de configuración (*configuration database*) y ACC una base de datos de contabilidad (*accounting database*).

Usr-1	EC-1	CA	CDB	ACC	EC-2	Usr-2
	←	Petición de notificación				
	Acuse de recibo	→				
Descolgado	Notificar	→				
	←	Acuse de recibo				
(Tono de invitación a marcar)	←	Crear conexión + petición de notificación				
	Acuse de recibo(SDP1)	→				
Dígitos	Notificar	→				
	←	Acuse de recibo				
(en curso)	←	Petición de notificación				
	Acuse de recibo	→				
		Consulta(E.164)	→			
		←	IP			
		Crear conexión(SDP1) + Petición de notificación	---	---	→	
		←	---	---	P-Acuse de recibo(SDP2)	
		←	---	---	Acuse de recibo(SDP2)	(timbre)
		Acuse de recibo	---	---	→	
(tono de llamada de retorno)	←	Modificar conexión(SDP2) + Petición de notificación				
	Acuse de recibo	→				
		←	---	---	Notificar	Descolgado
		Acuse de recibo	---	---	→	

Usr-1	EC-1	CA	CDB	ACC	EC-2	Usr-2
	←	Modificar conexión + Petición de notificación				
	Acuse de recibo	→				
	(intervención)	Inicio de llamada	----	→		
		Petición de notificación	----	----	→	
		←	----	----	Acuse de recibo	
		(Llamada establecida)				
		←	----	----	Notificar	Colgado
		Acuse de recibo	----	----	→	
	←	Suprimir conexión				
		Suprimir conexión	----	----	→	
	Acuse de recibo (Datos Perf)	→				
		←	----	----	Acuse de recibo(Datos Perf)	
		Final de llamada	----	→		
		Petición de notificación	----	----	→	
		←	----	----	Acuse de recibo	
Colgado	Notificar	→				
	←	Acuse de recibo				
	←	Petición de notificación				
	Acuse de recibo	→				

Durante estos intercambios el agente de llamada utiliza el perfil NCS de MGCP para controlar ambos clientes incorporados. Los intercambios tienen lugar en ambos lados.

La primera instrucción es una NotificationRequest, enviada por el agente de llamada al cliente incorporado de entrada. La petición constará de las siguientes líneas:

```
RQNT 1201 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
N: ca@ca1.whatever.net:5678
X: 0123456789AB
R: hd
```

El cliente incorporado, en este punto, recibirá una instrucción para que espere un evento de descolgado e informe sobre él. El cliente enviará en primer lugar una respuesta a la instrucción, repitiendo en la respuesta la id de transacción que el agente de llamada adjuntó a la pregunta y proporcionando un código de retorno indicando el éxito de la transacción.

200 1201 OK

Cuando se reconoce el evento de descolgado, el cliente incorporado envía un mensaje Notify al agente de llamada.

```
NTFY 2001 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
N: ca@ca1.whatever.net:5678
X: 0123456789AB
O: hd
```

El agente de llamada acusa inmediatamente recibo de la notificación:

```
200 2001 OK
```

El agente de llamada examina los servicios asociados con un evento descolgado para este punto extremo (podría emprender acciones especiales en el caso de una línea directa, abono no vigente, etc.). La mayoría de veces enviará una instrucción combinada de CreateConnection y NotificationRequest para crear una conexión, proporcionar el tono de invitación a marcar y recoger dígitos DMTF³⁵:

```
CRCX 1202 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: recvonly
N: ca@ca1.whatever.net:5678
X: 0123456789AC
R: hu, [0-9##*T] (D)
D: (0T | 00T | [2-9]xxxxxx | 1[2-9]xxxxxxxxxx | 011xx.T)
S: dl
```

El cliente incorporado acusa recibo de la transacción y envía hacia atrás la identificación de la conexión recientemente creada y la descripción de sesión utilizada para recibir datos de audio:

```
200 1202 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
```

La especificación SDP, en nuestro ejemplo, especifica la dirección en la que el cliente incorporado se encuentra dispuesto para recibir datos de audio (128.96.41.1), el protocolo de transporte (RTP), el puerto RTP (3456) y el perfil audio (AVP). El perfil audio se refiere a la norma RFC 1890, la cual estipula que se ha asignado el tipo 0 de carga útil para la transmisión de ley μ G.711.

El cliente incorporado comenzará la acumulación de dígitos de acuerdo con el mapa de dígitos. Cuando posteriormente se realiza la concordancia del mapa de dígitos, el cliente incorporado notificará los eventos observados al agente de llamada:

```
NTFY 2002 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
N: ca@ca1.whatever.net:5678
X: 0123456789AC
O: 1,2,0,1,8,2,9,4,2,6,6
```

El agente de llamada acusará inmediatamente recibo de esta notificación.

```
200 2002 OK
```

³⁵ El mapa de dígitos real depende del plan de marcación de la zona local así como de los servicios a los que se está abonado. El mapa de dígitos presentado debe considerarse sólo un ejemplo de mapa de dígitos.

En este momento, el agente de llamada enviará una NotificationRequest para detener la recogida de dígitos si todavía continúa la espera de una transición a descolgado. El agente de llamada decide además acusar recibo de recepción de las respuestas para la transacción 1202:

```
RQNT 1203 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
K: 1202
X: 0123456789AD
R: hu
```

El cliente incorporado acusa inmediatamente recibo de esta instrucción.

```
200 1203 OK
```

El agente de llamada debe ahora crear una conexión en el cliente incorporado de salida, EC-2, y activar también el timbre de teléfono asociado al cliente incorporado. Esto se efectúa mediante el envío al cliente incorporado de una instrucción combinada de CreateConnection y NotificationRequest:

```
CRCX 2001 aaln/1@ec-2.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: sendrecv
X: 0123456789B0
R: hd
S: rg
```

```
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
```

El cliente incorporado de salida, en este punto, recibe instrucciones para que active el timbre del teléfono, y para que espere un evento de descolgar e informe sobre el mismo. El evento de descolgado y la señal de llamada (timbre) están sincronizados, de modo que cuando sucede el evento descolgar se para la señal de llamada. La porción crear conexión de la instrucción combinada tiene los mismos parámetros que la instrucción enviada al cliente incorporado de entrada, con dos diferencias:

- El identificador de punto extremo apunta hacia el circuito de salida.
- El mensaje transporta la descripción de sesión devuelta por el cliente incorporado de entrada.
- Como está presente la descripción de sesión, el parámetro "modo" se fija a "emisión/recepción ("send/receive").

Se señala que el identificador de llamada es idéntico en las dos conexiones. Esto es normal porque las dos conexiones pertenecen a la misma llamada.

Suponemos que esta instrucción no termina de ejecutarse inmediatamente³⁶, y el cliente incorporado de salida devuelve por tanto una respuesta provisional acusando recibo de la instrucción, enviando en la descripción de sesión sus propios parámetros como dirección, puertos y perfil RTP así como el identificador de conexión para la nueva conexión:

```
100 2001 Pending
I: 32F345E2
```

³⁶ Esto podría ser debido, por ejemplo, a la reserva de recursos externos, aunque no hemos incluido esta circunstancia en nuestro ejemplo.

```
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
```

Una vez que termina la ejecución de la transacción, el cliente incorporado envía al agente de llamada la respuesta definitiva en la que repite la información que proporcionó en la respuesta provisional:

```
200 2001 OK
K:
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
```

Cuando el agente de llamada recibe la respuesta definitiva, reconoce la presencia del atributo acuse de recibo de respuesta vacío y emite por consiguiente un acuse de recibo de respuesta para la transacción:

```
000 2001
```

El agente de llamada retransmitirá la información al cliente incorporado de entrada, y le dará instrucciones para que genere tonos de llamada, utilizando una instrucción combinada ModifyConnection y NotificationRequest:

```
MDCX 1204 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
I: FDE234C8
M: recvonly
X: 0123456789AE
R: hu
S: rt@FDE234C8
```

```
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
```

El cliente incorporado acusa inmediatamente recibo de la modificación:

```
200 1204 OK
```

En este momento, el agente de llamada ha establecido un trayecto de transmisión semidúplex. El teléfono asociado al cliente incorporado de entrada podrá recibir las señales, como tonos o avisos, que pueden ser generadas en caso de error, así como la conversación inicial que más probablemente será generada cuando el usuario de destino conteste al teléfono.

Cuando se observa el evento descolgado, el cliente incorporado de salida envía un mensaje Notify al agente de llamada:

```
NTFY 3001 aaln/1@ec-2.whatever.net MGCP 1.0 NCS 1.0
X: 0123456789B0
O: hd
```

El agente de llamada acusa recibo inmediatamente de esta notificación.

```
200 3001 OK
```

El agente de llamada envía ahora una instrucción combinada ModifyConnection y NotificationRequest al cliente incorporado de entrada para poner la conexión en el modo enviar/recibir y detiene los tonos de llamada:

```
MDCX 1206 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
I: FDE234C8
M: sendrecv
X: 0123456789AF
R: hu
```

El cliente incorporado responde inmediatamente a la instrucción:

```
200 1206 OK
```

Al mismo tiempo, el agente de llamada pide al cliente incorporado de salida que notifique la ocurrencia de un evento colgar. Realiza esto enviando una NotificationRequest al cliente incorporado³⁷:

```
RQNT 2002 aaln/1@ec-2.whatever.net MGCP 1.0 NCS 1.0
X: 0123456789B1
R: hu
```

El cliente incorporado responde inmediatamente a la instrucción:

```
200 2002 OK
```

En este punto, la llamada está totalmente establecida.

En algún momento posterior, el teléfono asociado al cliente incorporado de salida, en nuestro escenario, cuelga. Este evento es comunicado al agente de llamada, de conformidad con las normas recibidas en la última NotificationRequest mediante el envío de una instrucción Notify:

```
NTFY 2003 aaln/1@ec-2.whatever.net MGCP 1.0 NCS 1.0
X: 0123456789B1
O: hu
```

El agente de llamada responde inmediatamente a la instrucción:

```
200 2003 OK
```

El agente de llamada determina ahora que la llamada está finalizando y envía por tanto a ambos clientes incorporados una instrucción DeleteConnection:

```
DLCX 1207 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
I: FDE234C8
```

```
DLCX 2004 aaln/1@ec-2.whatever.net MGCP 1.0 NCS 1.0
C: A3C47F21456789F0
I: 32F345E2
```

³⁷ Debe señalarse que, aunque el evento colgar es persistente, el modo "lockstep" requiere que el agente de llamada envíe una nueva NotificationRequest al cliente incorporado.

Los clientes incorporados responderán con acuses de recibo que incluyen los parámetros de conexión para la conexión:

```
250 1207 OK
P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48
250 2004 OK
P: PS=790, OS=45700, PR=1230, OR=61875, PL=15, JI=27, LA=48
```

El agente de llamada enviará también una nueva NotificationRequest al cliente incorporado de salida para que esté preparado a recibir el siguiente evento descolgado detectado por el cliente incorporado:

```
RQNT 2005 aaln/1@ec-2.whatever.net MGCP 1.0 NCS 1.0
X: 0123456789B2
R: hd
```

El cliente incorporado acusa recibo de este mensaje:

```
200 2005 OK
```

Por último, el cliente incorporado de entrada cuelga el teléfono generando con ello un mensaje Notify para el agente de llamada:

```
NTFY 1208 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
X: 0123456789AF
O: hu
```

El agente de llamada responde inmediatamente a la instrucción:

```
200 1208 OK
```

El agente de llamada envía a continuación una nueva NotificationRequest al cliente incorporado de entrada para que esté preparado a recibir el siguiente evento descolgado detectado por el cliente incorporado:

```
RQNT 1209 aaln/1@ec-1.whatever.net MGCP 1.0 NCS 1.0
X: 0123456789B3
R: hd
```

El cliente incorporado acusará recibo de este mensaje:

```
200 1209 OK
```

Ambos clientes incorporados están, en este punto, preparados para la llamada siguiente.

APÉNDICE IV

Interacciones de modo

Una conexión MGCP puede establecer uno o más trenes de medios. Estos trenes pueden ser entrantes (procedentes de un punto extremo distante) o salientes (generados en el micrófono de un puesto telefónico). El parámetro "modo de conexión" establece la dirección y generación de estos trenes. Cuando hay solamente una conexión con un punto extremo, el cuadro de correspondencia de estos trenes es sencillo; el aparato telefónico trata el tren entrante en el auricular y genera el tren saliente a partir de la señal del micrófono, dependiendo del parámetro modo.

Sin embargo, cuando hay varias conexiones establecidas con un punto extremo, puede haber muchos trenes entrantes y salientes. Dependiendo del modo de conexión utilizado estos trenes pueden interactuar de forma diferente con cada uno de los demás y con los trenes entrantes y salientes del aparato telefónico.

En el cuadro a continuación se describe cómo deben combinarse diferentes conexiones cuando están "activas" a la vez. Una conexión activa se define aquí como una conexión que se encuentra en uno de los siguientes modos:

- "send/receive" (emisión/recepción)
- "send only" (emisión solamente)
- "receive only" (recepción solamente)
- "replicate" (repetir)
- "conference" (conferencia)

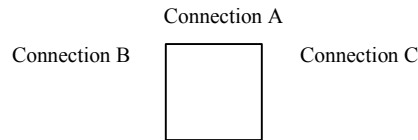
Las conexiones en los modos "network loopback" (bucle de red), "network continuity test" (prueba de continuidad de red), o "inactive"(inactivo) no se ven afectadas por conexiones en los modos "active" (activo). El cuadro aplica los siguientes convenios:

- A_{in} es el tren de medios entrante procedente de conexión A.
- B_{in} es el tren de medios entrante procedente de conexión B.
- H_{in} es el tren de medios entrante procedente del micrófono del aparato telefónico.
- A_{out} es el tren de medios saliente hacia conexión A.
- B_{out} es el tren de medios saliente hacia conexión B.
- H_{out} es el tren de medios saliente hacia el auricular del aparato telefónico.
- NA indica que no hay ningún tren en absoluto.

		Modo de la conexión A						
		sendonly	recvonly	sendrecv	confrnce	inactive	netwloop/ netwtest	replcate
Modo de la conexión B	sendonly	$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=NA$	$A_{out}=NA$ $B_{out}=H_{in}$ $H_{out}=A_{in}$	$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=A_{in}$	$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=A_{in}$	$A_{out}=NA$ $B_{out}=H_{in}$ $H_{out}=NA$	$A_{out}=A_{in}$ $B_{out}=H_{in}$ $H_{out}=NA$	$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=NA$
	recvonly		$A_{out}=NA$ $B_{out}=NA$ $H_{out}=A_{in}+B_{in}$	$A_{out}=H_{in}$ $B_{out}=NA$ $H_{out}=A_{in}+B_{in}$	$A_{out}=H_{in}$ $B_{out}=NA$ $H_{out}=A_{in}+B_{in}$	$A_{out}=NA$ $B_{out}=NA$ $H_{out}=B_{in}$	$A_{out}=A_{in}$ $B_{out}=NA$ $H_{out}=B_{in}$	$A_{out}=H_{in}+B_{in}$ $B_{out}=NA$ $H_{out}=B_{in}$
	sendrecv			$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=A_{in}+B_{in}$	$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=A_{in}+B_{in}$	$A_{out}=NA$ $B_{out}=H_{in}$ $H_{out}=B_{in}$	$A_{out}=A_{in}$ $B_{out}=H_{in}$ $H_{out}=B_{in}$	$A_{out}=H_{in}+B_{in}$ $B_{out}=H_{in}$ $H_{out}=B_{in}$
	confrnce				$A_{out}=H_{in}+B_{in}$ $B_{out}=H_{in}+A_{in}$ $H_{out}=A_{in}+B_{in}$	$A_{out}=NA$ $B_{out}=H_{in}$ $H_{out}=B_{in}$	$A_{out}=A_{in}$ $B_{out}=H_{in}$ $H_{out}=B_{in}$	$A_{out}=H_{in}+B_{in}$ $B_{out}=H_{in}$ $H_{out}=B_{in}$
	inactive					$A_{out}=NA$ $B_{out}=NA$ $H_{out}=NA$	$A_{out}=A_{in}$ $B_{out}=NA$ $H_{out}=NA$	$A_{out}=H_{in}$ $B_{out}=NA$ $H_{out}=NA$
	netwloop/ netwtest						$A_{out}=A_{in}$ $B_{out}=B_{in}$ $H_{out}=NA$	$A_{out}=H_{in}$ $B_{out}=B_{in}$ $H_{out}=NA$
	replcate							$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=NA$

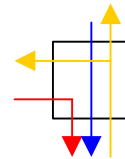
Si hay tres o más canales "activos" interactuarán aún como se ha definido en el cuadro anterior, con los trenes de medios salientes combinados para cada interacción (Unión de todos los trenes). Si se agotan los recursos internos y los trenes no pueden mezclarse, la pasarela debe devolver un código de error recursos No disponibles.

Estas conexiones pueden representarse gráficamente como sigue:

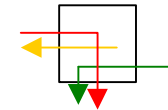


Por ejemplo, si conexión A es Sendrecv, conexión B confnrc y conexión C recvonly, las salidas en cada modo se obtendrán a partir del cuadro anterior y serán:

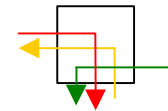
Interacción de A a B: $B_{out}=H_{in}$ $A_{out}=H_{in}$ $H_{out}=A_{in}+B_{in}$



Interacción de A a C: $A_{out}=H_{in}$ $C_{out}=NA$ $H_{out}=A_{in}+C_{in}$



Interacción de B a C: $B_{out}=H_{in}$ $C_{out}=NA$ $H_{out}=B_{in}+C_{in}$



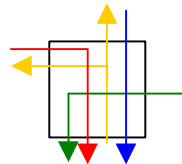
Tomando la unión de todos los trenes en cada salida obtenemos:

$$A_{out}=H_{in}$$

$$B_{out}=H_{in}$$

$$C_{out}=NA$$

$$H_{out}=B_{in}+A_{in}+C_{in}$$



Para mayor claridad, el cuadro presentado anteriormente se repite a continuación en forma de gráfico [excluido el modo "replicate" ("repetir")]:

		Modo de la conexión A (arriba)					
		sendonly	recvonly	sendrecv	confrnce	inactive	netwloop/ netwtest
Modo de la conexión B (izquierda)	sendonly						
	recvonly						
	sendrecv						
	confrnce						
	inactive						
	netwloop/ netwtest						

APÉNDICE V

Información de compatibilidad

Este apéndice proporciona información de compatibilidad del protocolo NCS.

Compatibilidad MGCP

NCS es un perfil de MGCP 1.0; sin embargo NCS ha introducido también una pareja de adiciones. En la siguiente lista se recogen las adiciones de NCS que no están actualmente incluidas en el MGCP:

- **Esquema de denominación de los puntos extremos (Endpoint Naming Scheme)** – Las reglas de aplicación de comodines son más restrictivas que en MGCP.
- **ModifyConnection incorporada (Embedded ModifyConnection)** – Se ha introducido una nueva acción ModifyConnection incorporada.
- **Calidad de servicio dinámica (Dynamic Quality of Service)** – NCS soporta los servicios de seguridad de IPCablecom. Esto influye en LocalConnectionOptions, Capabilities y SDP. Se añade también un nuevo parámetro de retorno, ResourceID (identificador de recursos), para las instrucciones CreateConnection y ModifyConnection.
- **Seguridad (Security)** – NCS soporta los servicios de seguridad de IPCablecom. Esto influye en LocalConnectionOptions, Capabilities y SDP.
- **Recuperación de nombre de punto extremo (Endpoint Name Retrieval)** – Se ha ampliado la instrucción AuditEndpoint con una capacidad para devolver el número de puntos extremos que aplican un comodín y con un mecanismo de recuperación bloque a bloque de estos nombres de puntos extremos. Además de la ampliación en sí de la instrucción AuditEndpoint, ello implica la introducción de dos nuevos nombres de parámetro: MaxEndPointIds y NumEndPoints
- **Versiónes soportadas (Supported Versions)** – La respuesta RestartInProgress y la instrucción AuditEndpoint se han ampliado con un parámetro VersionSupported que permita a los agentes de llamada y a las pasarelas determinar qué versiones de protocolo soportan cada uno.
- **Códigos de error (Error Codes)** – Se han introducido dos nuevos códigos de error: 532 y 533.
- **Utilización de SDP (Usage of SDP)** – Se incluye en NCS un nuevo perfil de utilización de SDP. Como característica más notable, el perfil y todos los ejemplos de utilización requieren una estricta conformidad con el SDP, con independencia de la utilidad de los campos incluidos. Asimismo, IPCablecom especifica extensiones que se han añadido a SDP.
- **Respuesta provisional (Provisional Response)** – Se ha incluido en NCS la especificación del mecanismo de respuesta provisional y detalles adicionales. Se ha introducido una respuesta acuse de recibo de respuesta (000), se ha admitido un parámetro ResponseAck vacío en las respuestas definitivas que siguen a las respuestas provisionales y se ha especificado un procedimiento para el mecanismo citado.
- **Parámetros señal (Signal Parameters)** – Se ha ampliado la sintaxis de parámetros de señal para tener en cuenta la utilización de paréntesis balanceados dentro de los parámetros de señal. Todas las señales de temporización pueden tener sus valores de temporización alterados por un parámetro de señal.
- **Lotes de eventos (Event Packages)** – NCS introduce un conjunto de nuevos lotes de eventos.

Finalmente, debe observarse que NCS proporciona interpretaciones del comportamiento del protocolo MGCP base, y en algunos casos una especificación o clarificación adicional del mismo que pueden o no reflejar el comportamiento deseado del MGCP.

APÉNDICE VI

Otros ejemplos de lotes de eventos

Este apéndice define ejemplos de lotes de eventos adicionales para los distintos de puntos extremos definidos actualmente para clientes incorporados.

Líneas de acceso analógicas

Para los puntos extremos línea de acceso analógica se definen normalmente los lotes siguientes:

- Línea japonesa.
- ADSI.

Lote línea japonesa

Nombre del lote: J

Los códigos dados a continuación se utilizan para identificar eventos y señales del lote "Japanese line" (línea japonesa) de "líneas de acceso analógicas":

1) *Tipos de señalización de línea de abonado*

Las señales de línea de abonado (señales) se pueden clasificar en señales relativas al control de la conexión (señales de supervisión), señales relativas al control de la selección (señales de selección) y señales de tonos audibles (tonos audibles).

2) *Señales de supervisión*

Código	Nombre de señal	Evento	Señal	Información adicional
cs	Señal de llamada	P, S	–	Notificación de llamada de origen (= Transición a descolgado)
ir	Tono de llamada	–	TO	Notificación de llamada entrante Temporización = infinito Véase el artículo 31, punto 2 de Carriers Telecommunication Facilities Regulations
as1	Señal de respuesta 1	P, S	–	Notificación de que el terminal llamado ha contestado (Terminal a red) (= Transición a descolgado)
as2	Señal de respuesta 2	–	TO	Notificación de que el terminal llamado ha contestado (Red a terminal) Temporización = infinito
ds1	Señal de desconexión 1	P, S	–	Notificación de que la comunicación se ha completado (Terminal a red) (= Transición a descolgado)
ds2	Señal de desconexión 2	–	TO	Notificación de que el terminal de origen ha terminado la comunicación (Red a terminal) Temporización = infinito

Código	Nombre de señal	Evento	Señal	Información adicional
cbs	Señal de colgar	P, S	–	Notificación de que el terminal llamado ha terminado la comunicación (= Transición a descolgado)
hs	Señal de cambio	P	–	Para la "llamada en espera" y el "servicio tripartito"
sir	Señal de extensión de llamada	–	TO	Generada por el sistema de extensión centralizado (CES, <i>centralized extension system</i>). Temporización = infinito
tir	Señal de advertencia de reenvío de llamada	–	TO	Para el servicio "Voice Warp". Temporización = 2-3 s
car	Señal de activación del terminal de recepción de datos	–	TO	Notificación por la señal de MÓDEM. Temporización = infinito
pas	Señal de respuesta primaria	P, S	–	Para visualización de número. (= Transición a descolgado)
iss	Señal de llamada entrante fructuosa	P, S	–	Para visualización de número (= Transición a colgado)
cei1(nu)	ID de llamado (tono de PB)	–	BR	"nu" indica número
cei2(nu)	ID de llamado (tono de módem)	–	BR	"nu" indica número
ci	ID de llamante	–	BR	"nu" indica número
aw	Tono de respuesta	✓	–	
ft	Tono de fax	✓	–	
mt	Tono de módem	✓	–	
ma	Arranque de medios	C	–	
oc	Operación completada	✓	–	
of	Operación fracasada	✓	–	
t	Temporizador	✓	–	
l	DTMF de larga duración	✓	–	
ld	Conexión de larga duración	C	–	

3) *Señal de selección*

Código	Nombre de señal	Evento	Señal	Información adicional
ssn	Señal de selección (0-9,*,#)	✓	BR	Temporización de marcación parcial = 20-30 s Temporización entre dígitos = 4-6 s
ssw	Comodín de tonos PB	✓	–	Corresponde a cualquiera de los dígitos "0-9"

4) *Tonos audibles*

Código	Nombre de señal	Evento	Señal	Información adicional
dt	Tono de invitación a marcar	–	TO	Preparado para recibir la señal de selección Temporización = 20-30 s
sdt	Segundo tono de invitación a marcar	–	TO	Para servicios que utilizan registro, como el "servicio de reenvío de llamada", el "servicio de respuesta telefónica automática") Temporización = 20-30 s
rbt	Tono de llamada de retorno	–	C,TO	Temporización = infinito
bt	Tono de ocupado	–	TO	Temporización = 60-70 s
cpt	Tono de aceptación	–	BR	Para servicios que utilizan registro, como el "servicio de reenvío de llamadas", el "servicio de respuesta telefónica automática"
hst	Tono de servicio de retención de llamada	–	TO	Temporización = infinito
iit	Tono de identificación de llamada entrante	–	C, BR	Para el "servicio de respuesta telefónica automática"
siit	Tono de identificación de llamada entrante específica	–	C, BR	En caso de doble contrato con el "servicio de respuesta telefónica automática" y el "servicio "NARIWAKE"
nft	Tono de notificación	–	TO	Solamente para el "servicio de recepción de identificación de mensaje" Temporización = 3-4 s
how1	Tono de zumbador 1	–	TO	Temporización = 10-22 s
how2	Tono de zumbador 2	–	TO	Temporización = infinito

Las definiciones de cada uno de los eventos y señales son como sigue:

Señal de llamada (cs, *calling signal*): Notifica a la red la originación de una llamada.

Señal de timbre (ir, ringing signal): Véase el artículo 31, punto 2 de Carriers Telecommunication Facilities Regulations. El proceso de aprovisionamiento puede definir la cadencia del timbre. La señal de timbre puede ser parametrizada con el parámetro de señal "rep" que especifica el número máximo de ciclos (repeticiones) de la señal de timbre que se ha de aplicar. En lo que sigue aplicaremos una señal de timbre de hasta 6 ciclos de llamada.

S: ir(rep=6)

Se considera un error probar y hacer sonar un teléfono que está descolgado, por lo que debe devolverse un error cuando se producen tales tentativas.

Señal de respuesta (as, answer signal): Notifica a la red que el terminal llamado ha contestado (as1). En el sentido inverso, la red notifica al terminal de origen que el terminal llamado ha contestado (as2).

Señal de desconexión (ds, disconnect signal): El terminal de origen notifica a la red que la comunicación se ha completado (ds1). En el sentido inverso, la red notifica al terminal llamado que el terminal de origen ha terminado la comunicación (ds2).

Señal de colgar (cbs, clear back signal): Notifica a la red que el terminal llamado ha terminado la comunicación.

Señal de cambio (hs, hooking signal): El terminal notifica a la red que una asignación o un servicio ha cambiado durante la comunicación. Esta señal se utiliza para la "llamada en espera" y el "servicio tripartito".

Señal de extensión de llamada (sir, extension call signal): Con un teléfono del sistema de extensión centralizado (CES, *centralized extension system*), la red notifica al terminal que hay una llamada entrante que está siendo reenviada. Además, en el caso del "servicio NARIWAKE", la red comunica al terminal que hay una llamada entrante procedente de una parte que desea ser identificada.

Señal de advertencia de reenvío de llamada (tir, call forward warning signal): Durante el arranque del servicio "reenvío telefónico" ("Forwarding telephone") o el modo de transferencia incondicional en "Voice warp", la red notifica al terminal que hay una llamada entrante para el cliente abonado y que se ha activado el reenvío.

Señal de activación del terminal de recepción de datos (car, data reciving terminal activation signal): La red notifica a un terminal de recepción de datos que hay una llamada entrante con información notificada por la señal de módem.

Señal de respuesta primaria (pas, primary answer signal): El terminal llamado notifica a la red que el aparato telefónico ha sido descolgado. Esta función se utiliza para la visualización del número.

Señal entrante fructuosa (iss, incoming successful signal): La red notifica al terminal de origen que la señal entrante se ha recibido con éxito. Esta función se utiliza para la visualización del número.

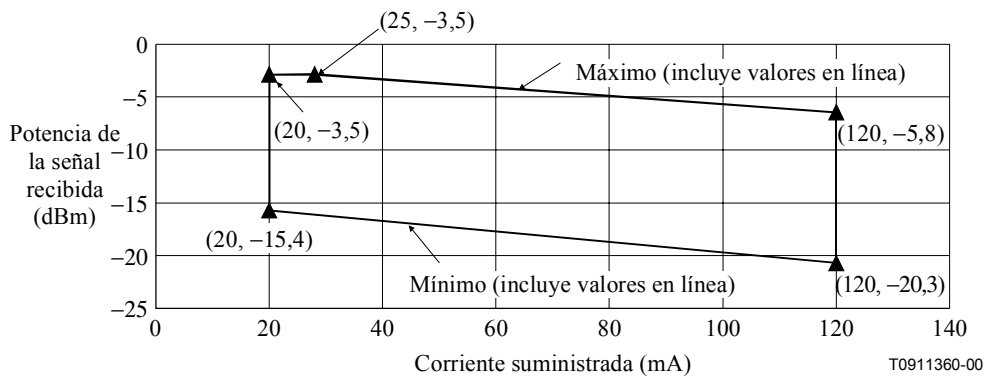
Señal de selección (ss, selection signal): El terminal de origen notifica a la red el tipo de servicio y el número de la otra parte. El código se asigna para la señal de selección (0-9, *, #) como ssn, los comodines de tonos PB como ssw. Las frecuencias y los niveles de recepción de señales de marcación por teclado (PB, *push button*) se muestran en los cuadros y figuras a continuación.

1) *Frecuencia*

Frecuencia del grupo frecuencias altas	1209 Hz	1336 Hz	1477 Hz
Frecuencias del grupo frecuencias bajas			
697 Hz	1	2	3
770 Hz	4	5	6
852 Hz	7	8	9
941 Hz	*	0	#

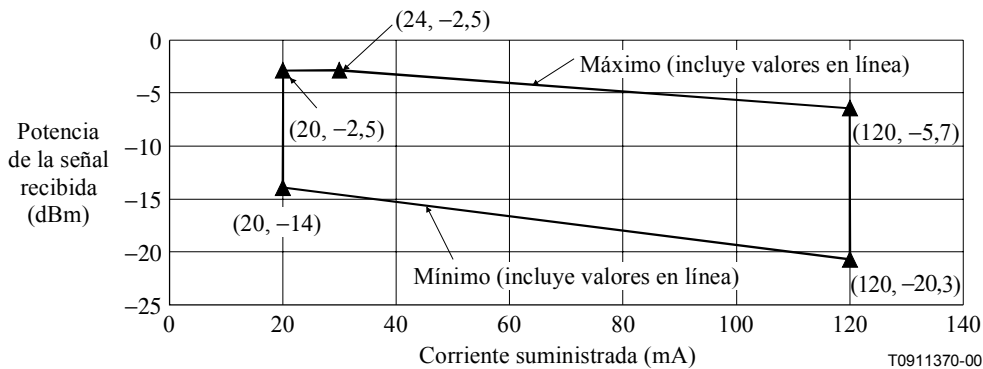
2) *Norma de recepción*

Ítem		Norma
Desviación de frecuencia de la señal		Dentro de $\pm 1,5\%$
Gama de tolerancia de la potencia de la señal recibida	Frecuencias del grupo de frecuencias bajas	Se muestra en la figura VI.1
	Frecuencias del grupo de frecuencias altas	Se muestra en la figura VI.2
	Desviación de potencia eléctrica entre dos frecuencias	Dentro de 5 dB, si bien la potencia eléctrica para la frecuencia del grupo de frecuencias bajas debe ser inferior que para la frecuencia del grupo de frecuencias altas.
Tiempo de salida de la señal		50 ms o más
Pausa mínima		30 ms o más
Ciclo		120 ms o más
NOTA 1 – La pausa mínima es el tiempo muerto más corto entre señales adyacentes.		
NOTA 2 – Un ciclo es la suma del tiempo de emisión de la señal y la pausa mínima.		



NOTA – La potencia recibida de señal cuando la corriente suministrada es menor de 20 mA debe estar comprendida entre $-15,4$ dBm y $-3,5$ dBm. Cuando la corriente suministrada es mayor de 120 mA, debe estar comprendida entre $-20,3$ dBm y $-5,8$ dBm.

Figura VI.1/J.162 – Gama de tolerancia de la potencia de la señal recibida (frecuencia del grupo frecuencias bajas)



NOTA – La potencia de la señal recibida cuando la corriente suministrada es menor de 20 mA debe estar comprendida entre -14 dBm y $-2,5$ dBm. Cuando la corriente suministrada es mayor de 120 mA, debe estar comprendida entre $-20,3$ dBm y $-5,7$ dBm.

Figura VI.2/J.162 – Gama de tolerancia de la potencia de la señal recibida (frecuencia del grupo de frecuencias bajas)

Otras condiciones se estipulan en la Ordenanza 13 del Ministerio de Correos y Telecomunicaciones, 1998.

Tono de invitación a marcar (dt, dial tone): La red notifica al terminal de origen que está preparada para recibir la señal de selección. En una llamada off-net procedente de un teléfono de una red miembro, la red notifica al terminal de origen que está preparada para recibir la señal de selección. El tono de invitación a marcar es un tono de c.a. con una frecuencia 400 Hz y niveles comprendidos entre $(-22-L)$ y -19 dBm, siendo L la pérdida de transmisión en un bucle de abonado a 400 Hz.

Segundo tono de invitación a marcar (sdt, second dial tone): La red notifica al terminal de origen que está preparada para recibir la segunda señal de selección. En una llamada off-net procedente de un teléfono de una red miembro, la red notifica al terminal de origen que está preparada para recibir la señal de selección. El segundo tono de invitación a marcar es un tono de c.a. con frecuencia de 400 Hz y niveles entre $(-22-L)$ y -19 dBm, siendo L la pérdida de transmisión en un bucle de

abonado a 400 Hz. La relación relación reposo-trabajo y la relación de trabajo están dentro de 240 IPM y 50%, respectivamente.

Tono de llamada de retorno (rbt, *ringing back tone*): La red notifica al terminal de origen que está llamando al terminal receptor. El tono se termina cuando se recibe una señal de respuesta procedente del terminal llamado. El tono de llamada audible es una combinación de dos tonos de c.a. con frecuencias de 400 y 15-20 Hz y niveles entre -4 y $(-29-L)$ dBm, siendo L la pérdida de transmisión en un bucle de abonado a 400 Hz. La relación reposo-trabajo y la relación de trabajo están dentro de $20 \text{ IPM} \pm 20\%$ y $33 \pm 10\%$, respectivamente. (Índice de modulación: dentro de $85 \pm 15\%$.)

Tono de ocupado (bt, *busy tone*): La red notifica al terminal de origen que el terminal receptor está comunicando, por lo que no puede ejecutar el servicio o conexión que solicita el terminal de origen. El tono de ocupado es un tono de c.a. con frecuencia de 400 Hz y niveles entre $(-29-L)$ y -4 dBm, siendo L la pérdida de transmisión en un bucle de abonado a 400 Hz. La relación reposo-trabajo y la relación de trabajo están dentro de $60 \text{ IPM} \pm 20\%$ y $50 \pm 10\%$, respectivamente.

Tono de aceptación (cpt, *acceptance tone*): La red notifica al terminal de origen que ha recibido la petición de servicio. El tono de aceptación es un tono de c.a. con frecuencia de 400 Hz y niveles entre $(-26-L)$ y -16 dBm, siendo L la pérdida de transmisión en un bucle de abonado a 400 Hz.

Tono para el servicio de retención de llamada (hst, *hold service tone*): La red notifica a un terminal en espera que el estado de espera continúa. Un tono del servicio de retención audible es una combinación de dos tonos de c.a. con frecuencias de 400 y 16 Hz y niveles entre -14 y $(-22-L)$ dBm, siendo L la pérdida de transmisión en un bucle de abonado a 400 Hz. (Índice de modulación: dentro de 85%.)

Tono de identificación de llamada entrante (iit, *incoming identification tone*): La red notifica al terminal llamado pertinente que ha recibido una llamada entrante procedente de un tercero durante la conversación con un segundo. El tono de identificación de llamada entrante audible es una combinación de dos tonos de c.a. con frecuencias de 400 y 16 Hz y niveles entre -14 y $(-25-L)$ dBm, siendo L la pérdida de transmisión en un bucle de abonado a 400 Hz. (Índice de modulación: dentro de 85%.)

Tono de identificación de llamada entrante específica (siit, *specific incoming identification tone*): La red notifica al terminal llamado pertinente que ha recibido una llamada entrante procedente de un tercero que ha sido identificado. El tono de identificación de llamada entrante específica audible es una combinación de dos tonos de c.a. con frecuencias de 400 y 16 Hz y niveles entre -14 y $(-25-L)$ dBm, siendo L la pérdida de transmisión en un bucle de abonado a 400 Hz. (Índice de modulación: dentro de 85%.)

Tono de notificación (nft, *notification tone*): La red notifica al terminal de un cliente que está abonado al "servicio de recepción de identificación de mensajes" que ha recibido una identificación de mensaje. El tono de notificación es un tono de c.a. de una frecuencia de 400 Hz y niveles entre $(-26-L)$ y -16 dBm, donde L es la pérdida de transmisión en un bucle de abonado a 400 Hz.

Tono de zumbador (how, *howler tone*): La red notifica a un terminal que un receptor telefónico no utilizado ha estado descolgado durante un cierto tiempo para pedir con apremio que el microteléfono sea colgado. Se proporcionan los tonos de zumbador. El tono 1 (how1) es un tono c.a. de una frecuencia de 400 Hz y niveles de $+35$ dBm o inferiores. El tono 1 es un sonido que aumenta gradualmente durante 3-15 segundos con una temporización de 10-22 segundos. El tono de zumbador 2 (Howler tone 2 - how2) se genera por una combinación de tres tonos en las frecuencias de 1600 Hz, 1000 Hz y 2000 Hz con una cadencia de 0,5 segundos de duración para el tono de 1600 Hz, y la repetición dos veces de 0,125 segundos de los tonos de 1000 Hz y 2000 Hz. El nivel del tono combinado es -1 dBm o menor. Entre estos tonos audibles se inserta un aviso vocal tal como "el receptor está descolgado". Se considera un error intentar que suene el tono de zumbador en un teléfono que está colgado, y debe por consiguiente devolverse un error cuando se llevan a cabo tales tentativas. El tono 2 (how2) es una señal con una temporización infinita.

ID de llamado (Callee ID) (cei1(nu)): La selección directa de extensiones requiere el ID del llamado para el sistema de señalización PB (por teclado).

ID del llamado (Callee ID) (cei2(nu)): La selección directa de extensiones requiere el ID del llamado para el sistema de señalización de módem.

Id del llamante (ci(tiempo, número, nombre)) (Caller Id) (ci(time, number, name)): Los tres campos son opcionales independientemente, si bien siempre se incluirán cada una de las comas correspondientes.

- El parámetro **time (tiempo)** se codifica como "MM/DD/HH/MM", donde MM es un valor de dos dígitos para el mes entre 01 y 12, DD un valor de dos dígitos para el día del mes entre 1 y 31 y las horas y los minutos son valores de dos dígitos codificados según el tiempo local militar, por ejemplo, 00 es medianoche, 01 es 1 a.m. y 13 es 1 p.m.
- El parámetro **number (número)** se codifica como una cadena de caracteres ASCII de dígitos decimales que identifican el número de la línea llamante. Los espacios en blanco están permitidos cuando la cadena está entrecomillada, si bien estos espacios serán ignorados.
- El parámetro **name (nombre)** se codifica como una cadena de caracteres ASCII que identifica el nombre de la línea llamante. Los espacios en blanco están permitidos si la cadena está entrecomillada.

Una "P" en los campos nombre o número se utiliza para indicar que se trata de un nombre o número privado, y una "O" se utiliza para indicar que el nombre o número no está disponible. En los siguientes ejemplos se ilustra la utilización de la señal ID del llamante:

```
S: ci(02/20/19/47, "5273 4671", JCTEA)
```

Tono de respuesta (aw, answer tone): Tono de respuesta es un tono que puede ser proporcionado por un módem o un fax que contesta a una llamada entrante. El tono consiste en una señal sinusoidal de 2100 Hz – véase UIT-T V.8.

Tono de fax (ft, fax tone): El evento tono de fax se genera siempre que se detecta una llamada de fax – véase, por ejemplo, UIT-T T.30 o UIT-T V.21.

Arranque de medios (ma, media start): El evento arranque de medios tiene lugar en una conexión cuando se recibe por la conexión el primer paquete de medios RTP válido³⁸. 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 se puede detectar en una conexión. Cuando no se especifica ninguna conexión, el evento se aplica a todas las conexiones para el punto extremo, con independencia de cuándo se han creado las conexiones.

Tonos de módem (mt, modem tones): El evento tono de módem se genera siempre que detecta una llamada de módem es detectada – véase, por ejemplo, UIT-T V.8.

Operación completada (oc, operation complete): El evento operación completada se genera cuando se pide a la pasarela que aplique una o varias señales de tipo TO en el punto extremo, y una o más de aquellas señales son completadas sin que detengan por la detección de un evento solicitado, tal como la transición a descolgado o la marcación de dígitos. El informe de compleción puede cursar como parámetro el nombre de la señal que ha llegado al final de su tiempo de vida, como en:

```
O: L/oc(L/dt)
```

³⁸ Cuando se utilizan los servicios de autenticación y seguridad, un paquete RTP no se considera válido hasta que ha superado las pruebas de seguridad.

Cuando la señal informada se ha aplicado a una conexión, el parámetro suministrado deberá incluir también el nombre de la conexión, como en:

O: L/oc(L/rbt@0A3F58)

Cuando se solicita el evento operación completada, éste puede ser parametrizado con cualesquiera parámetros de evento. Cuando se omite el nombre de lote, se supone el nombre de lote por defecto.

El evento operación completada puede generarse además del modo que se define en el protocolo base, por ejemplo, cuando se completa con éxito una instrucción "modificar conexión incorporada", como en³⁹:

O: L/oc(B/C)

Operación fracasada (of, *operation failure*): En general, el evento operación fracasada se puede generar cuando se ha pedido al punto extremo que aplique una o varias señales de tipo TO en el punto extremo, y una o más de aquellas señales han fracasado antes de que finalice la temporización. El informe de compleción puede cursar como parámetro el nombre de la señal que ha fracasado, como en:

O: L/of(L/ir)

Cuando se ha aplicado a una conexión la señal informada, el parámetro suministrado deberá incluir también el nombre de la conexión, como en:

O: L/of(L/rbt@0A3F58)

Cuando se solicita el evento operación fracasada, los parámetros del evento no puede especificarse. Cuando se omite el nombre de lote, se supone el nombre de lote por defecto.

El evento operación fracasada puede generarse además como se especifica en el protocolo de base, por ejemplo, cuando falla una instrucción "modificar conexión incorporada", como en³⁹:

O: L/of(B/C(M(sendrecv(AB2354))))

Temporizador (t, *timer*): El temporizador T es un temporizador suministrable que solamente puede ser cancelado por una entrada multifrecuencia bitono (DTMF). Cuando se utiliza el temporizador T con la acción "accumulate according to digit map" ("acumular de acuerdo con mapa de dígitos"), el temporizador no se arranca hasta que ha entrado en primer dígito, y el temporizador se rearranca después de la entrada de cada nuevo dígito hasta que se produce una concordancia o una discordancia con el mapa de dígitos. En este caso, el temporizador T funciona como un temporizador entre dígitos y toma uno de los dos valores, T_{par} o T_{crit} . Cuando se requiere al menos un dígito para que la cadena de dígitos concuerde con alguno de los esquemas del mapa de dígitos, el temporizador T toma el valor T_{par} , correspondiente a la temporización de marcación parcial. Si dispone de todo lo necesario para producir una concordancia, el temporizador T toma el valor T_{crit} correspondiente a la temporización crítica. Un ejemplo de utilización es:

S: dt
R: [0-9T] (D)

Cuando el temporizador T se utiliza sin la acción "accumulate according to digit map", el temporizador T toma el valor T_{crit} , y es arrancado inmediatamente y cancelado sencillamente (pero no rearrancado) tan pronto como ha entrado un dígito. En este caso, el temporizador T se puede utilizar como un temporizador entre dígitos cuando se aplica la emisión con solapamiento, por ejemplo:

³⁹ Obsérvese la utilización que se hace aquí de "B" como prefijo del parámetro informado.

R: [0-9] (N) , T(N)

Obsérvese que solamente se puede utilizar una de las dos formas en un momento dado, ya que un evento de concreto sólo se puede especificar una vez.

El valor por defecto de T_{par} es 16 segundos y el valor por defecto de T_{crit} es 4 segundos. El proceso de aprovisionamiento puede alterar estos dos valores.

DTMF de larga duración (l, *DTMF long duration*): La señal "DTMF de larga duración" se observa cuando se produce una señal DTMF durante un periodo superior a dos segundos. En este caso, la pasarela detectará dos eventos sucesivos: en primer lugar, cuando la señal ha sido reconocida, la señal DTMF, y a continuación, 2 segundos más tarde, la señal de larga duración.

Conexión de larga duración (ld, *long duration connection*): La señal "conexión de larga duración" se detecta cuando se ha establecido una conexión para un tiempo superior a un periodo determinado. El valor por defecto es 1 hora, si bien se puede modificar en el proceso de aprovisionamiento.

El evento se puede detectar en una conexión. Cuando no se especifica ninguna conexión, el evento se aplica a todas las conexiones para el punto extremo, con independencia del momento en que se han creado las conexiones.

Comodín de tonos de teclado (PB) (x, *PB tones wildcard*): El comodín de tonos PB sustituye a cualquier dígito PB entre 0 y 9.

Lote ADSI

Nombre de lote: JS

Código	Nombre de señal	Evento	Señal	Información adicional
adsi(string)	Visualización ADSI	-	BR	

Visualización ADSI (adsi(string)): La interfaz de servicio con visualización analógica (ADSI, *analogue display services interface*) se utiliza principalmente para presentar visualmente el número de teléfono del originador. Véase 4.2, Funciones de recepción para el número de teléfono del originador (presentación visual del número), de la Referencia técnica de las interfaces del servicio telefónico.

Vídeo

Los lotes de eventos para vídeo se proporcionarán en una versión futura de esta Recomendación.

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie B	Medios de expresión: definiciones, símbolos, clasificación
Serie C	Estadísticas generales de telecomunicaciones
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 multimedia
Serie I	Red digital de servicios integrados
Serie J	Redes de cable y transmisión de programas radiofónicos y televisivos, y de otras señales multimedia
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	RGT y mantenimiento de redes: sistemas de transmisión, circuitos telefónicos, telegrafía, facsímil y circuitos arrendados internacionales
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 y comunicación entre sistemas abiertos
Serie Y	Infraestructura mundial de la información y aspectos del protocolo Internet
Serie Z	Lenguajes y aspectos generales de soporte lógico para sistemas de telecomunicación