



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

**UIT-T**

SECTOR DE NORMALIZACIÓN  
DE LAS TELECOMUNICACIONES  
DE LA UIT

**J.171**

(02/2002)

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

IPCablecom

---

**Protocolo de control de pasarela de circuitos  
troncales IPCablecom**

Recomendación UIT-T J.171

---

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
<b>IPCablecom</b>	<b>J.160–J.179</b>
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.171**

### **Protocolo de control de pasarela de circuitos troncales IPCablecom**

#### **Resumen**

Esta Recomendación describe perfiles IPCablecom de una interfaz de programación de aplicación (API, *application programming interface*) para el control de pasarelas de medios de la RTPC de voz por IP (VoIP, *voice-over-IP*), desde elementos de control de llamada externos.

#### **Orígenes**

La Recomendación UIT-T J.171, 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 13 de febrero de 2002.

## 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

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

## ÍNDICE

	<b>Página</b>
1 Alcance.....	1
2 Referencias.....	1
2.1 Referencias normativas.....	1
2.2 Referencias informativas.....	1
3 Términos, definiciones, acrónimos y abreviaturas.....	2
3.1 Definiciones.....	2
3.2 Abreviaturas.....	2
3.3 Convenios.....	3
Anexo A – Perfil 1 de TGCP.....	4
A.1 Alcance.....	4
A.2 Interfaz de control de pasarela de medios (MGCI, <i>media gateway control interface</i> ).....	4
A.2.1 Modelo y convenios de denominación.....	4
A.2.2 Utilización de SDP.....	11
A.2.3 Funciones de control de pasarela.....	11
A.2.4 Estados, condiciones de cambio-por-fallo y pugna por llegar primero (carrera).....	32
A.2.5 Códigos de retorno y códigos de error.....	43
A.2.6 Códigos de motivo.....	44
A.3 Protocolo de control de pasarela de medios.....	44
A.3.1 Descripción general.....	45
A.3.2 Encabezamiento de instrucción.....	45
A.3.3 Formatos de encabezamiento de respuesta.....	57
A.3.4 Codificación de descripción de sesión.....	61
A.3.5 Transmisión por UDP.....	67
A.3.6 Remolque.....	68
A.3.7 Identificadores de transacción y triple toma de contacto.....	69
A.3.8 Respuestas provisionales.....	70
A.4 Seguridad.....	71
Anexo A.A – Lotes de eventos.....	72
A.A.1 Lote de circuitos troncales SUP.....	73
Apéndice A.I – Interacciones de modos.....	75
Modo de la conexión B.....	76
Apéndice A.II – Ejemplos de codificación de instrucciones.....	77
A.II.1 NotificationRequest.....	77
A.II.2 Notify.....	77

	<b>Página</b>
A.II.3 CreateConnection .....	77
A.II.4 ModifyConnection .....	79
A.II.5 DeleteConnection (procedente del controlador de pasarela de medios (MGC)).....	79
A.II.6 DeleteConnection (procedente de la pasarela de circuitos troncales).....	80
A.II.7 DeleteConnection (procedente del controlador de pasarela de medios, en caso de múltiples conexiones).....	80
A.II.8 AuditEndpoint .....	80
A.II.9 AuditConnection .....	81
A.II.10 RestartInProgress .....	82
Apéndice A.III – Ejemplo de flujo de llamadas .....	83
Apéndice A.IV – Requisitos de los puntos extremos .....	86
A.IV.1 Modos de conexión soportados .....	86
Apéndice A.V – Información sobre compatibilidad.....	86
A.V.1 Compatibilidad con NCS .....	86
A.V.2 Compatibilidad con MGCP .....	87
Apéndice A.VI – Ejemplos de lotes de eventos .....	88
A.VI.1 Lote de servicios de operador MF FGD.....	88
A.VI.2 Lote protocolo de terminación MF.....	91
Apéndice A.VII – Bibliografía .....	94
Anexo A.B – Perfil 2 de TGCP .....	94

# Recomendación UIT-T J.171

## Protocolo de control de pasarela de circuitos troncales IPCablecom

### 1 Alcance

Esta Recomendación describe perfiles IPCablecom de una interfaz de programación de aplicación (API, *application programming interface*) para el control de pasarelas de medios de la RTPC de voz por IP (VoIP, *voice-over-IP*), desde elementos de control de llamada externos.

En anexos a esta Recomendación se especifican perfiles para esta finalidad.

### 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

- Recomendación 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.*
- Recomendación UIT-T J.162 (2001), *Protocolo de señalización de llamada de red para la prestación de servicios dependientes del tiempo por redes de televisión por cable que utilizan módems de cable.*
- Recomendación UIT-T J.170 (2002), *Especificación de seguridad de IPCablecom.*
- IETF RFC 2327 (1998), *SDP: Session Description Protocol.*

NOTA – El hecho de que en esta Recomendación se haga referencia a un documento no le confiere, en su carácter de documento individual, la categoría de una Recomendación.

#### 2.2 Referencias informativas

- IETF Internet Draft (draft-huitema-sgcp-v1-02.txt), *Simple Gateway Control Protocol (SGCP).*
- IETF Internet Draft (draft-taylor-ipdc-00.txt), *IPDC Base Protocol.*
- IETF RFC 1889 (1996) *RTP: A Transport Protocol for Real-Time Applications.*
- IETF RFC 1890 (1996), *RTP: Profile for Audio and Video Conferences with Minimal Control.*
- IETF RFC 2543 (1999), *SIP: Session Initiation Protocol.*
- IETF RFC 2326 (1998), *Real Time Streaming Protocol (RTSP).*
- Recomendación UIT-T E.180/Q.35 (1998), *Características técnicas de los tonos para el servicio telefónico.*
- Recomendación UIT-T Q.761 (1999), *Sistema de señalización N.º 7 – Descripción funcional de la parte usuario de la RDSI.*

- Recomendación UIT-T Q.762 (1999), *Sistema de señalización N.º 7 – Funciones generales de los mensajes y señales de la parte usuario de la RDSI.*
- Recomendación UIT-T H.323 (2000), *Sistemas de comunicación multimedios basados en paquetes.*
- Recomendación UIT-T H.225.0 (2000), *Protocolo de señalización de llamada y paquetización de trenes de medios para sistemas de comunicación multimedios por paquetes.*
- Recomendación UIT-T H.245 (2001), *Protocolo de control para comunicación multimedios.*
- IETF RFC 1825 (1995), *Security Architecture for the Internet Protocol.*
- IETF RFC 1826 (1995), *IP Authentication Header.*
- IETF RFC 2705 (1999), *Media Gateway Control Protocol (MGCP) Version 1.0.*
- TCP/IP Illustrated, volume 1 (2001), *The Protocols*, Addison-Wesley, 1994.
- Recomendación UIT-T J.163 (2001), *Calidad de servicio dinámica para la prestación de servicios en tiempo real por las redes de televisión por cable que utilizan módems de cable.*

### 3 Términos, definiciones, acrónimos y abreviaturas

#### 3.1 Definiciones

En esta Recomendación se definen los siguientes términos:

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

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

#### 3.2 Abreviaturas

En esta Recomendación se utilizan las siguientes siglas:

DNS	Sistema de nombres de dominio ( <i>domain name system</i> )
IP	Protocolo Internet ( <i>Internet protocol</i> )
IPSec	Seguridad del protocolo Internet ( <i>Internet protocol security</i> )
MGC	Controlador de pasarela de medios ( <i>media gateway controller</i> )
MGCP	Protocolo de control de pasarela de medios ( <i>media gateway control protocol</i> )
MIB	Base de información de gestión ( <i>management information base</i> )
MTA	Adaptador de terminal de medios ( <i>media terminal adapter</i> )
MWD	Máximo tiempo de espera ( <i>maximum waiting delay</i> )
NCS	Señalización de llamada basada en la red ( <i>network-based call signaling</i> )
NTP	Protocolo de tiempo de red ( <i>network time protocol</i> )
PU-RDSI	Parte usuario de la red digital de servicios integrados
QoS	Calidad de servicio ( <i>quality of service</i> )
RTCP	Protocolo de control en tiempo real ( <i>real time control protocol</i> )



RTO	Temporizador de retransmisión ( <i>retransmission timeout</i> )
RTP	Protocolo en tiempo real ( <i>real time protocol</i> )
SDP	Protocolo de descripción de sesión ( <i>session description protocol</i> )
SG	Pasarela de señalización ( <i>signaling gateway</i> )
SPI	Índice de parámetros de seguridad ( <i>security parameters index</i> )

### 3.3 Convenios

Al implementar esta Recomendación, se tendrá en cuenta que la obligatoriedad de la especificación se expresa mediante el verbo modal "deber" (modal inglés *MUST*) o un verbo en tiempo futuro con valor imperativo, por ejemplo "expirará" (modal inglés *SHALL*) o el adjetivo "OBLIGATORIO" (*REQUIRED*).

A continuación, se indican otras expresiones que se aplican a determinados requisitos con significado de obligación o posibilidad.

**DEBER (*must*)** Este verbo (u otros con significado de obligación, como "tener que/de", "haber que/de") o un verbo en tiempo futuro con valor imperativo o el adjetivo OBLIGATORIO (*REQUIRED*, *MANDATORY*) indican que se tiene la obligación de hacer lo que expresa la especificación.

**NO DEBER (*must not*)** La negación indica que se prohíbe hacer lo que expresa la especificación.

**DEBERÍA (*should*)** El modo condicional de estos verbos, u otros verbos con significado de conveniencia (aconsejar, recomendar, ser conveniente) o el adjetivo RECOMENDADO (*RECOMMENDED*) indica que puede haber motivos fundados para que en determinadas circunstancias no se haga cierta cosa, pero que antes de hacer algo diferente, es preciso entender todas las consecuencias y sopesar el caso.

**NO DEBERÍA (*should not*)** La negación indica la posibilidad de que haya motivos fundados para que en determinadas circunstancias la acción sea aceptable e incluso útil, pero que antes de realizarla es preciso entender todas las consecuencias y sopesar el caso.

**PODER (*may*)** Éste u otros verbos que indican posibilidad o probabilidad (deber de) o el adjetivo OPCIONAL (*OPTIONAL*) se refieren a la libertad de elegir. Un proveedor puede incluir un elemento porque el mercado lo exige o porque mejora el producto, mientras que otro puede optar por no hacerlo.

## Anexo A

### Perfil 1 de TGCP

#### A.1 Alcance

Este anexo describe un perfil IPCablecom de una interfaz de programación de aplicación (API, *application programming interface*) denominada interfaz de control de pasarela de medios (MGCI, *media gateway control interface*) y un protocolo correspondiente (MGCP) para el control de pasarelas RTPC de voz por IP (VoIP, *voice-over-IP*) desde elementos de control de llamada externos. MGCP presupone una arquitectura de control de llamada en la que la "inteligencia" de control de llamada esta fuera de las pasarelas y es manejada por elementos de control de llamada externos. El perfil IPCablecom descrito en este anexo se designará por el Protocolo de control de pasarela de circuitos troncales (TGCP, *trunking gateway control protocol*) IPCablecom.

Este anexo se basa en la Rec. UIT-T J.162 sobre señalización de llamada basada en la red IPCablecom, y en IETF RFC 2705 Media Gateway Control Protocol (MGCP). El presente anexo, que define el protocolo TGCP IPCablecom, es una especificación es independiente del MGCP. El perfil TGCP del MGCP está definido estricta y únicamente por el contenido de este anexo.

NOTA – La especificación contenida en este anexo se utiliza en América del Norte.

#### A.2 Interfaz de control de pasarela de medios (MGCI, *media gateway control interface*)

Las funciones MGCI proporcionan control de conexión, control de punto extremo, auditorización, e informes de status. Estas funciones utilizan el mismo modelo de sistema y los mismos convenios de denominación.

##### A.2.1 Modelo y convenios de denominación

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

##### A.2.1.1 Nombres de punto extremo

Los nombres de punto extremo, llamados también identificadores de punto extremo, tienen dos componentes que, por definición, son insensibles a la escritura en mayúscula o minúscula:

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

Los nombres de pasarela serán de la forma

`local-endpoint-name@domain-name`

donde `domain-name` es un nombre de dominio absoluto tal como se define en IETF RFC 1034 e incluye una porción anfitrión (host), por lo que un ejemplo de nombre de dominio podría ser:

`MyTrunkingGateway.cablelabs.com`

Asimismo, `domain-name` puede ser una dirección IPv4 en forma de expresión con puntos decimales, representada como una cadena de texto y encerrada entre un corchete izquierdo y un corchete derecho ("[" y "]") como en "[128.96.41.1]"; para una información detallada, consúltese IETF RFC 821. En cambio, la utilización de direcciones IP está desaconsejada de manera general.

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

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

`term1/term2/term3`

y si para uno de los términos del nombre de punto extremo local se usa un comodín, el nombre de punto extremo local tiene el siguiente aspecto:

`term1/term2/*` si se usa un comodín para `term3`.

`term1/*/*` si se usa un comodín para `term2` y `term3`.

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

- El uso de comodín sólo está permitido desde la derecha; por tanto, si se usa un comodín para un término, el comodín abarca los términos a la derecha de éste.
- En aquellos casos en que se utilizan en forma mixta los comodines signo de dólar y asterisco, los signos de dólar sólo están permitidos desde la derecha; por tanto, si un término tiene un comodín signo de dólar, todos los términos a la derecha de ese término deben también contener comodines signo de dólar.
- Un término representado por un asterisco debe interpretarse como: "utilizar *todos* los valores de este término conocidos dentro del alcance de la pasarela de circuitos troncales en cuestión".
- Un término representado por un signo de dólar debe interpretarse como: "utilizar *cualquier* valor de este término que sea conocido dentro del alcance de la pasarela de circuitos troncales en cuestión".
- Cada punto extremo puede especificar detalles adicionales en las reglas de denominación para ese tipo de punto extremo; sin embargo, dichas reglas no podrán estar en contradicción con lo anteriormente expuesto.

Debe señalarse que si se utilizan tipos de punto extremo diferentes, o incluso si se utilizan términos secundarios diferentes, por ejemplo, "líneas, dentro del mismo tipo de punto extremo, el resultado será dos nombres de punto extremo local diferentes. En consecuencia, cada "línea" se tratará como un punto extremo distinto.

#### **A.2.1.1.1 Nombres de punto extremo de pasarela de circuitos troncales**

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

Las pasarelas de circuitos troncales soportarán el siguiente endpoint-type (tipo de punto extremo) básico:

- `ds` Un circuito troncal DS-0

Cabe esperar que el tipo de punto extremo básico se suministre con información adicional sobre el tipo de señalización soportado en el circuito troncal y el papel de sistema de conmutación que desempeña.

Puntos extremos de circuito troncal:

Además de los convenios de denominación especificados anteriormente, los nombre de punto extremo local para puntos extremos de pasarela de circuitos troncales RTPC de tipo "ds" observarán las siguientes reglas:

- Los nombre de punto extremo local consistirán en una serie de términos separados entre sí por una barra oblicua ("/"), que describen la jerarquía física dentro de la pasarela:  
`ds/<unit-type1>-<unit #>/<unit-type2>-<unit #>/.../<channel #>`
- El primer término (`ds`) identifica el esquema de denominación de puntos extremos utilizado y el tipo de punto extremo básico.
- El último término es un número decimal que indica el número de *canal*<sup>1</sup> en el nivel más bajo de la jerarquía.
- Los términos intermedios entre el primer término (`ds`) y el último término (número de canal) representan niveles intermedios de la jerarquía y consisten en `<unit-type>` y `<unit #>` separados por un guión ("-") donde:
  - el `<unit-type>` identifica el nivel jerárquico particular. Son valores de `<unit-type>` actualmente definidos: "s", "su", "oc3", "ds3", "e3", "ds2", "e2", "ds1", "e1" donde "s" indica número de intervalo de tiempo y "su" indica una subunidad dentro de un intervalo de tiempo. También se permitirán otros valores que representan niveles jerárquicos físicos que no han sido incluidos en esta lista, pero que siguen las mismas reglas de denominación.
  - el `<unit #>` es un número decimal que se utiliza para hacer referencia a una instancia particular de un `<unit-type>` en ese nivel de la jerarquía.
- El número de niveles y la denominación de esos niveles se basa en la jerarquía física dentro de la pasarela de medios, como se muestra en los siguientes ejemplos:
  - Una pasarela de medios que tiene cierto número de interfaces DS1:  
`ds/ds1-#/#`
  - Una pasarela de medios que tiene cierto número de interfaces OC3, que contienen jerarquías DS3 y DS1 canalizadas:  
`ds/oc3-#/ds3-#/ds1-#/#`
  - Una pasarela de medios que contiene cierto número de intervalos de tiempo, cada uno de los cuales tiene cierto número de interfaces DS3:  
`ds/s-#/ds3-#/ds1-#/#`
- Algunos puntos extremos pueden no contener todos los niveles posibles de una jerarquía; sin embargo, todos los niveles soportados por un punto extremo dado están contenidos en el esquema de denominación de puntos extremos. Por ejemplo, un DS3 sin entramado DS1 podría representarse por el siguiente esquema de denominación:

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

---

<sup>1</sup> Obsérvese la utilización del término "channel" (canal) por oposición a "timeslot" (intervalo de tiempo).

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

- La denominación mediante uso de comodines sigue los convenios indicados en A.2.1.1, donde el carácter asterisco ("\*") hace referencia a "all" (todos), y el carácter signo de dólar ("\$") hace referencia a "any" (cualquiera). También está soportado un convenio de uso de comodines en forma de una gama "[N-M]" que representa una "gama" ("range") de canales desde el canal N hasta el canal M inclusive:
  - Debe señalarse que la utilización del comodín "all" para el primer término (ds) hace referencia a todos los tipos de punto extremo en la pasarela de medios cualquiera que sea su tipo. Se espera que, generalmente, esta posibilidad será utilizada para fines administrativos, por ejemplo, auditorización o rearranque.
  - Un nombre de punto extremo local puede ser subespecificado suministrando un número de términos, menor que el normal, comenzando por la izquierda del nombre de punto extremo. En ese caso, se supone que los términos que faltan a la derecha del último término especificado son el carácter comodín "\*", referente a "all of" (la totalidad de), a menos que los términos especificados contengan el carácter comodín "any of" (cualquiera de), en cuyo caso se supone que los términos que faltan a la derecha del último término especificado son el carácter comodín "any of".
  - Dondequiera que se permita el uso del comodín "all", el comodín gama de canales "[N-M]" puede utilizarse en el último término (es decir, <channel-#>) del nombre de punto extremo local, en vez de aquél. El comodín "range" (gama) hará entonces referencia a todos los canales de N a M. Las reglas y restricciones aplicables a la utilización del comodín "all" también son aplicables a la utilización del comodín "range".

Los siguientes ejemplos ilustran la utilización de comodines:

ds/ds1-3/*	Todos los canales en ds1 número 3 de la pasarela de medios en cuestión.
ds/ds1-3/\$	Cualquier canal en ds1 número 3 de la pasarela de medios en cuestión.
ds/*	Todos los puntos extremos de circuitos troncales de la pasarela de medios en cuestión.
*	Todos los puntos extremos (cualquiera que sea el tipo de punto extremo) de la pasarela de medios en cuestión.
ds/ds1-3/[1-24]	Canales 1 a 24 en ds1 número 3 de la pasarela de medios en cuestión.

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

### A.2.1.2 Nombres de llamada

Los nombres de llamada se identifican por identificadores únicos, independientes de las plataformas o agentes subyacentes. Los identificadores de llamada son cadenas hexadecimales, creadas por el MGC. La longitud máxima de los identificadores de llamada es 32 caracteres.

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

### A.2.1.3 Nombres de conexión

La pasarela crea identificadores de conexión cuando se le pide que cree una conexión. Estos identificadores identifican la conexión en el contexto de un punto extremo. Los identificadores de conexión se tratan en MGCP como cadenas hexadecimales. La pasarela DEBE asegurarse de que transcurre un periodo de espera adecuado, de al menos tres minutos, entre el momento en que termina una conexión en que se utilizó un identificador y el momento en que se utiliza el mismo identificador en una nueva conexión para el mismo punto extremo. La longitud máxima de un nombre de conexión es de 32 caracteres.

### A.2.1.4 Nombres de controladores de pasarela de medios y de otras entidades

El protocolo de control de pasarela de medios ha sido diseñado con miras a una mayor fiabilidad de la red y a la implementación de MGC redundantes. Esto significa que no existe una vinculación fija entre entidades plataformas de equipo físico o interfaces de red.

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

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

Para asegurar la fiabilidad se toman las siguientes medidas de precaución:

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

Además de la indirección proporcionada por la utilización de nombres de dominio y del DNS, el concepto de "entidad notificada" es esencial a los efectos de la fiabilidad y del cambio-por-fallo en el MGCP. La "entidad notificada" para un punto extremo es el MGC que controla ese punto extremo en ese momento. En cualquier instante dado, un punto extremo tiene asociada una, y sólo una, "entidad notificada" y, cuando necesita enviar una instrucción al MGC, DEBE enviarla a la "parte notificada" a que pertenece el punto extremo o los puntos extremos en ese momento. Tras el arranque inicial, se DEBE fijar la "entidad notificada" a un valor suministrado. La mayor parte de las instrucciones enviadas por el MGC permiten denominar explícitamente la "entidad notificada" mediante la utilización de un parámetro "NotifiedEntity". La "entidad notificada" permanecerá en el mismo estado hasta que se reciba el nuevo parámetro "NotifiedEntity" o hasta que el punto extremo rearranque. Si la "entidad notificada" para un punto extremo está vacía, o no se ha fijado explícitamente a un valor<sup>2</sup>, adoptará como valor por defecto la dirección fuente de la última conexión que ha tratado la instrucción o petición de notificación recibida para el punto extremo. Por tanto, la auditorización no modificará la "entidad notificada".

En A.2.4 se presenta una descripción más detallada de la fiabilidad y el cambio-por-fallo.

---

<sup>2</sup> Esto podría suceder como resultado de la especificación de un parámetro NotifiedEntity vacío.

### A.2.1.5 Mapas de dígitos

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

Ninguno de los tipos de circuito troncal soportados por la actual versión de la Recomendación sobre TGCP necesita mapa de dígitos; por esta razón, los mapas de bits no se han incluido en la presente Recomendación.

### A.2.1.6 Eventos y señales

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

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

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

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

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

Cada lote DEBE tener una definición (de lote), que DEBE definir el nombre del lote, y la definición de cada evento perteneciente al lote. La definición de evento DEBE incluir el nombre preciso del evento, es decir, el código de evento, una definición en texto simple del evento y, cuando proceda, la definición precisa de las señales correspondientes, por ejemplo las frecuencias exactas de las señales de audio como el tono de llamada de retorno o el tono de facsímil. Las definiciones de evento deben asimismo especificar si los eventos son persistentes (véase A.2.3.1) y si contienen estados de evento autorizables (véase A.2.3.8.1). Las señales DEBEN también tener su tipo definido (señales de

activación/desactivación), señales sometidas a la expiración de un periodo de temporización, o señales breves, y las señales sometidas a la expiración de un periodo de temporización DEBEN tener un valor de periodo de temporización por defecto definido (véase A.2.3.1).

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

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

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

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

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

Los eventos y señales se detectan y generan por defecto en puntos extremos; sin embargo, algunos eventos y señales pueden detectarse y generarse además de las conexiones a un punto extremo o en lugar de estas conexiones. Por ejemplo, se puede pedir a los puntos extremos que proporcionen un tono de llamada de retorno en una conexión. Para que un evento o señal pueda ser detectado o generado en una conexión, la definición del evento/señal DEBE especificar explícitamente que el evento/señal puede detectarse o generarse en una conexión.

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

IT/rt@0A3F58

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

IT/ma@\*

El carácter comodín "\$" (signo de dólar) puede utilizarse para designar "la conexión actual". Este convenio NO SE UTILIZARÁ a menos que la petición de notificación de evento esté "encapsulada" dentro de una instrucción CreateConnection o ModifyConnection. Cuando se usa este convenio, la pasarela generará o detectará el evento en la conexión que se está creando o modificando en ese momento. Un ejemplo del uso de este convenio es:

IT/rt@\$

El identificador de conexión, o un comodín sustitutivo, pueden utilizarse junto con los convenios relativos a "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 punto extremo o puntos extremos afectados.



### A.2.2 Utilización de SDP

El MGC utiliza el MGCP para proporcionar a las pasarelas la descripción de parámetros de conexión tales como direcciones IP, puerto UDP, y perfiles RTP. Salvo que se indique o implique otra cosa en esta Recomendación, las descripciones SDP DEBEN seguir los convenios indicados en el protocolo de descripción de sesión (SDP, *session description protocol*), que actualmente constituye una norma IETF propuesta, documentada en IETF RFC 2327.

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

### A.2.3 Funciones de control de pasarela

Esta cláusula describe las instrucciones del MGCP en forma de una llamada de procedimiento a distancia (RPC, *remote procedure call*) como una interfaz de programación de aplicación (API, *application programming interface*), a la que se hará referencia como la interfaz de control de pasarela de medios (MGCI, *media gateway control interface*). Para cada instrucción MGCP se define una función MGCI, donde esta función toma y retorna los mismos parámetros que la correspondiente instrucción MGCP. Las funciones mostradas en esta cláusula proporcionan una descripción de alto nivel de la operación de MGCP y describen un ejemplo de una API similar a RPC que PUEDE utilizarse para una implementación de MGCP. Aunque la API de MGCI es simplemente un ejemplo, el comportamiento semántico definido por MGCI es parte integrante de esta Recomendación, y todas las implementaciones DEBEN ser conformes con la semántica especificada para MGCI. Los mensajes efectivamente intercambiados, incluidos los formatos y codificaciones utilizados, se definen en A.3. Las pasarelas de circuitos troncales DEBEN implementarlos exactamente como están especificados.

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

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

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

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

Las señales de audio recibidas del punto extremo se envían por cualquier conexión para ese punto extremo cuyo modo sea "sólo lectura", o "envío/recepción".

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

- Las señales de audio recibidas en paquetes de datos a través de conexiones en los modos "inactivo", "conexión en bucle" o "prueba de continuidad" se descartan.
- Las señales de audio recibidas en paquetes de datos a través de conexiones en los modos "sólo recepción", o "envío/recepción" se mezclan y después se envían al punto extremo<sup>3</sup>.
- Las señales de audio originadas en el punto extremo se transmiten a través de todas las conexiones que funcionen en los modos "sólo envío", o "envío/recepción".
- Las señales de audio recibidas en paquetes de datos a través de conexiones en los modos "conexión en bucle de red" o "prueba de continuidad de red" se enviarán en retorno por la conexión, como se describe más adelante.

Los modos "conexión en bucle" y "prueba de continuidad" se utilizan durante operaciones de mantenimiento y prueba de continuidad. La prueba de continuidad (COT, *continuity test*) tiene dos variantes: una especificada para uso general, y la otra especificada en varias redes nacionales. En la primera variante, la prueba se efectúa mediante conexión en bucle. El conmutador de origen envía un tono (el tono de ida) por el circuito portador y espera que el conmutador de terminación conecte en bucle el circuito. Si el conmutador de origen comprueba que se ha retornado el mismo tono (el tono de retorno), la prueba de continuidad ha tenido éxito. Si no, la prueba de continuidad ha fracasado. En la segunda variante, los tonos de ida y de retorno son diferentes. El conmutador de origen envía cierto tono de ida. El conmutador de terminación lo detecta, e inserta en el sentido de retorno un tono de retorno, que es diferente del tono de ida. Si el conmutador de origen detecta el tono de retorno, la prueba de continuidad ha tenido éxito. Si el conmutador de origen no detecta el tono de retorno dentro de un determinado periodo de tiempo, la prueba de continuidad ha fracasado.

Si el modo está fijado a "conexión en bucle", se prevé que la pasarela envíe la señal entrante, procedente del punto extremo, en retorno al mismo punto extremo. Este es el procedimiento general. Si el modo se ha fijado a "prueba de continuidad", se informa a la pasarela que el otro extremo del circuito ha iniciado un procedimiento de prueba de continuidad de acuerdo con los procedimientos especificados para varias redes nacionales. La pasarela pondrá el circuito en el modo transpondedor requerido para la prueba de continuidad con dos tonos.

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

- Las señales de audio recibidas por cualquier conexión para el punto extremo *no* se enviarán al punto extremo.

---

<sup>3</sup> Actualmente, los puntos extremos TGCP no están obligados a soportar la mezcla de estas señales de audio.

- Las señales de audio recibidas en el punto extremo *no* se enviarán a ninguna conexión para el punto extremo.

Si el modo está fijado a "conexión en bucle de red", las señales de audio recibidas de la conexión serán devueltas en eco por la misma conexión. El modo "conexión en bucle de red" DEBERÍA funcionar simplemente como un reflector de paquetes RTP.

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

Las conexiones nuevas y las conexiones existentes NO SERÁN afectadas por conexiones efectuadas en modo "conexión en bucle de red" o "prueba de continuidad de red". No obstante, constricciones relativas a recursos locales pueden limitar el número de las nuevas conexiones que podrán efectuarse.

Para ejemplos de interacciones de los modos, véase el apéndice A.I.

### A.2.3.1 NotificationRequest

La instrucción NotificationRequest (petición de notificación) se utiliza para pedir a la pasarela que envíe una notificación cuando se produzcan ciertos eventos en un punto extremo. Por ejemplo, se puede pedir que se envíe una notificación cuando en un punto extremo se detectan tonos relacionados con la comunicación de facsímil. La entidad que recibe esta notificación, por lo general el MGC, podrá entonces considerar que se debe utilizar un tipo diferente de codificación en las conexiones vinculadas a este punto extremo y dar a la pasarela la orden correspondiente<sup>4</sup>.

ReturnCode

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

**EndpointId** es el identificador del punto extremo o de los puntos extremos en la pasarela en la que se ejecuta NotificationRequest. El EndpointId sigue las reglas para los nombres de punto extremo especificadas en A.2.1.1. El comodín "any of" NO SE UTILIZARÁ.

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

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

**SignalRequests** es un parámetro que contiene el conjunto de señales cuya aplicación se ha pedido a la pasarela. A menos que se especifique otra cosa, las señales se aplican al punto extremo; sin embargo, algunas señales pueden aplicarse a una conexión. Son ejemplos de señales<sup>5</sup>:

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

<sup>4</sup> La nueva instrucción sería una instrucción ModifyConnection.

<sup>5</sup> Para una lista completa de las señales, véase el anexo A.A.

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

- **On/off (OO)** – Estas señales producen efecto desde que son activadas (conmutadas a On) hasta que son desactivadas (conmutadas a Off). Esto sólo puede suceder como resultado de una nueva SignalRequests cuando la señal es desactivada (véase más adelante). Las señales de tipo OO son, por definición, idempotentes, por lo que múltiples peticiones de activación (o de desactivación) de una determinada señal OO son perfectamente válidas y NO PROVOCARÁN ningún error. Una vez activadas, NO SERÁN desactivadas hasta que el MGC lo ordene explícitamente o el punto extremo rearranque.
- **Time-out (TO)** – Estas señales producen efecto desde que son activadas (conmutadas a On) hasta que, o bien son suprimidas (por la aparición de un evento o por el hecho de no ser incluida en una subsiguiente lista de señales (que puede ser una lista vacía)), o bien transcurre un determinado periodo de temporización. Una señal que desaparece por el transcurso de un periodo de temporización genera un evento "operación completa" (para una definición más elaborada de este evento, véase el anexo A.A.1). Una señal TO podría ser "efectuar llamada MF con un periodo de temporización de 16 segundos". Si se produce un evento antes de la expiración del periodo de 16 segundos, la señal será detenida, por defecto<sup>6</sup>. Si la señal no es detenida, expirará el periodo de temporización para la señal, se detendrá la señal, y se generará un evento "operación completa", que el MGC podrá o no haber solicitado que se le notifique. Si el MGC ha solicitado que se le notifique el evento "operación completa", el evento "operación completa" enviado al MGC incluirá el nombre o nombres de la señal o señales que desaparecieron por temporización<sup>7</sup>. La señal o señales generadas en una conexión incluirán el nombre de esa conexión. Las señales afectadas por un periodo de temporización tienen un valor de temporización por defecto, que puede ser modificado en el proceso de provisión. Asimismo, el periodo de temporización puede proporcionarse como un parámetro de la señal. El valor cero indica que el periodo de temporización es infinito. Una señal TO que fracasa después de haber sido iniciada, pero antes de generar un evento "operación completa", generará un evento "fallo de operación", que incluirá el nombre o nombres de la señal o señales a que se aplica la temporización<sup>7</sup>.
- **Brief (BR)** – La duración de estas señales es tan corta que terminan por sí mismas. Si se produce un evento de detención de la señal, o se aplica una nueva instrucción SignalRequests, una señal BR que se encuentre en curso en ese momento no se detendrá. Sin embargo, se suprimirá toda señal BR pendiente que todavía no haya comenzado a aplicarse.

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

Cuando se aplica una señal a una conexión que ha recibido un RemoteConnectionDescriptor (véase A.2.3.3), el tren de medios generado por esa señal se reenviará por la conexión *cualquiera que sea* el modo de la conexión. Si no se ha recibido un RemoteConnectionDescriptor, la pasarela DEBE retornar un error (código de error 527 – falta RemoteConnectionDescriptor).

Cuando se suministra una lista de señales (que puede estar vacía), esta lista sustituye íntegramente la actual lista de señales sometidas a temporización que están activas. Las señales sometidas a temporización que estén activas y no hayan sido incluidas en la nueva lista DEBEN ser detenidas y la nueva o nuevas señales proporcionadas no serán activadas. Las señales sometidas a temporización

---

<sup>6</sup> La acción "mantener señal(es) activa(s)" puede prevalecer sobre este comportamiento.

<sup>7</sup> Si se pasaron parámetros a la señal, no se informarán los parámetros.

que estén activas en ese momento y que se hayan incluido en la nueva lista de señales DEBEN permanecer activas sin interrupción, y el temporizador para esas señales no será afectado. Por tanto, actualmente no hay forma de rearrancar el temporizador para una señal sometida a temporización que está activa en un momento dado sin antes desactivar la señal. Si la señal de temporización tiene parámetros, el conjunto inicial de parámetros seguirá vigente cualquiera que sean los valores que se suministren posteriormente. Una determinada señal NO APARECERÁ más de una vez en una SignalRequests.

Las señales actualmente definidas se indican en el anexo A.A.

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

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

Los eventos actualmente definidos se indican en el anexo A.A.

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

- notificar el evento inmediatamente, junto con la lista acumulativa de eventos observados;
- acumular el evento;
- ignorar el evento;
- mantener las señales activas;
- NotificationRequest insertada;
- ModifyConnection insertada.

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

Los eventos persistentes siempre se detectan en un punto extremo. Si se produce un evento persistente que no esté incluido en la lista de RequestedEvents, el evento será detectado de todas formas, y procesado al igual que todos los demás, como si dicho evento persistente se hubiera solicitado con una acción Notify<sup>8</sup>. En consecuencia, desde un punto de vista informal, puede considerarse que los eventos persistentes siempre han sido incluidos en la lista de RequestedEvents con una acción para Notify, si bien no se realizará ninguna detección de doble toma, ni otra acción similar<sup>9</sup>. Los eventos persistentes se identifican como tales por su definición (véase el anexo A.A).

Eventos no persistentes son aquellos que deben ser incluidos implícitamente en la lista de RequestedEvents. Una nueva lista (que puede estar vacía) de eventos solicitados sustituye íntegramente a una anterior lista de eventos solicitados. El punto extremo sólo detectará los eventos persistentes, y los eventos que estén especificados en la lista de eventos solicitados. Si un evento persistente está incluido en la lista de RequestedEvents, la acción especificada sustituirá a la acción por defecto asociada con el evento durante el tiempo en que esté vigente la lista de RequestedEvents; una vez transcurrido este periodo de tiempo, se restablece la acción por defecto. Un evento dado NO APARECERÁ más de una vez en una lista de RequestedEvents.

---

<sup>8</sup> Por tanto, el RequestIdentifier será el RequestIdentifier de la NotificationRequest en curso.

<sup>9</sup> Normalmente, cuando, por ejemplo, se solicita la detección de un evento de descolgar, la petición sólo puede tener éxito si el teléfono no está descolgado en ese momento.

Puede especificarse más de una acción para un evento, pero una acción no puede aparecer más de una vez para un evento dado. La siguiente matriz del cuadro A.1 indica las combinaciones legales de acciones:

**Cuadro A.1/J.171 – Combinaciones legales de acciones**

	<b>Notificar</b>	<b>Acumular</b>	<b>Ignorar</b>	<b>Mantener señales activas</b>	<b>Notification Request insertada</b>	<b>ModifyConnection insertada</b>
Notificar	–	–	–	√	–	√
Acumular	–	–	–	√	√	√
Ignorar	–	–	–	√	–	√
Mantener señales activas	√	√	√	–	√	√
NotificationRequest insertada	–	√	–	√	–	√
ModifyConnection insertada	√	√	√	√	√	–

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

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

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

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

Si se desea que la señal o señales sometidas a temporización continúen cuando se produce un evento buscado, puede utilizarse la acción "mantener señales activas". Esta acción tiene por efecto mantener activas todas las señales sometidas a temporización que están activas en ese momento, con lo que se niega la detención por defecto de las señales sometidas a temporización cuando ocurre el evento.

Si se desea que las señales comiencen cuando se produce un evento buscado, puede utilizarse la acción "NotificationRequest insertada". La NotificationRequest insertada puede incluir una nueva lista de RequestedEvents, y nuevas SignalRequests. Sin embargo, la "NotificationRequest insertada" no puede incluir otra "NotificationRequest insertada". Cuando se activa la "NotificationRequest insertada", la lista de eventos observados y la memoria intermedia para cuarentena no serán afectadas (véase A.2.4.3.1).

La acción NotificationRequest insertada permite al MGC establecer un "mini-script" que habrá de ser procesado por la pasarela inmediatamente después de la detección del evento asociado. Toda SignalRequests especificada en la NotificationRequest insertada comenzará inmediatamente. Hay que proceder con suma cautela para evitar discrepancias entre el MGC y la pasarela. Sin embargo, no deberían producirse discrepancias de larga duración, pues las nuevas SignalRequests sustituyen totalmente a la antigua lista de señales sometidas a temporización activas, y las señales de tipo BR siempre se detienen por sí mismas. Se aconseja limitar el número de las señales de tipo On/Off. Se

considera que es una buena práctica, para un MGC, activar ocasionalmente todas las señales de tipo On/Off que deberían estar activadas, y después desactivar todas las señales de dicho tipo que deberían estar desactivadas.

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

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

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

NOTA – En A.2.4.3.1 se presentan detalles adicionales sobre la semántica de la detección e informes de eventos. Se invita al lector a examinarla detenidamente.

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

Las RequestedEvents y SignalRequests generalmente hacen referencia a los mismos eventos. En un caso se pide a la pasarela que detecte la aparición del evento y, en el otro caso, se le pide que genere el evento. Esta regla tiene pocas excepciones, entre las cuales debe señalarse especialmente los tonos de facsímil y de módem, que pueden ser detectados, pero no señalizados. Sin embargo, no debe esperarse que todos los puntos extremos detecten todos los eventos. Los eventos y señales concretos que un determinado punto extremo puede detectar o generar vienen determinados por la lista de lotes de eventos soportados por el punto extremo en cuestión. Cada lote especifica una lista de eventos y señales que pueden ser detectados o aplicados. Una pasarela a la que se pide que detecte o aplique un evento perteneciente a un lote que no está soportado por el punto extremo especificado DEBE retornar un error (código de error 512 o 513 – no equipado para detectar evento o generar señal). Cuando el nombre de evento no está calificado por un nombre de lote, se supone que el nombre de lote es el nombre de lote por defecto para el punto extremo. Si el nombre de evento no está registrado en este lote por defecto, la pasarela DEBE retornar un código de error 522 – no hay tal evento o señal).

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

**QuarantineHandling** es un parámetro facultativo que especifica opciones para el tratamiento de los eventos almacenados en la memoria intermedia de cuarentena (véase A.2.4.3.1). Permite al MGC especificar si los objetos puestos en cuarentena deberán procesarse o descartarse. Si este parámetro está ausente, los eventos puestos en cuarentena DEBEN procesarse.

**DetectEvents** es un parámetro facultativo que especifica una lista mínima de eventos cuya detección en los estados "notification" y "lockstep" se pide a la pasarela. La lista persiste hasta que se notifique un nuevo estado. Una explicación más completa de este parámetro figura en A.2.4.3.1.

**ReturnCode** es un parámetro retornado por la pasarela. Indica el resultado de la instrucción y consiste en un número entero (véase A.2.5) facultativamente seguido de un comentario.

### A.2.3.2 Notificaciones

Para notificar un evento observado, la pasarela envía notificaciones por medio de la instrucción Notify:

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

**EndpointId** es el nombre del punto extremo en la pasarela que emite la instrucción Notify, como se define en A.2.1.1. El identificador DEBE ser un nombre de punto extremo completamente calificado, que incluye el nombre de dominio de la pasarela. La parte local del nombre NO SEGUIRÁ el convenio sobre uso de comodines.

**NotifiedEntity** es un parámetro facultativo que identifica la entidad a la que se envía la notificación. Este parámetro es idéntico al parámetro NotifiedEntity de la NotificationRequest que ocasionó esta notificación. El parámetro está ausente si no había tal parámetro en la petición que ocasión la notificación. Cualquiera que sea el valor del parámetro "NotifiedEntity", la notificación DEBE enviarse a la "entidad notificada" actual para el punto extremo.

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

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

**ReturnCode** es un parámetro retornado por el MGC. Indica el resultado de la instrucción y consiste en un número entero (véase A.2.5) seguido facultativamente de un comentario.



### A.2.3.3 CreateConnection

Esta instrucción se utiliza para crear una conexión.

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

Esta función se utiliza cuando se establece una conexión entre dos puntos extremos. Una conexión se define por sus atributos y por los puntos extremos que asocia. Los parámetros de entrada de CreateConnection proporcionan los datos necesarios para construir la "visión" de uno de los dos puntos extremos de una conexión.

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

**EndpointId** es el identificador para el punto extremo de la pasarela en que se ejecuta la instrucción CreateConnection. El EndpointId puede especificarse completamente asignando al parámetro EndpointId un valor sin comodín en la invocación de la función, o se puede especificar de una forma menos estricta utilizando el convenio de comodín "anyone". Si el punto extremo está subespecificado, el identificador de punto extremo será asignado por la pasarela y su valor completo será retornado en el parámetro **SpecificEndPointId** de la respuesta. El convenio de uso de comodín "all" NO SE UTILIZARÁ.

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

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

- **Método de codificación:** Una lista de nombres de literal para el algoritmo de compresión (método de codificación/decodificación) utilizado para enviar y recibir medios en la conexión se DEBE especificar con al menos un valor. Los elementos de la lista están ordenados según su nivel de preferencia. El punto extremo DEBE elegir exactamente uno de los códecs, y el códec se DEBERÍA elegir de acuerdo con el nivel de preferencia indicado. Si el punto extremo recibe por la conexión cualquier medio codificado por un método de codificación diferente PUEDE descartarlo. El punto extremo también DEBE indicar cuál de los algoritmos de compresión restantes desea soportar como algoritmos alternativos. Para más detalles, véase A.3.4.1). En un documento aparte relativo a IPCablecom se especifica una lista de los métodos de codificación admisibles.

- **Periodo de paquetización:** El periodo de paquetización, en milisegundos, definido en la norma SDP (IETF RFC 2327) DEBE ser especificado; este periodo debe tener un solo valor. Este valor sólo atañe a los medios que se envían. En una Recomendación aparte relativa a IPCablecom se especifica una lista de los periodos de paquetización admisibles.
- **Compensación de eco:** Este campo indica si habrá o no de utilizarse compensación de eco en el lado circuito troncal<sup>10</sup>. El parámetro puede tener el valor "on" (cuando se solicita compensación de eco) u "off" (cuando no se solicita). Este parámetro es facultativo. Cuando se omite el parámetro, la pasarela de circuitos troncales DEBE aplicar compensación de eco.
- **Tipo de servicio** Especifica la clase de servicio que se utilizará para el envío de medios por la conexión, codificando el valor del tipo de servicio del parámetro del encabezamiento IP como dos dígitos hexadecimales. El parámetro es facultativo. Cuando se omite este parámetro, se aplica un valor por defecto de A0<sub>H</sub> que corresponde a un valor de cinco fijado para los bits de precedencia IP.
- **Supresión (de los periodos de) silencio:** Este campo indica si se debe o no utilizar supresión de silencio en el sentido de ida. El parámetro puede tener el valor "on" (supresión de silencio) u "off" (sin supresión de silencio). Este parámetro es facultativo. Cuando se omite este parámetro, no hay supresión de silencio.

Además, se utilizan los siguientes campos LocalConnectionOptions para el soporte de servicios de seguridad IPCablecom:

- **Secreto:** El campo facultativo Secreto es un valor de semilla que DEBE utilizarse para derivar claves de criptación de extremo a extremo para los servicios de seguridad RTP y RTCP especificados en la Rec. UIT-T J.170 sobre seguridad de IPCablecom. El secreto DEBERÍA codificarse como texto claro si sólo contiene valores de caracteres ASCII comprendidos en la gama 21<sub>H</sub> a 7E<sub>H</sub>. En todos los demás casos, el secreto DEBE codificarse utilizando un procesamiento de codificación de base 64. Si no se suministra ningún valor, o si se omite el parámetro y han de proporcionarse servicios de seguridad, el punto extremo DEBE generar su propio secreto<sup>11</sup>. Cuando el MGC suministra un secreto, DEBERÍA utilizarse este secreto.
- **Serie de texto cifrado RTP:** Una lista de series de texto cifrado para seguridad RTP, ordenadas según su nivel de preferencia. El primer elemento entre los indicados en la lista es el preferido. El punto extremo DEBE elegir exactamente una de las series de texto cifrado. El punto extremo también DEBERÍA indicar cuáles de las restantes series de texto cifrado desea soportar como series de texto cifrado alternativas (para más detalles, véase A.3.4.1). Cada serie de texto cifrado se representa por una cadena de caracteres ASCII formada por dos subcadenas (que pueden estar vacías), separadas por una barra oblicua ("/"), de las cuales la primera identifica el algoritmo de autenticación y la segunda el algoritmo de criptación. En la Rec. UIT-T J.170 se especifica una lista de series de texto cifrado admisibles.
- **Serie de texto cifrado RTCP:** Una lista de series de texto cifrado para seguridad RTCP, ordenadas por su nivel de preferencia. El primer elemento entre los indicados en la lista es el preferido. El punto extremo DEBE elegir exactamente una de las series de texto cifrado. El punto extremo también DEBERÍA indicar cuáles de las restantes series de texto cifrado desea soportar como series de texto cifrado alternativas (para más detalles, véase A.3.4.1). Cada serie de texto cifrado se representa por una cadena de caracteres ASCII formada por dos subcadenas (que pueden estar vacías) separadas por una barra oblicua ("/"), de las cuales

<sup>10</sup> La compensación de eco en el lado paquetes no está soportada.

<sup>11</sup> Incluye tanto la generación de un nuevo secreto, como la utilización de un secreto suministrado en un RemoteConnectionDescriptor.

la primera identifica el algoritmo de autenticación y la segunda el algoritmo de criptación. En la Rec. UIT-T J.170 sobre la seguridad de IPCablecom se especifica una lista de series de texto cifrado admisibles.

Además, el TGCP soporta la vigilancia electrónica de IPCablecom (véase PKT-SP-ESP-I01-991229). Cuando una conexión está sometida a vigilancia electrónica, todos los paquetes de medios válidos recibidos a través de la conexión y todos los paquetes de medios enviados a través de la conexión serán replicados y reenviados a una función de entrega de vigilancia electrónica<sup>12</sup> tras la inclusión de un identificador de conexión de contenido de llamada. La replicación seguirá el modo de conexión utilizado para la conexión, con la excepción de que un medio generado por señales se aplica a la conexión, que será replicada cualquiera que sea el modo de conexión. Por ejemplo, una conexión en modo "inactivo" no generará ningún medio interceptado<sup>13</sup>, en tanto que una conexión en modo "sólo envío" sólo generará medios interceptados en el sentido de ida. Los paquetes replicados no se incluirán en las estadísticas de la conexión. Los siguientes campos LocalConnectionOptions se utilizan para el soporte de la vigilancia electrónica de IPCablecom Electronic Surveillance (para más detalles, véase PKT-SP-ESP-I01-991229):

- **Identificador de conexión de contenido de llamada:** El identificador de conexión de contenido de llamada (*CCC, call content connection*) es un valor de 32 bits que especifica el identificador de conexión de contenido de llamada que habrá de utilizarse para la conexión sometida a vigilancia electrónica. Se añadirá al encabezamiento de paquetes vocales interceptados.
- **Destino de contenido de llamada:** El destino de contenido de llamada especifica una dirección Ipv4 seguida del carácter dos puntos (":") y un número de puerto UDP. El destino de contenido de llamada especifica la dirección IP de destino y el puerto para el contenido de llamada interceptado.

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

El **RemoteConnectionDescriptor** es el 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 las LocalConnectionOptions), es decir, los campos que describen una sesión de acuerdo con la norma SDP. La cláusula A.3.4 describe detalladamente el uso soportado de SDP en el perfil TGCP. Este parámetro puede tener un valor nulo cuando no se conoce la información para el extremo distante. Esto sucede porque la entidad que construye una conexión comienza enviando una instrucción CreateConnection a una de las dos pasarelas que intervienen. Para la primera CreateConnection emitida no hay información disponible sobre el otro lado de la conexión. Esta información puede proporcionarse posteriormente invocando ModifyConnection.

El perfil TGCP supone actualmente que los mismos parámetros de medio se aplican a una conexión en el sentido de envío y en el sentido de recepción. Parte de la información del RemoteConnectionDescriptor es por tanto redundante. por lo que existe la posibilidad de inconsistencia con las LocalConnectionOptions. Sin embargo, es puramente responsabilidad del MGC asegurarse de que envía instrucciones coherentes a cada punto extremo para garantizar que se especifican parámetros de medio coherentes. Si, no obstante, una pasarela detecta inconsistencia, las LocalConnectionOptions tendrán simplemente precedencia. Cuando se modifican códecs en el curso de una comunicación, pueden existir cortos periodos de tiempo en los que los puntos extremos utilizan códecs diferentes. Como se ha expresado antes, las pasarelas de circuitos troncales PUEDEN

---

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

<sup>13</sup> Suponiendo que no se aplicó a la conexión una señal generadora de medios.

descartar cualquier medio recibido que haya sido codificado con un códec diferente del especificado en las LocalConnectionOptions para una conexión.

**Modo** indica el modo de funcionamiento para este lado de la conexión. Las opciones son "sólo envío", "sólo recepción", "envío/recepción", "inactivo", "conexión en bucle de red" y "prueba de continuidad de red". El tratamiento de estos modos se especifica al principio de A.2.3. Algunos puntos extremos pueden no estar aptos para soportar todos los modos (véase A.V.1). Si la instrucción especifica un modo no soportado por el punto extremo, se DEBE retornar un error (código de error 517 – modo no soportado). Asimismo, si una conexión todavía no ha recibido un RemoteConnectionDescriptor, se DEBE retornar un error si se intenta efectuar la conexión en cualquiera de los modos "sólo envío", o "envío/recepción" (código de error 527 – falta el RemoteConnectionDescriptor).

**ConnectionId** es un parámetro retornado por la pasarela que identifica unívocamente la conexión en el contexto del punto extremo en cuestión.

**LocalConnectionDescriptor** es un parámetro retornado por la pasarela, constituido por una descripción de sesión que contiene información sobre, por ejemplo, direcciones y puertos RTP para conexiones "IN" definidas en SDP. Es similar al RemoteConnectionDescriptor, del que sólo se diferencia en que especifica este lado de la conexión. En A.3.4 se da información detallada sobre la utilización soportada de SDP en el perfil de TGCP.

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

- Si el modo se fijó a "sólo recepción", la pasarela DEBE aceptar las señales vocales recibidas por la conexión y transmitir las al punto extremo.
- Si el modo se fijó a "inactivo", "conexión en bucle", o "prueba de continuidad", la pasarela DEBE (como siempre) descartar las señales vocales recibidas por la conexión.
- Si el modo se fijó a "conexión en bucle de red" o "prueba de continuidad de red", la pasarela DEBE efectuar la devolución en eco o la respuesta esperadas. El medio devuelto en eco o generado DEBE enviarse entonces a la fuente del medio recibido.

Obsérvese que, cuando el punto extremo no tiene un RemoteConnectionDescriptor para la conexión, la conexión, por definición, no puede estar en el modo "sólo envío", ni en el modo "envío/recepción".

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

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

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

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

Cuando estos parámetros están presentes, la creación de la conexión y la petición de notificación DEBEN estar sincronizadas, lo que significa que o bien ambas son aceptadas, o bien ambas son rechazadas. En este ejemplo, la instrucción `CreateConnection` debe ser rechazada si la pasarela no tiene suficientes recursos o no puede obtener recursos adecuados de la red de acceso local. La petición de notificación de iniciación de la llamada deberá rechazarse en la condición de doble toma si el circuito ya ha sido tomado. En este ejemplo, la llamada no podrá efectuarse si la conexión no puede establecerse, y la conexión no se efectuará si el circuito ya está tomado. En tal caso, se retornaría un error (código de error 401 – circuito ya tomado), por el que se informa al MGC la condición de doble toma.

**ReturnCode** es un parámetro retornado por la pasarela. Indica el resultado de la instrucción y consta de un número entero (véase A.2.5) seguido facultativamente de un comentario.

#### A.2.3.4 **ModifyConnection**

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

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

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

El **EndpointId** DEBE ser un nombre de punto extremo totalmente calificado. En el nombre local NO SE UTILIZARÁ el convenio sobre uso de comodín.

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

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

La instrucción sólo retornará un **LocalConnectionDescriptor** si se modifican parámetros de conexión local, tales como puertos RTP, etc. En consecuencia, si, por ejemplo, sólo se modifica el modo de la conexión, no se retornará un `LocalConnectionDescriptor`. Si se omite un parámetro de conexión, por ejemplo, modo o supresión de silencio, se retendrá el antiguo valor de ese parámetro, si es posible. Si una modificación de parámetro requiere la modificación de uno o más parámetros *no especificados*, la pasarela puede escoger libremente valores adecuados para los parámetros no especificados que deban modificarse<sup>14</sup>.

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 MGC<sup>15</sup>. Cuando se da información de dirección RTP a una pasarela de circuitos troncales para una conexión, la pasarela de circuitos troncales DEBERÍA aceptar solamente trenes de medios (y RTCP) de la dirección RTP especificada. Todo otro tren de medios recibido de cualquier otra dirección DEBERÍA descartarse. Para otros requisitos de seguridad se debe consultar la Rec. UIT-T J.170.

Los parámetros **RequestedEvents**, **RequestIdentifier**, **SignalRequests**, **QuarantineHandling**, y **DetectEvents** son facultativos. El MGC puede utilizar estos parámetros para incluir una petición de notificación que se vincula a la modificación de la conexión y se ejecuta simultáneamente con ella. Si se suministra uno o más de estos parámetros, el `RequestIdentifier` DEBE ser uno de ellos.

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

**ReturnCode** es un parámetro retornado por la pasarela. Indica el resultado de la instrucción y está constituido por un número entero (véase A.2.5) seguido facultativamente de un comentario.

### A.2.3.5 **DeleteConnection (desde el controlador de pasarela de medios)**

Esta instrucción se utiliza para terminar un conexión. Como efecto secundario toma datos estadísticos sobre la ejecución de la conexión.

---

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

<sup>15</sup> Por ejemplo, si el medio tiene que atravesar un guardafuego.

```

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

```

En esta forma de la instrucción DeleteConnection, el identificador de punto extremo DEBE estar totalmente calificado. Los convenios sobre el uso de comodín NO SE UTILIZARÁN.

En el caso general en el que una conexión tiene dos extremos, esta instrucción hay que enviarla a las dos pasarelas que intervienen en la conexión. Una vez suprimida la conexión, los trenes de medios de la red de paquetes anteriormente soportados por la conexión dejan de estar disponibles. Todo paquete de medios recibido para la antigua conexión se descarta, simplemente, y no se envía ningún nuevo paquete de medios para el tren de medios.

En respuesta a la instrucción DeleteConnection, la pasarela retorna una lista de parámetros que describen el status de la conexión<sup>16</sup>. Estos parámetros son:

- **Número de paquetes enviados:** Número total de paquetes de datos RTP transmitidos por el emisor a través de la conexión desde el comienzo de la transmisión. La cuenta no se reinicia si el emisor cambia su identificador de fuente de sincronización (SSRC, definido en RTP), por ejemplo como resultado de una instrucción Modify. El valor es cero si, por ejemplo, la conexión siempre estuvo fijada en el modo "sólo recepción".
- **Número de octetos enviados:** Número total de octetos de cabida útil (es decir, sin contar los octetos de encabezamiento ni los de relleno) transmitidos en paquetes de datos RTP por el emisor desde el inicio de la transmisión en la conexión. La cuenta no se reinicia si el emisor cambia su identificador SSRC, por ejemplo como resultado de una instrucción ModifyConnection. El valor es cero si, por ejemplo, la conexión siempre estuvo fijada en el modo "sólo recepción".
- **Número de paquetes recibidos:** Número total de paquetes de datos RTP recibidos por el emisor desde el comienzo de la recepción en la conexión. La cuenta incluye los paquetes recibidos de una SSRC diferente si el emisor utilizó varios valores. El valor es cero si, por ejemplo, la conexión siempre estuvo fijada en el modo "sólo envío".
- **Número de octetos recibidos:** Número total de octetos de cabida útil (es decir, sin contar los octetos de encabezamiento ni los de relleno) transmitidos en paquetes de datos RTP por el emisor desde el comienzo de la transmisión en la conexión. La cuenta incluye los paquetes recibidos de una SSRC diferente si el emisor utilizó varios valores. El valor es cero si, por ejemplo, la conexión siempre estuvo fijada en el modo "sólo envío".
- **Número de paquetes perdidos:** Número total de paquetes de datos RTP que se han perdidos desde el comienzo de la recepción. Este número está definido por el número de paquetes que se esperaba recibir menos el número de paquetes efectivamente recibidos, donde el número de paquetes recibidos incluye los paquetes recibidos tardíamente y los que están duplicados. La cuenta incluye los paquetes recibidos de una SSRC diferente si el emisor utilizó varios valores. Por tanto, los paquetes recibidos tardíamente no se cuentan como perdidos, y la cuenta de paquetes perdidos puede ser negativa si se reciben paquetes duplicados. La cuenta incluye paquetes recibidos de una SSRC diferente si el emisor utilizó varios valores. El número de paquetes que se espera recibir es, por definición, el último

---

<sup>16</sup> Los valores calculados no incluirán paquetes creados como resultado de la vigilancia electrónica.

número secuencial extendido recibido, menos el número secuencial inicial recibido. La cuenta incluye paquetes recibidos de una SSRC diferente, si el emisor utilizó varios valores. El valor es cero si, por ejemplo, la conexión siempre estuvo fijada en el modo "sólo envío".

- **Fluctuación entre-llegadas:** Estimación de la varianza estadística del tiempo entre llegadas de paquetes de datos RTP medido en milisegundos y expresado como un entero sin signo. La fluctuación entre-llegadas "J" se define como la desviación media (valor absoluto alisado) de la diferencia "D" en el espaciamiento de los paquetes en el receptor, en comparación con el emisor, para un par de paquetes. En IETF RFC 1889 se presentan algoritmos de cálculo detallados. La cuenta incluye paquetes recibidos de una SSRC diferente si el emisor utilizó varios valores. El valor es cero si, por ejemplo, la conexión siempre estuvo fijada en el modo "sólo envío".
- **Retardo promedio de transmisión:** Estimación de la latencia de la red, expresada en milisegundos. Es el valor promedio de la diferencia entre la indicación de tiempo NTP de los emisores de los mensajes RTCP y la indicación de tiempo NTP de los receptores, medida cuando se recibieron los mensajes. El promedio se obtiene dividiendo la suma de todas las estimaciones por el número de mensajes RTCP recibidos. Debe señalarse que el cálculo correcto de este parámetro presupone que los relojes están sincronizados. Los dispositivos de pasarela de circuitos troncales PUEDEN, como una alternativa, calcular el retardo promedio de transmisión dividiendo por dos el tiempo medido de ida y retorno.

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

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

Cuando estos parámetros están presentes, la instrucción de supresión de conexión y la petición de notificación DEBEN estar sincronizadas, lo que significa que o bien ambas deberán ser aceptadas, o bien ambas deberán ser rechazadas.

**ReturnCode** es un parámetro retornado por la pasarela. Indica el resultado de la instrucción y está constituido por un número entero (véase A.2.5) seguido facultativamente de un comentario.

#### A.2.3.6 **DeleteConnection (desde la pasarela de circuitos troncales)**

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

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

El **EndpointId**, en esta forma de la instrucción **DeleteConnection**, DEBE estar totalmente calificado. Los convenios sobre uso de comodín NO SE UTILIZARÁN.

El **Reason-code** es una cadena de texto que comienza por un código de motivo numérico y va seguido facultativamente de una cadena de texto descriptiva. En A.2.6 figura una lista de códigos de motivo.



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

**ReturnCode** es un parámetro retornado por el MGC. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase A.2.5).

#### A.2.3.7 DeleteConnection (múltiples conexiones desde el controlador de pasarela de medios)

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

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

El **EndpointId**, en esta forma de la instrucción DeleteConnection, NO UTILIZARÁ el comodín "any of". Todas las conexiones para el punto extremo o los puntos extremos con el CallId especificado serán suprimidas. La instrucción no retorna estadísticas ni parámetros individuales de la llamada.

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

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

Con esta forma de la instrucción DeleteConnection, los MGC pueden servirse de la estructura de denominación jerárquica de los puntos extremos para suprimir todas las conexiones pertenecientes a un grupo de puntos extremos. En este caso, parte del componente "local endpoint name" del EndpointId puede especificarse utilizando el convenio de uso de comodín "all" especificado en A.2.1.1. NO SE UTILIZARÁ el convenio de uso de comodín "any of". La instrucción no retorna estadísticas ni parámetros de llamada individuales.

Una vez suprimida la conexión, los trenes de medios de la red de paquetes antes soportados por la conexión dejan de estar disponibles. Todo paquete de medios recibido para la antigua conexión será, simplemente, descartado y no se enviará ningún nuevo paquete de medios.

**ReturnCode** es un parámetro retornado por la pasarela. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase A.2.5).

#### A.2.3.8 Auditorización

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

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

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

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

Una gestión de red que vaya más allá de las capacidades proporcionadas por estas instrucciones, por ejemplo, información sobre el status de la pasarela de circuitos troncales por oposición al de puntos extremos individuales, es deseable, en general. Cabe esperar que esas capacidades sean soportadas mediante el empleo del protocolo de gestión de red simple (SNMP, *simple network management protocol*) y por la definición de una MIB para la pasarela de circuitos troncales; estos dos aspectos están fuera del ámbito de esta Recomendación.

#### A.2.3.8.1 AuditEndPoint

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

```

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

```

El **EndPointId** identifica el punto extremo que se audita. NO SE UTILIZARÁ el convenio de uso de comodín "any of".

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

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

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

La respuesta, a su vez, contendrá información sobre cada uno de los puntos con relación a los cuales se solicitó información de auditorización:

- **RequestedEvents:** Valor actual de RequestedEvents que el punto extremo está utilizando, incluida la acción asociada con cada evento. Los eventos persistentes se incluyen en la lista.

- **SignalRequests:** Lista de las señales sometidas a temporización que se encuentran activas en ese momento, señales On/Off que están activadas ("on") en ese momento para el punto extremo (con o sin parámetro), y cualquier señal breve pendiente<sup>17</sup>. No se incluirán las señales sometidas a temporización para las cuales haya expirado el periodo de temporización, ni las señales breves que estén produciendo efecto en ese momento. Las señales parametrizadas se informan junto con los parámetros con los que se aplican.
- **RequestIdentifier:** RequestIdentifier para la última NotificationRequest recibida por el punto extremo (incluye la petición de notificación insertada en primitivas de tratamiento de la conexión). Si no se ha recibido ninguna petición de notificación, se retornará el valor cero.
- **NotifiedEntity:** Actual "entidad notificada" para el punto extremo.
- **ConnectionIdentifiers:** Lista de ConnectionIdentifiers, separados por un carácter coma (",") para todas las conexiones que existen en ese momento para el punto extremo especificado.
- **DetectEvents:** valor de DetectEvents que el punto extremo está utilizando. Los eventos persistentes se incluyen en la lista.
- **ObservedEvents:** Lista actual de eventos observados para el punto extremo.
- **EventStates:** Para eventos que tienen asociados estados auditorizables, el evento que corresponde al estado en que se encuentra el punto extremo, por ejemplo, toma si el circuito troncal MF para el punto extremo está tomado en ese momento. La definición de cada evento individual determinará si el evento en cuestión tiene asociado un estado auditorizable.
- **VersionSupported:** Lista de las versiones de protocolo soportadas por el punto extremo.
- **Capabilities:** Capacidades del punto extremo, similares a las del parámetro LocalConnectionOptions e incluye lotes de eventos y modos de conexión. Si es necesario especificar que algunos parámetros, como por ejemplo, supresión de silencio, sólo son compatibles con algunos códecs, la pasarela retornará varios conjuntos de capacidades. Si se interroga a un punto extremo sobre una capacidad que no entiende, el punto extremo NO GENERARÁ un error, sino que DEBE omitir el parámetro en su respuesta:
  - **Algoritmo de compresión** Lista de códecs soportados. Los parámetros restantes se aplicarán a todos los códecs especificados en esta lista.
  - **Periodo de paquetización** Se puede especificar un valor individual o una gama de valores.
  - **Anchura de banda** Se puede especificar un valor individual o una gama que corresponda a la gama del periodo de paquetización (suponiendo que no haya supresión de silencio).
  - **Compensación de eco** Indica si la compensación de eco está o no soportada<sup>18</sup>.
  - **Supresión de silencio** Indica si la supresión de silencio está o no soportada.
  - **Tipo de servicio** Indica si el tipo de servicio está o no soportado.
  - **Lotes de eventos** Lista de los lotes de eventos soportados. El primer lote de eventos en la lista será el lote por defecto.
  - **Modos** Lista de los modos de conexión soportados.
  - **Seguridad** Indica si los servicios de seguridad IPCablecom están o no soportados. Si están soportados, los siguientes parámetros también pueden estar presentes:

<sup>17</sup> En ese momento no debería haber señales breves pendientes.

<sup>18</sup> Actualmente, todos los puntos extremos TGCP deben soportar la compensación de eco.

- **Series de texto cifrado RTP:** Lista de algoritmos de autenticación y criptación.
- **Series de texto cifrado RTCP:** Lista de algoritmos de autenticación y criptación soportados por RTCP.
- **Vigilancia electrónica:** Indica si la vigilancia electrónica IPCablecom está o no soportada.

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

**ReturnCode** es un parámetro retornado por la pasarela. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase A.2.5).

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

Debe señalarse que todas las informaciones retornadas son del tipo "fotografía instantánea". Las nuevas instrucciones recibidas, actividades locales, etc. pueden modificar lo anteriormente expuesto. Por ejemplo, el estado de toma puede cambiar antes de que el MGC reciba la información antes mencionada.

#### A.2.3.8.2 AuditConnection

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

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

El **EndpointId** identifica el punto extremo objeto de la auditorización; **NO SE UTILIZARÁN** comodines. **RequestedInfo** (que puede estar vacía) describe la información que se solicita para el **ConnectionId** dentro del `EndpointId` especificado. Con esta instrucción puede auditorizarse la siguiente información de conexión:

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

La respuesta, a su vez, incluirá información sobre cada uno de los puntos sobre los cuales se solicitó información de auditorización:

- **CallId** El CallId para la llamada a que pertenece la conexión
- **NotifiedEntity** La actual "entidad notificada" para el punto extremo.
- **LocalConnectionOptions** Las LocalConnectionOptions suministradas para la conexión.
- **Mode** El actual modo de conexión.
- **ConnectionParameters** Los parámetros de conexión actuales para la conexión.
- **LocalConnectionDescriptor** El LocalConnectionDescriptor suministrado por la pasarela para la conexión.

- **RemoteConnectionDescriptor** El RemoteConnectionDescriptor suministrado por la pasarela para la conexión.

**ReturnCode** es un parámetro retornado por la pasarela. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase A.2.5).

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

### A.2.3.9 Rearranque en curso

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

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

El **EndpointId** identifica los puntos extremos que son retirados del servicio o restablecidos al servicio. Puede utilizarse el convenio de uso de comodín "all of" para aplicar la instrucción a un grupo de puntos extremos, por ejemplo a todos los puntos extremos que están asociados a una interfaz especificada, o incluso a todos los puntos extremos que están asociados a una pasarela dada. NO SE UTILIZARÁ el convenio de uso de comodín "any of".

El parámetro RestartMethod especifica el tipo de rearmenque:

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

El parámetro "retardo de rearmenque" es facultativo y se expresa como un número de segundos. Si se omite el número de segundos, debe considerarse que el valor del retardo es nulo. En el caso del método "paulatino", un retardo nulo indica que el MGC debería simplemente esperar la terminación natural de las conexiones existentes, y no establecer nuevas conexiones. El retardo de rearmenque se considera siempre nulo en el caso del método "forzado" y del método "cancelar-paulatino". Un retardo de rearmenque nulo en el caso del método "rearmenque" indica que el servicio ya ha sido restablecido. Esto suele ocurrir cuando la pasarela ha rearmenqueado/reinicializado. Para mitigar los efectos de un cambio de dirección IP, el MGC PUEDE desear resolver el nombre de dominio de la

pasarela interrogando al servidor de nombres de dominio (DNS) cualquiera que sea el tiempo de vida (TTL) de un registro de recursos actual para la pasarela rearmada.

Las pasarelas de circuitos troncales DEBERÍAN enviar un mensaje RestartInProgress "paulatino" o "forzado" al MGC cuando son retirados del servicio, por ejemplo, cuando son cerrados o retirados del servicio por un sistema de gestión de red, aunque el MGC no puede confiar en que siempre habrá de recibir tales mensajes. Las pasarelas de circuitos troncales DEBEN enviar un mensaje RestartInProgress "rearranque" con un retardo nulo a su MGC cuando retornan al servicio de acuerdo con el procedimiento de rearranque especificado en A.2.4.3.5; los MGC pueden confiar en que recibirán este mensaje. Asimismo, las pasarelas de circuitos troncales DEBEN enviar un mensaje RestartInProgress "desconectado" a su "entidad notificada" actual de acuerdo con el procedimiento "desconectado" especificado en A.2.4.3.6. El parámetro "retardo de rearranque" NO SE UTILIZARÁ con el método de rearranque "forzado".

El mensaje RestartInProgress se enviará a la actual "entidad notificada" para el EndpointId en cuestión. Cabe esperar que para cada punto extremo se haya proporcionado un MGC por defecto, es decir, una "entidad notificada", por lo que tras una reinicialización, el MGC por defecto será la "entidad notificada" para cada punto extremo. Las pasarelas de circuitos troncales DEBEN servirse plenamente del uso de comodines para minimizar el número de mensajes RestartInProgress generados cuando rearrancan múltiples puntos extremos en una pasarela y los puntos extremos son gestionados por el mismo MGC.

**ReturnCode** es un parámetro retornado por el MGC. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase A.2.5).

Con la respuesta del MGC se puede retornar adicionalmente una **NotifiedEntity**:

- Si la respuesta indicaba éxito (código de retorno 200 – transacción ejecutada), el procedimiento de rearranque ha sido completado y la NotifiedEntity retornada es la nueva "entidad notificada" para el punto extremo o los puntos extremos.
- Si la respuesta del MGC indicaba un error, el procedimiento de rearranque no ha sido completado todavía, y hay que iniciarlo de nuevo. Si se retornó un parámetro NotifiedEntity, éste especifica entonces la nueva "entidad notificada" para el punto o los puntos extremos, que, por consiguiente, habrá que utilizar cuando se vuelva a intentar el procedimiento de rearranque.

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

#### **A.2.4 Estados, condiciones de cambio-por-fallo y pugna por llegar primero (carrera)**

Para implementar una señalización adecuada de las llamadas, el MGC debe seguir el rastro del estado del punto extremo, y la pasarela debe asegurarse de que los eventos son correctamente notificados al MGC. Pueden existir condiciones especiales cuando la pasarela o el MGC son rearmados: la pasarela puede tener que ser redireccionada hacia un nuevo MGC en el curso de procedimientos de "cambio-por-fallo". De manera similar, el MGC puede tener que ejecutar acciones especiales cuando se pone fuera de línea o se rearmada la pasarela.

##### **A.2.4.1 Aspectos importantes**

Como se expresó en A.2.1.4, los MGC se identifican por su nombre de dominio y, en cualquier instante dado, cada punto extremo tiene asociada una, y sólo una "entidad notificada". En esta cláusula se destacan los aspectos que son de especial importancia para la fiabilidad y el cambio-por-fallo en MGCP:

- Un MGC se identifica por su nombre de dominio, no por sus direcciones de red, y varias direcciones pueden estar asociadas con un nombre de dominio.

- En cualquier instante dado, un punto extremo tiene asociado un MGC, y sólo uno. El MGC asociado con un punto extremo es el valor actual de la "entidad notificada".
- La "entidad notificada" se fija inicialmente a un valor suministrado. Cuando para un punto extremo se reciben instrucciones con un parámetro NotifiedEntity que incluye nombres de punto extremo que contienen comodines, la "entidad notificada" se fija al valor especificado. Si la "entidad notificada" para un punto extremo está vacía o ha sido fijada explícitamente<sup>19</sup>, la "entidad notificada" adopta por defecto la dirección fuente de la última instrucción de tratamiento de la conexión o petición de notificación recibida para el punto extremo. En este caso, el MGC será identificado de esa forma por su dirección de red, lo que sólo DEBERÍA hacerse excepcionalmente.
- Las respuestas a instrucciones siempre se envían a la dirección fuente de la instrucción, cualquiera que sea la "entidad notificada" actual. Cuando es necesario remolcar un mensaje Notify con la respuesta, el datagrama se envía aún a la dirección fuente de la nueva instrucción recibida, sin tener en cuenta la NotifiedEntity para cualquiera de las instrucciones.
- Cuando la "entidad notificada" hace referencia a un nombre de dominio que se resuelve en múltiples direcciones IP, los puntos extremos pueden conmutar de cada de cada una de estas direcciones a cada una de las demás, pero no pueden, por sí mismos, cambiar la "entidad notificada" por otro nombre de dominio. Sin embargo, un MGC puede ordenarles que conmuten a una nueva dirección proporcionándoles una nueva "entidad notificada".
- Si un MGC se torna indisponible, los puntos extremos gestionados por ese MGC resultarán finalmente "desconectados". Estos puntos extremos sólo pueden volver a conectarse si se da cualquiera de estos dos casos: que el MGC que falló se torne de nuevo disponible, o que otro MGC (un MGC de respaldo) establezca contacto con los puntos extremos afectados y les proporcione una nueva "entidad notificada".
- Cuando otro MGC (un MGC de respaldo) ha asumido el control de un grupo de puntos extremos, se supone que el MGC que ha fallado se comunicará y sincronizará con el MGC de respaldo para que éste restituya a aquél el control de los puntos extremos afectados, si así se desea. Otra posibilidad es que el MGC falló desempeñe ahora el papel de MGC de respaldo.

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

#### **A.2.4.2 Retransmisión, y detección de asociaciones perdidas**

El protocolo MGCP está organizado como un conjunto de transacciones cada una de las cuales se compone de una instrucción y una respuesta. Los mensajes MGCP, que se transportan mediante UDP, pueden sufrir pérdidas. En ausencia de una respuesta tempestiva (véase A.3.5), las instrucciones se repiten. Las pasarelas DEBEN mantener en memoria una lista de las respuestas que envían a transacciones recientes, y una lista de las transacciones que están en ejecución. El hecho de ser reciente se define en este contexto por el valor  $T_{hist}$ , que especifica el número de segundos durante los cuales hay que conservar las respuestas a las transacciones antiguas. El valor por defecto de  $T_{hist}$  es 30 segundos.

Los identificadores de transacción de las instrucciones entrantes se comparan primeramente con los identificadores de transacción de las respuestas recientes. Si se encuentra una concordancia, la pasarela no ejecuta la transacción y se limita a repetir la respuesta antigua. Si no se encuentra una concordancia con una transacción a la que se haya respondido antes, el identificador de transacción

---

<sup>19</sup> Esto podría suceder, por ejemplo, cuando se especifica un parámetro NotifiedEntity vacío.

de la instrucción entrante se compara con la lista de transacciones cuya ejecución todavía no ha terminado. Si se encuentra tal concordancia, la pasarela no ejecuta la transacción, que, simplemente, no se tiene en cuenta; cuando termina la ejecución de la instrucción, se proporciona una respuesta.

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

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

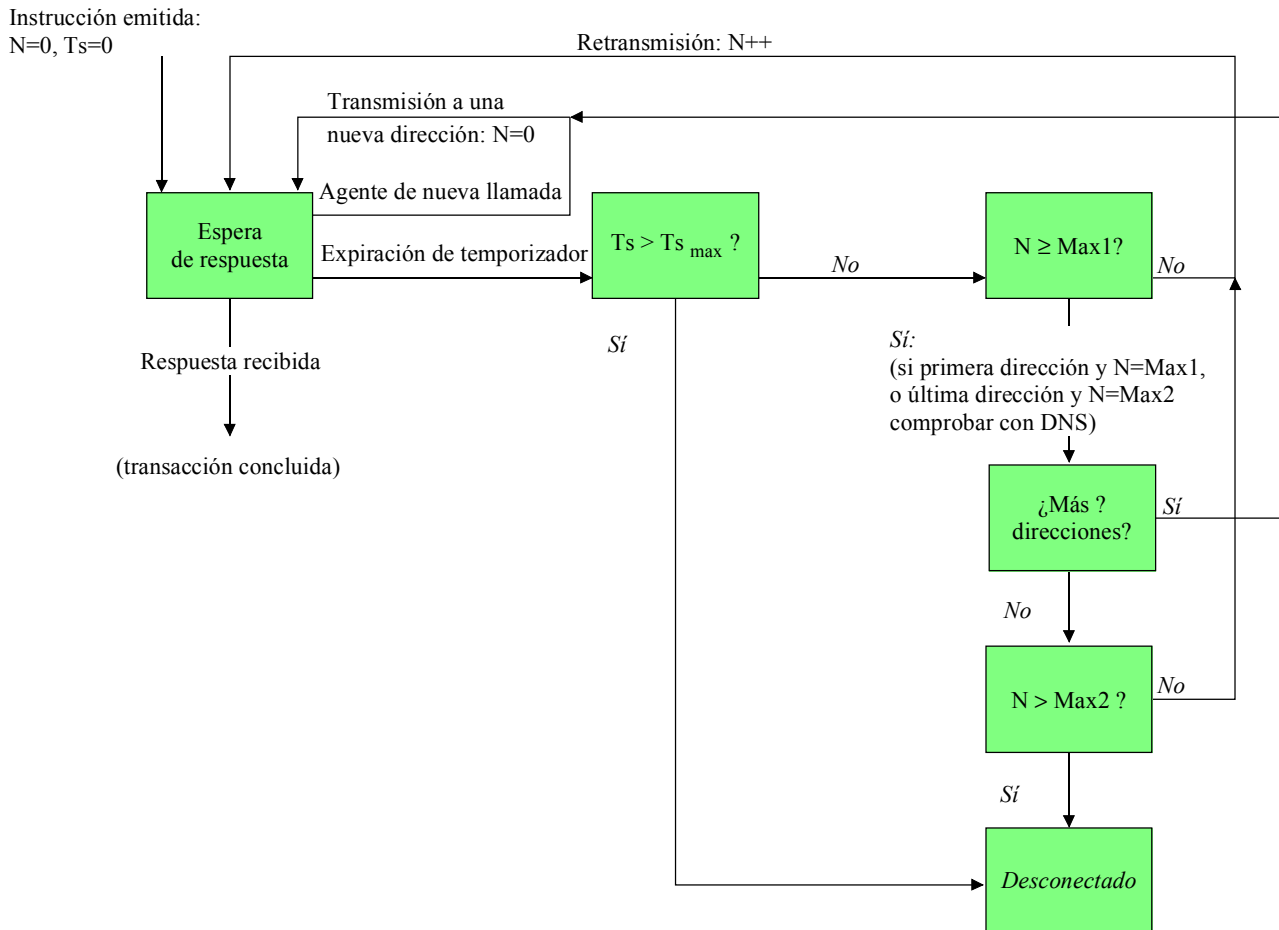
Los elementos deben poder derivar, de los datos históricos, una estimación de la tasa de pérdida de paquetes. En un sistema debidamente configurado, esta tasa debería ser muy baja, típicamente inferior a 1% en promedio. Si un MGC o una pasarela tienen que repetir un mensaje varias veces, cabe suponer que se esté en presencia de algo más que un error de transmisión. Por ejemplo, dada una tasa de pérdidas uniformemente distribuida de 1%, la probabilidad de que fracasen cinco intentos de transmisión consecutivos es de 1 en  $10^{-11}$ , suceso que sólo debería producirse una vez cada diez días en el caso de un MGC que procesa 1000 transacciones por segundo. (En realidad, el número de repeticiones que se considere excesivo debe estar en función de la tasa efectiva de pérdida de paquetes.) Cuando los errores no están distribuidos uniformemente, la probabilidad de fallos consecutivos puede ser algo más alta. Se debe señalar que el "umbral de sospecha", que se designará por "Max1", normalmente es más bajo que el "umbral de desconexión", que se designará por "Max2", y que debería fijarse a un valor mayor.

Un algoritmo de retransmisión clásico (véase la figura A.1) contaría simplemente el número de repeticiones sucesivas, y llegaría a la conclusión de que la asociación se rompe después de retransmitir el paquete un número excesivo de veces (típicamente, entre 7 y 11 veces). Para tener en cuenta la posibilidad de un "cambio-por-fallo" no detectado o en curso, el algoritmo clásico se ha modificado como sigue:

- La pasarela siempre DEBE efectuar una comprobación para determinar la presencia de un nuevo MGC. Esto se puede saber por:
  - la recepción de una instrucción en la que la NotifiedEntity apunta a un nuevo MGC, o
  - la recepción de una respuesta de redireccionamiento que apunta a un nuevo MGC.
- Si se detecta un nuevo MGC, la pasarela DEBE redireccionar hacia ese nuevo MGC las retransmisiones de todas las instrucciones pendientes dirigidas inicialmente al punto o puntos extremos. Las respuestas a instrucciones nuevas o antiguas seguirán transmitiéndose a la dirección fuente de la instrucción.
- Antes de efectuar una retransmisión, se verifica que el tiempo transcurrido desde el envío del datagrama inicial no es mayor que  $T_{s_{max}}$ . Si el tiempo transcurrido es mayor que  $T_{s_{max}}$ , el punto extremo es desconectado.
- Si el número de retransmisiones a este MGC es igual a "Max1", la pasarela PUEDE interrogar al servidor de nombres para detectar un posible cambio de interfaces de MGC, cualquiera que sea el tiempo de vida (TTL, *time to live*) asociado al registro del servidor DNS.
- La pasarela puede haber aprendido varias direcciones IP para el MGC. Si el número de retransmisiones para una de estas direcciones IP es mayor que "Max1" y menor que "Max2", y si hay más direcciones IP con las cuales no se han intentado retransmisiones, la pasarela DEBE dirigir las retransmisiones a las direcciones alternativas restantes en su lista local.



- Si no queda más ninguna interfaz por ensayar, y el número de retransmisiones es Max2, la pasarela DEBERÍA establecer contacto con el DNS una vez más para averiguar si una o más de esas otras interfaces se han tornado disponibles. En caso negativo, el punto o puntos extremos gestionados por este MGC son entonces desconectados. Cuando un punto extremo es desconectado, DEBE iniciar el procedimiento "desconectado", como se especifica en A.2.4.3.6.



T0912530-02

**Figura A.1/J.171 – Algoritmo de retransmisión**

Para adaptarse automáticamente a la carga de la red, MGCP especifica temporizadores que aumentan exponencialmente (véase A.3.5.2). Si el periodo inicial de temporización se fija a 200 milisegundos, la pérdida de una quinta retransmisión se detectará después de unos seis segundos. Probablemente, este es un tiempo de espera aceptable para la detección de un cambio-por-fallo. Las retransmisiones deberían continuar después de ese plazo no sólo para resolver tal vez un problema transitorio de conectividad, sino también para dar un poco más de tiempo a la ejecución de un cambio-por-fallo; un periodo de espera (en total) de 30 segundos probablemente sea aceptable.

Debe señalarse que existe una estrecha relación entre  $T_{s_{max}}$ ,  $T_{t_{hist}}$ , y el máximo tiempo de tránsito,  $T_{p_{max}}$ . Específicamente, para evitar que las instrucciones retransmitidas se ejecuten más de una vez se DEBE satisfacer la siguiente relación:

$$T_{t_{hist}} \geq T_{s_{max}} + T_{p_{max}}$$

El valor por defecto de  $T_{s_{max}}$  es 20 segundos. Por tanto, si el máximo tiempo de propagación supuesto es diez segundos, las respuestas a las transacciones antiguas deben conservarse durante un periodo de al menos 30 segundos. Es muy importante lograr que el emisor y el receptor convengan en estos valores.

El valor por defecto para Max1 es cinco retransmisiones y el valor por defecto para Max2 es siete retransmisiones. Estos dos valores pueden ser modificados por el proceso de provisión.

Además, el proceso de provisión DEBE poder inhabilitar la interrogación de DNS sobre Max1, sobre Max2, o sobre ambos valores.

#### **A.2.4.3 Condiciones de pugna por llegar primero (condiciones de carrera)**

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

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

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

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

##### **A.2.4.3.1 Lista de cuarentena**

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

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

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

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

El punto extremo sale del "estado notificación" cuando se recibe la respuesta a la instrucción Notify<sup>20</sup>. La instrucción Notify puede retransmitirse en el "estado notificación", como se especifica en A.2.4.2.

Cuando el punto extremo sale del "estado notificación" reinicia la lista de eventos observados del punto extremo al valor nulo.

El perfil de TGCP prescribe la utilización del "modo lockstep", lo que implica que la pasarela DEBE recibir una nueva instrucción NotificationRequest después que ha enviado una instrucción Notify. Hasta que esto suceda, el punto extremo está en el "estado lockstep", y los eventos que sucedan y deban ser detectados se almacenan simplemente en la memoria intermedia de cuarentena. Los eventos que habrán de ponerse en cuarentena son los mismos que están en el "estado notificación". Una vez recibida y ejecutada correctamente la nueva NotificationRequest, el punto extremo sale del "estado lockstep".

Una pasarela puede recibir en cualquier momento una nueva instrucción NotificationRequest para el punto extremo, la cual producirá también el efecto de hacer que el punto extremo salga del "estado notificación", suponiendo que la NotificationRequest se ejecute correctamente.

Cuando se recibe una nueva NotificationRequest en el "estado notificación", la pasarela garantizará que el MGC recibirá la instrucción Notify pendiente antes de que se haya emitido con éxito una respuesta a la nueva NotificationRequest. Para ello utiliza la funcionalidad "remolque" del protocolo y crea los mensajes (instrucciones y respuestas) que habrán de enviarse por orden cronológico, empezando por el más antiguo. Los mensajes se enviarán entonces en un solo paquete a la fuente de la nueva NotificationRequest, cualesquiera que sean la fuente y la "entidad notificada" para las instrucciones antigua y nueva. Se siguen los siguientes pasos:

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

Tras la recepción de una instrucción NotificationRequest, la lista "eventos solicitados" se reemplaza por los nuevos parámetros recibidos, y la lista de "eventos observados" se reinicializa al valor nulo. El comportamiento subsiguiente está entonces condicionado por el valor del parámetro QuarantineHandling. Este parámetro puede especificar que los eventos puestos en cuarentena deban descartarse, en cuyo caso se descartarán todos los eventos que estén en cuarentena. Si el parámetro especifica que los eventos puestos en cuarentena deben procesarse, la pasarela comenzará a procesar la lista de eventos puestos en cuarentena, utilizando la nueva lista de "eventos solicitados". Cuando procesa estos eventos, la pasarela puede encontrar un evento que provoca el envío de una instrucción Notify. En tal caso, la pasarela transmite inmediatamente una instrucción Notify que informa todos

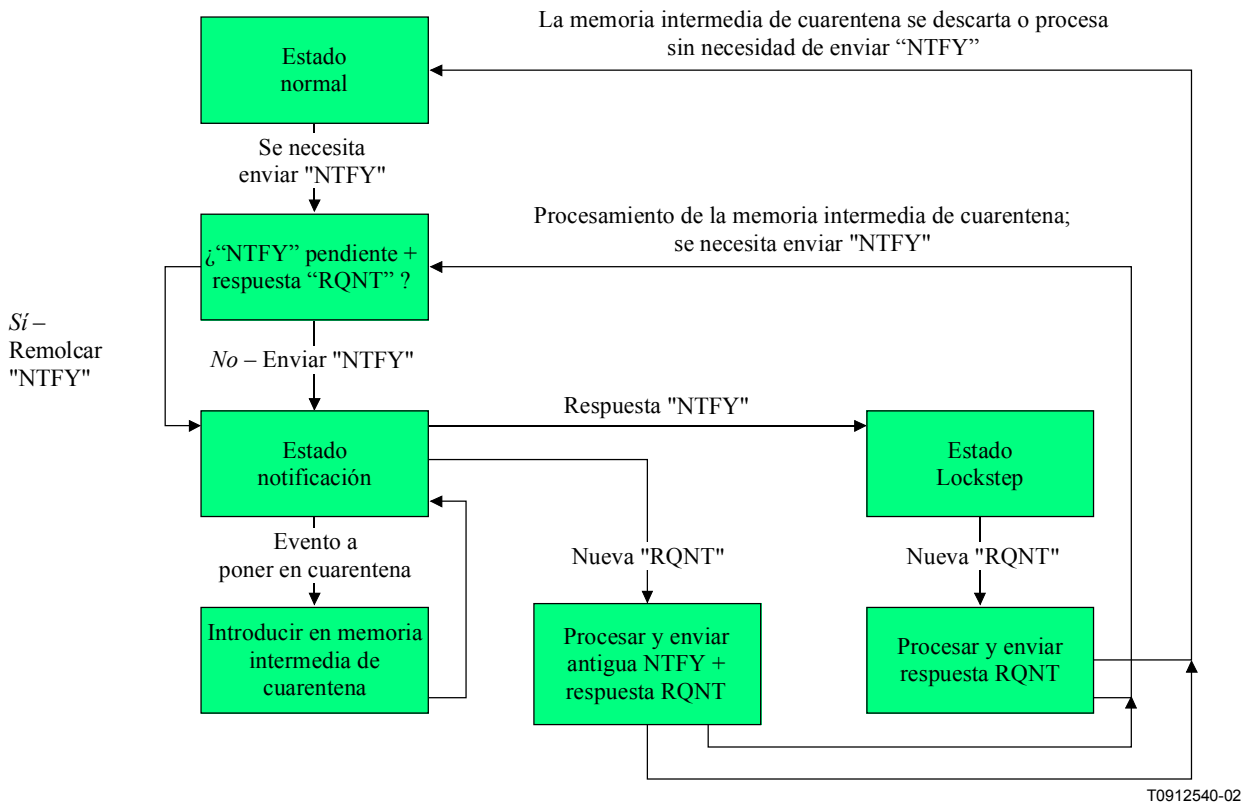
---

<sup>20</sup> Debe señalarse que la acción Notify no puede combinarse con una NotificationRequest insertada.

los eventos que estaban acumulados en la lista de "eventos observados" hasta el evento de activación, inclusive, y deja en la memoria intermedia de cuarentena los eventos que no han sido procesados. Después de esto, el punto extremo retorna al "estado notificación".

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

La figura A.2 ilustra dicho procedimiento, partiendo del supuesto de que todas las transacciones se ejecutan con éxito:



**Figura A.2/J.171 – Algoritmo de cuarentena**

Los MGC DEBERÍAN proporcionar la respuesta a un mensaje Notify exitoso y a la nueva NotificationRequest en el mismo datagrama utilizando el mecanismo de remolque<sup>21</sup>.

#### A.2.4.3.2 Detección explícita

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

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

este aspecto es, sin embargo, de sumo interés para los circuitos troncales CAS, que no están soportados en esta versión de la Recomendación).

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

- 1) Si se pide a la pasarela que notifique una transición a "toma"<sup>22</sup> en un momento en que el circuito está ya tomado (código de error 401 – circuito tomado).
- 2) Si se pide a la pasarela que notifique una condición de no-toma<sup>23</sup> en un momento en que el circuito no está tomado (código de error 402 – circuito no tomado).

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

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

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

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

Puede producirse otra condición de carrera cuando se emite una instrucción Notify poco tiempo después de que la pasarela haya recibido una NotificationRequest. El RequestIdentifier se utiliza para correlacionar instrucciones Notify con instrucciones NotificationRequest, lo que permite al MGC determinar si la instrucción Notify fue generada antes o después de que la pasarela recibió la nueva NotificationRequest.

#### **A.2.4.3.3 Semántica transaccional**

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

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

Para reflejar esta situación, la semántica más simple consiste en formular que todas las precondiciones DEBEN ser satisfechas desde el instante en que se inicia hasta el instante en que concluye la transacción. Por tanto, si cualquiera de las precondiciones cambia durante la ejecución

---

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

<sup>23</sup> Por ejemplo, solicitando el evento “rel” en un circuito troncal de servicios de operador MF sin que haya una llamada en curso.

de la transacción, ésta DEBE fracasar. Además, a partir del instante mismo en que se inicia la transacción, todos los nuevos eventos que se produzcan se pondrán en cuarentena. Una vez que se sepa el resultado de la transacción, se procesan los eventos que están en cuarentena.

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

#### **A.2.4.3.4 Ordenación de las instrucciones, y su tratamiento cuando no están ordenadas**

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

- Las instrucciones Notify pueden ser demoradas y llegar al MGC después de la transmisión de una nueva instrucción de petición de notificación.
- Si se transmite una nueva instrucción NotificationRequest antes de recibirse una respuesta a una anterior, no se garantiza que la primera instrucción no habrá de recibirse en segunda posición.

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

- 1) Cuando una pasarela trata varios puntos extremos, unas instrucciones pertenecientes a diferentes puntos extremos pueden enviarse en paralelo, por ejemplo siguiendo un modelo en el que cada punto extremo es controlado por su propio proceso o su propio hilo.
- 2) Cuando se crean varias conexiones en un mismo punto extremo, las instrucciones pertenecientes a diferentes conexiones pueden enviarse en paralelo.
- 3) En una conexión dada, normalmente debe haber una sola instrucción pendiente (instrucción de crear o modificar). Sin embargo, en cualquier momento se puede emitir una instrucción DeleteConnection. Por consiguiente, una pasarela puede a veces recibir una instrucción ModifyConnection que es aplicable a una conexión anterior que ha sido suprimida. Se DEBE ignorar tal instrucción y retornar un error (código de error 515 – identificador de conexión incorrecto).
- 4) En un punto extremo dado, normalmente debe haber una sola instrucción NotificationRequest pendiente en un momento dado cualquiera. El parámetro RequestId se utiliza para correlacionar instrucciones Notify con la NotificationRequest activadora.
- 5) En algunos casos, una instrucción DeleteConnection, con uso implícito o explícito de comodín, aplicable a un grupo de puntos extremos, puede aparecer antes que una instrucción CreateConnection pendiente. El MGC deberá suprimir cada una de las conexiones cuya compleción esté pendiente al tiempo de la instrucción DeleteConnection global. Asimismo, no deberían enviarse nuevas instrucciones CreateConnection para puntos extremos denominados mediante el uso de comodines, hasta que se haya recibido una respuesta a la instrucción DeleteConnection en la que se emplearon comodines.
- 6) Cuando se insertan instrucciones unas dentro de otras, se DEBEN cumplir los requisitos de secuenciación para todas las instrucciones. Por ejemplo una instrucción CreateConnection dentro de la cual hay una petición de notificación debe cumplir, al mismo tiempo, los requisitos de secuenciación para CreateConnection y NotificationRequest.
- 7) AuditEndpoint y AuditConnection no están sometidas a secuenciación.

- 8) RestartInProgress debe ser la primera instrucción enviada por un punto extremo, de acuerdo con lo definido para el procedimiento de rearme (véase A.2.4.3.5). Toda otra instrucción o respuesta deberá entregarse después de esta instrucción RestartInProgress (está permitido el remolque).
- 9) Cuando múltiples mensajes son remolcados en un solo paquete, los mensajes siempre son procesados en su orden.

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

#### **A.2.4.3.5 Prevención de los rearmes simultáneos de numerosas pasarelas**

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

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

El procedimiento de rearme expresa simplemente que el punto extremo DEBE enviar una instrucción RestartInProgress al MGC por la que le informe sobre el rearme y, además, garantizar que el primer mensaje (de instrucción o respuesta) que el MGC perciba como procedente de este punto extremo sea esta instrucción RestartInProgress. Para conseguir esto, el punto extremo DEBE aprovechar la posibilidad que se le ofrece de transportar en remolque. Por ejemplo, si se produce una actividad de toma de circuito antes de la expiración del temporizador de rearme, se generará un paquete que contendrá la instrucción RestartInProgress, con una instrucción Notify transportada en remolque para el evento de toma. En caso de que el temporizador de rearme expire sin que se haya producido ninguna otra actividad, la pasarela envía simplemente un mensaje RestartInProgress.

Si la pasarela pasa al estado "desconectado" mientras está efectuando el procedimiento de rearme, se DEBE ejecutar el procedimiento desconectado especificado en A.2.4.3.6, con la salvedad de que en el curso de este procedimiento se envía un mensaje "rearme" en lugar de un mensaje "desconectado".

Se espera que a cada punto extremo en una pasarela será posible asignarle un MGC, es decir, una "entidad notificada", al cual se deberá dirigir el mensaje inicial de rearme. Cuando el conjunto de puntos extremos en una pasarela es gestionado por más de un MGC, el procedimiento antes descrito se efectuará para cada conjunto de puntos extremos gestionados por un MGC dado. La pasarela debe aprovechar la posibilidad de utilizar comodines para minimizar el número de mensajes

RestartInProgress generados cuando múltiples puntos extremos en una pasarela rearrancan y los puntos extremos son gestionados por el mismo MGC.

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

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

#### **A.2.4.3.6 Puntos extremos desconectados**

Además del procedimiento de re arranque, las pasarelas de circuitos troncales también disponen de un procedimiento "desconectado", que se inicia cuando un punto extremo es "desconectado", como se describe en A.2.4.2. Debe señalarse a este respecto que los puntos extremos sólo pueden ser desconectados cuando tratan de comunicar con el MGC. Un punto extremo que es "desconectado" da los pasos siguientes:

- 1) Un temporizador "desconectado" se inicializa a un valor aleatorio, uniformemente distribuido entre 0 y un periodo de espera inicial "desconectado" ( $T_{d_{init}}$ ), por ejemplo, de 15 segundos. En la generación de los números aleatorios, se DEBE tomar la precaución de evitar toda tendencia a una sincronización no deseada entre las múltiples pasarelas que vayan a utilizar el mismo algoritmo.
- 2) La pasarela espera entonces: sea la expiración del temporizador, sea la recepción de una instrucción del MGC, sea la detección de una actividad de circuito local para el punto extremo, como por ejemplo una transición a toma.
- 3) Cuando expira el temporizador "desconectado", o se recibe una instrucción, o se detecta una actividad de circuito local, la pasarela inicia el procedimiento "desconectado" para el punto extremo. En el caso de actividad de circuito local, es necesario, además, que haya transcurrido un mínimo periodo de espera "desconectado" ( $T_{d_{min}}$ ), suministrable, desde que la pasarela devino desconectada o desde la última vez que inició el procedimiento "desconectado" para limitar la velocidad a la que se aplica el procedimiento.
- 4) Si, pese a haberse aplicado el procedimiento "desconectado" al punto extremo, éste continúa desconectado, se duplica el valor del temporizador "desconectado", aunque respetando un máximo periodo de espera "desconectado" ( $T_{d_{max}}$ ), suministrable, por ejemplo, de 600 segundos, y la pasarela retorna al paso 2.

El procedimiento "desconectado" es similar al procedimiento de re arranque en cuanto a que ahora expresa simplemente que el punto extremo DEBE enviar una instrucción RestartInProgress al MGC por la que de informa que el punto extremo fue desconectado y, además, garantiza que el primer mensaje (de instrucción o de respuesta) que el MGC ahora percibe como procedente del punto extremo DEBE ser esta instrucción RestartInProgress. Para conseguir esto, el punto extremo DEBE aprovechar la posibilidad que se le ofrece de transportar en remolque. El MGC puede entonces, por ejemplo, optar por auditar el punto extremo o, sencillamente, liberar todas las conexiones para el punto extremo.



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

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

### A.2.5 Códigos de retorno y códigos de error

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

- el valor 000 indica un acuse de recibo de respuesta<sup>24</sup>,
- los valores entre 100 y 199 indican una respuesta provisional,
- los valores entre 200 y 299 indican una compleción exitosa,
- los valores entre 400 y 499 indican un error transitorio,
- los valores entre 500 y 599 indican un error permanente.

Los valores que han sido definidos se indican en el cuadro A.2:

**Cuadro A.2/J.171 – Códigos de retorno**

<b>Código</b>	<b>Significado</b>
000	Acuse de recibo de respuesta.
100	La transacción se está ejecutando en este momento. Seguirá ulteriormente un mensaje de compleción efectiva.
200	La transacción solicitada se ejecutó normalmente.
250	Las conexiones fueron suprimidas.
400	La transacción no pudo ejecutarse debido a un error transitorio.
401	El teléfono ya está descolgado o el circuito ya está tomado.
402	El teléfono ya está colgado o el circuito no está tomado.
500	La transacción no pudo ejecutarse porque el punto extremo es desconocido.
501	La transacción no pudo ejecutarse porque el punto extremo no está listo.
502	La transacción no pudo ejecutarse porque el punto extremo no tiene recursos suficientes.
510	La transacción no pudo ejecutarse porque se detectó un error de protocolo.
511	La transacción no pudo ejecutarse porque la instrucción contenía una extensión no reconocida.
512	La transacción no pudo ejecutarse porque la pasarela no está equipada para detectar uno de los eventos solicitados.
513	La transacción no pudo ejecutarse porque la pasarela no está equipada para generar una de las señales solicitadas.
514	La transacción no pudo ejecutarse porque la pasarela no puede enviar el anuncio especificado.
515	La transacción hace referencia a un identificador de conexión incorrecto (puede haber sido ya suprimido).
516	La transacción hace referencia a un identificador de llamada desconocido.
517	Modo no soportado o no válido.

<sup>24</sup> El acuse de recibo de respuesta se utiliza para respuestas provisionales (véase A.3.8).

<b>Código</b>	<b>Significado</b>
518	Lote no soportado o desconocido.
519	El punto extremo no tiene un mapa de dígitos.
520	La transacción no pudo ejecutarse porque el punto extremo está "rearrancando".
521	Punto extremo redireccionado a otro MGC.
522	No hay tal evento o señal.
523	Acción desconocida o combinación ilegal de acciones.
524	Inconsistencia interna en LocalConnectionOptions.
525	Extensión desconocida en LocalConnectionOptions.
526	Anchura de banda insuficiente.
527	Falta RemoteConnectionDescriptor.
528	Versión de protocolo incompatible.
529	Fallo de soporte físico interno.
532	Valor o valores no soportados en LocalConnectionOptions.
533	Respuesta demasiado grande.

### **A.2.6 Códigos de motivo**

Cuando la pasarela suprime una conexión, utiliza códigos de motivo para informar al MGC sobre el motivo por el que se suprime la conexión. El código de motivo es un número entero; se han definido los siguientes valores en el cuadro A.3:

**Cuadro A.3/J.171 – Códigos de motivo**

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

### **A.3 Protocolo de control de pasarela de medios**

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

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

Las instrucciones de los cuatro primeros tipos las envía el MGC a una pasarela. La instrucción Notify la envía la pasarela al MGC. La pasarela puede también enviar una instrucción DeleteConnection como se indica en A.2.3.6. El MGC puede enviar cualquiera de las dos instrucciones de auditorización a la pasarela y, por último, la pasarela puede enviar una instrucción RestartInProgress al MGC.

### **A.3.1 Descripción general**

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

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

Los encabezamientos y las descripciones de sesión se codifican como un conjunto de líneas de texto, separadas por un carácter de retorno del carro y un carácter de nueva línea (o, facultativamente, un solo carácter de nueva línea). El encabezamiento está separado de la descripción de sesión por línea vacía.

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

### **A.3.2 Encabezamiento de instrucción**

El encabezamiento de instrucción se componen de:

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

A menos que se indique otra cosa, o que en otras normas a que se hace referencia se tomen disposiciones diferentes, los componentes del encabezamiento de instrucción son insensibles a la escritura en mayúscula o minúscula. Esto es aplicable a los verbos, así como a los parámetros y valores, y en todas las comparaciones se DEBE considerar que las letras mayúsculas son equivalentes a las letras minúsculas, tanto cuando se utilizan individualmente, como combinadas unas con otras.

#### **A.3.2.1 Línea de instrucción**

La línea de instrucción se compone de lo siguiente:

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

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

### A.3.2.1.1 Codificación del verbo solicitado

Los verbos solicitados se codifican como códigos ASCII de cuatro letras, escritas en mayúscula y/o minúscula (las comparaciones deben ser insensibles a la escritura en mayúscula o minúscula), como se define en el siguiente cuadro A.4:

**Cuadro A.4/J.171 – Códigos del verbo solicitado**

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

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

Una pasarela que reciba una instrucción con un verbo experimental que ella no soporte DEBE retornar un código de error 511 – extensión no reconocida).

### A.3.2.1.2 Identificadores de transacción

Se utilizan identificadores de transacción para correlacionar instrucciones y respuestas.

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

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

Como mínimo, los identificadores de transacción para instrucciones enviadas a una determinada pasarela de circuitos troncales DEBEN ser únicos para el máximo tiempo de vida de las transacciones dentro de la colección de MGC que controlan esa pasarela de circuitos troncales (véase A.3.5). Por tanto, cualquiera que sea el MGC emisor, las pasarelas de circuitos troncales siempre pueden detectar transacciones duplicadas, simplemente examinando el identificador de transacción. Sin embargo, la coordinación de estos identificadores de transacción entre los MGC está fuera del ámbito de esta Recomendación.

Los identificadores de transacción para todas las instrucciones enviadas desde una determinada pasarela de circuitos troncales DEBEN ser únicos durante el máximo tiempo de vida de las transacciones (véase A.3.5) cualquiera que sea el MGC a que se envía la instrucción. Por tanto, un MGC siempre puede detectar una transacción duplicada procedente de una pasarela de circuitos troncales por la combinación del nombre de dominio del punto extremo y el identificador de transacción. La pasarela a su vez siempre puede detectar un acuse de recibo de respuesta duplicado examinando el identificador o los identificadores de transacción.

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

Los identificadores de transacción tienen valores comprendidos entre 1 y 999999999. Una entidad MGCP NO REUTILIZARÁ un identificador de transacción dentro de los tres minutos que siguen a la compleción de la instrucción precedente en la que se utilizó el identificador.

### **A.3.2.1.3 Codificación de los nombre de punto extremo, controlador de pasarela de medios y entidad notificada**

Los nombres de punto extremo y los nombres de MGC se codifican como direcciones de correo electrónico, según se define en IETF RFC 821. En estas direcciones, el nombre de dominio identifica el sistema en el que el punto extremo está vinculado, mientras que el lado izquierdo identifica un punto extremo concreto de ese sistema. Ambos componentes DEBEN ser insensibles a la escritura en mayúscula o minúscula.

Son ejemplos de tales nombres:

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

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

`MGC@mgc.whatever.net:5234`

Si se omite el número de puerto, se utiliza el puerto MGCP por defecto (2427). En A.2.1.1, se da una información más detallada sobre los nombres de punto extremo.

### **A.3.2.1.4 Codificación de la versión de protocolo**

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

La versión de protocolo para esta Recomendación DEBE codificarse como:

`MGCP 1.0 TGCP 1.0`

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

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

### **A.3.2.2 Líneas de parámetro**

Las líneas de parámetro se componen de un nombre de parámetro, que en la mayoría de los casos está formado por un solo carácter letra mayúscula, seguido de un carácter dos puntos (":"), un espacio blanco, y el valor de parámetro. En cambio, los nombres y los valores de parámetro son insensibles a la escritura en mayúscula o minúscula. Los parámetros que pueden estar presentes en instrucciones se definen en el cuadro A.5:

**Cuadro A.5/J.171 – Parámetros de instrucción**

<b>Nombre de parámetro</b>	<b>Código</b>	<b>Valor de parámetro</b>
ResponseAck <sup>25</sup>	K	Véase la descripción.
CallId	C	Cadena hexadecimal, 32 caracteres como máximo.
ConnectionId	I	Cadena hexadecimal, 32 caracteres como máximo.
NotifiedEntity	N	Un identificador, en formato IETF RFC 821, compuesto de una cadena cualquiera y del nombre de dominio de la entidad solicitante, posiblemente completado por un número de puerto, como en: Call-agent@ca.whatever.net:5234
RequestIdentifier	X	Véase la descripción.
LocalConnectionOptions	L	Véase la descripción.
ConnectionMode	M	Véase la descripción.
RequestedEvents	R	Véase la descripción.
SignalRequests	S	Véase la descripción.
ObservedEvents	O	Véase la descripción.
ConnectionParameters	P	Véase la descripción.
ReasonCode	E	Véase la descripción.
SpecificEndPointId	Z	Un identificador, en formato IETF RFC 821, compuesto de una cadena cualquiera, seguido facultativamente de un carácter "@" seguido del nombre de dominio de la pasarela de circuitos troncales a que este punto extremo está vinculado.
MaxEndPointIds	ZM	Cadena decimal, 16 caracteres como máximo.
NumEndpoints	ZN	Cadena decimal, 16 caracteres como máximo.
RequestedInfo	F	Véase la descripción.
QuarantineHandling	Q	Véase la descripción.
DetectEvents	T	Véase la descripción.
EventStates	ES	Véase la descripción.
RestartMethod	RM	Véase la descripción.
RestartDelay	RD	Un número de segundos codificado como un número decimal.
Capabilities	A	Véase la descripción.
VersionSupported	VS	Véase la descripción.

<sup>25</sup> El parámetro ResponseAck no se mostró en A.2.3 porque los identificadores de transacción no son visibles en el ejemplo de API. Los implementadores pueden elegir un método diferente.

Los parámetros no están necesariamente presentes en todas las instrucciones. El cuadro A.6 presenta la asociación entre parámetros e instrucciones. La letra M (*mandatory*) significa obligatorio, O (*optional*) significa facultativo, y F (*forbidden*) significa prohibido:

**Cuadro A.6/J.171 – Asociación de parámetros con instrucciones solicitadas**

Nombre de parámetro	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
ResponseAck <sup>25</sup>	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
ConnectionMode	M	O	F	F	F	F	F	F
RequestedEvents	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>	F	F	F	F
SignalRequests	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>	F	F	F	F
NotifiedEntity	O	O	O	O	O	F	F	F
ReasonCode	F	F	O	F	F	F	F	F
ObservedEvents	F	F	F	F	M	F	F	F
Connection parameters	F	F	O	F	F	F	F	F
SpecificEndpointId	F	F	F	F	F	O	F	F
MaxEndPointIds	F	F	F	F	F	O	F	F
NumEndPoints	F	F	F	F	F	F	F	F
RequestedInfo	F	F	F	F	F	O	O	F
QuarantineHandling	O	O	O	O	F	F	F	F
DetectEvents	O	O	O	O	F	F	F	F
EventStates	F	F	F	F	F	F	F	F
RestartMethod	F	F	F	F	F	F	F	M
RestartDelay	F	F	F	F	F	F	F	O
Capabilities	F	F	F	F	F	F	F	F
VersionSupported	F	F	F	F	F	F	F	F
RemoteConnectionDescriptor	O	O	F	F	F	F	F	F
<sup>a)</sup> Los parámetros RequestedEvents y SignalRequests son facultativos en la NotificationRequest. Si se omiten estos parámetros, se considerará que las listas correspondientes están vacías. En el caso de instrucciones de tratamiento de conexión, esto es también aplicable cuando se incluye un RequestIdentifier.								

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

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

X-FlowerOfTheDay: Daisy

Los nombres de parámetro que comienzan por "X+" son extensiones de parámetros obligatorios. Una pasarela que recibe una extensión de parámetro obligatorio que no entienda DEBE responder con un error (código de error 511 - extensión no reconocida).

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

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

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

#### **A.3.2.2.1 Acuse de recibo de respuesta**

El parámetro acuse de recibo de respuesta se utiliza para el soporte de la triple toma de contacto descrita en A.3.7. Contiene una lista de "gamas de identificadores de transacción confirmados" separadas por comas.

NOTA – El parámetro ResponseAck no se presentó en A.2.3 porque los identificadores de transacción no son visibles en el ejemplo de API aquí presentado. Los implementadores pueden optar por un método diferente.

Cada "gama de identificadores de transacción confirmados" se compone de un número decimal, cuando la gama comprende una sola transacción, o de dos números decimales, separados por un solo guión, que indican, respectivamente, el identificador de transacción más bajo y el más alto de los comprendidos en la gama.

Un ejemplo de acuse de recibo de respuesta es:

K: 6234-6255, 6257, 19030-19044

#### **A.3.2.2.2 Identificador de petición**

El identificador de petición correlaciona una instrucción Notify con la NotificationRequest que la ocasionó. Un RequestIdentifier es una cadena hexadecimal con una longitud máxima de 32 caracteres. La cadena "0" está reservada para el informe de eventos persistentes cuando todavía no se ha recibido ninguna NotificationRequest (véase A.2.3.2).

#### **A.3.2.2.3 Opciones de conexión local**

Las opciones de conexión local describen los parámetros operacionales que la pasarela debe utilizar para una conexión cuando así se lo ordena el MGC. Estos parámetros son:

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



Los parámetros LocalConnectionOptions utilizados para seguridad se codifican como sigue:

- El secreto se codifica como la palabra clave "sc-st" seguida de un carácter dos puntos (":"), un método, un carácter dos puntos, y el secreto en cuestión. El método es o bien la cadena "clear" si el secreto está codificado en texto claro, o bien la cadena "base64" si el secreto está codificado utilizando base64.
- La serie de texto cifrado RTP se codifica como la palabra clave "sc-rtcp" seguida de un carácter dos puntos (":"), una cadena de serie de texto cifrado RTP, como se define más adelante. Se puede especificar una lista de valores, en cuyo caso los valores irán separados por un carácter punto y coma (";").
- La serie de texto cifrado RTCP se codifica como la palabra clave "sc-rtcp" seguida de un carácter dos puntos (":"), una cadena de serie de texto cifrado RTCP, como se define más adelante. Puede especificarse una lista de valores, en cuyo caso los valores irán separados por un carácter punto y coma (";").

Las cadenas de texto cifrado RTP y RTCP observan la gramática:

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

donde ALPHA y DIGIT se definen en IETF RFC 2234. Dentro de una serie de texto cifrado no se permiten espacios blancos. El siguiente ejemplo ilustra la utilización de serie de texto cifrado:

62/51

La lista de las series de texto cifrado efectivamente soportadas por IPCablecom se presenta en la Rec. UIT-T J.170.

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

Son ejemplos de opciones de conexión local:

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

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

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

Los parámetros de LocalConnectionOptions utilizados para vigilancia electrónica son:

- El identificador de conexión de contenido de llamada codificado como la palabra clave "es-cci" seguida de un carácter dos puntos (":"), y una cadena de hasta ocho caracteres hexadecimales que corresponde a un identificador con una longitud de 32 bits para el identificador de conexión de contenido de llamada.

- El destino de contenido de llamada codificado como la palabra clave "es-ccd" seguida de un carácter dos puntos (":") y una dirección IP codificada en forma similar a la de una dirección IP para la porción nombre de dominio de un nombre de punto extremo. La dirección IP va seguida de un carácter dos puntos (":") y hasta cinco caracteres decimales para un número de puerto UDP que habrá de utilizarse.

#### A.3.2.2.4 Capabilities

El parámetro Capabilities (capacidades) informa al MGC sobre sus capacidades cuando es auditorizado. La codificación de Capabilities se basa en la codificación de las opciones de conexión local en el caso de parámetros que son comunes a ambas codificaciones. Capabilities puede contener, además, una lista de lotes soportados y una lista de modos soportados.

Se utilizan los siguientes parámetros:

- El periodo de paquetización en milisegundos, codificado como la palabra clave "p" seguida de un carácter dos puntos (":") y un número decimal. Una gama puede especificarse como dos números decimales separados por un guión.
- El nombre literal del algoritmo de compresión, codificado como la palabra clave "a" seguida de un carácter dos puntos (":") y una cadena de caracteres. Puede especificarse una lista de valores, en cuyo caso los valores irán separados por un carácter punto y coma (";").
- La anchura de banda en kilobits/segundo (1 000 bits/segundo), codificada como la palabra clave "b" seguida de un carácter dos puntos (":") y un número decimal. Una gama puede especificarse como dos números decimales separados por un guión.
- El parámetro compensación de eco, codificado como la palabra clave "e" seguida de un carácter dos puntos (":") y el valor "on" si la compensación de eco está soportada, o el valor "off" si no está soportada.
- El parámetro tipo de servicio, codificado como la palabra clave "t" seguida de un carácter dos puntos (":") y el valor "0" si el parámetro tipo de servicio no está soportado; todos los demás valores indican que este parámetro está soportado.
- El parámetro supresión de silencio, codificado como la palabra clave "s" seguida de un carácter dos puntos (":") y el valor "on" si la supresión de silencio está soportada, u "off" si no está soportada.
- Los lotes de eventos soportados por este punto extremo, codificados como la palabra clave "v" seguida de un carácter dos puntos (":") y una lista de los nombres de lotes soportados separados por un carácter punto y coma (";"). El primer valor especificado será el lote por defecto para el punto extremo.
- Los modos de conexión soportados por este punto extremo, codificados como la palabra clave "m" seguida de un carácter dos puntos (":") y una lista de modos de conexión definidos en A.3.2.2.7, separados por un carácter punto y coma (";").
- La palabra clave "sc-st" si está soportada la seguridad de IPCablecom. En este caso, las siguientes palabras clave indican las series de texto cifrado soportadas:
  - La palabra clave "sc-rtp" seguida de un carácter dos puntos (":") y una lista de AuthenticationAlgorithms RTP separados por un carácter punto y coma (";"), una barra oblicua, y una lista de EncryptionAlgorithms soportados separados por un carácter punto y coma (";").
  - La palabra clave "sc-rtcp" seguida de un carácter dos puntos (":") y una lista de AuthenticationAlgorithms RTCP separados por un carácter punto y coma (";"), una barra oblicua, y una lista de AuthenticationAlgorithms soportados separados por un carácter punto y coma (";").

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

Son ejemplos de capacidades:

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

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

- La palabra clave "es-cci" si la vigilancia electrónica de IPCablecom está soportada.

### A.3.2.2.5 Parámetros de conexión

Los parámetros de conexión se codifican como una cadena de pares de tipo y valor, donde el tipo es un identificador del parámetro, formado por dos letras, y el valor es un número entero decimal. Los tipos se separan de los valores por un signo "=". Los parámetros se separan unos de otros por una coma.

Los tipos de parámetros de conexión se especifican en el cuadro A.7:

**Cuadro A.7/J.171 – Tipos de parámetros de conexión**

Nombre de parámetro de conexión	Código	Valor de parámetro de conexión
Paquetes enviados	PS	Número de paquetes que fueron enviados por la conexión.
Octetos enviados	OS	Número de octetos que fueron enviados por la conexión.
Paquetes recibidos	PR	Número de paquetes que fueron recibidos por la conexión.
Octetos recibidos	OR	Número de octetos que fueron recibidos por la conexión.
Paquetes perdidos	PL	Número de paquetes que no fueron recibidos por la conexión; se deduce de los saltos en el número secuencial.
Fluctuación	JI	Fluctuación promedio de los tiempos de llegada entre los paquetes, en milisegundos, expresada como un número entero.
Latencia	LA	Latencia promedio, en milisegundos, expresada como un número entero.

Los nombres de los parámetros de conexión con extensión se componen de la cadena "X-" seguida de un nombre de parámetro con extensión formado por dos letras. Los MGC que reciban extensiones no reconocidas DEBEN ignorar estas extensiones sin ejecutar ninguna acción.

Un ejemplo de codificación de un parámetro de conexión es:

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

### A.3.2.2.6 Códigos de motivo

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

```
900 Endpoint malfunctioning
```

En A.2.6 se presenta una lista de códigos de motivo.

### A.3.2.2.7 Modo de conexión

El modo de conexión describe el modo de funcionamiento de la conexión. Los posibles valores están indicados en el cuadro A.8:

**Cuadro A.8/J.171 – Valores del modo de conexión**

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

### A.3.2.2.8 Codificación de nombre de evento/señal

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

Son ejemplos de nombres de evento válidos:

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

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

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

Por último, el asterisco puede utilizarse para designar "todas las conexiones", y el signo de dólar puede utilizarse para designar la conexión "actual". Los siguientes ejemplos de estas notaciones no son válidos:

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

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

### A.3.2.2.9 RequestedEvents

El parámetro RequestedEvents proporciona la lista de eventos que se ha solicitado. Los códigos de evento actualmente definidos se describen en el anexo A.A. Cada evento puede ser calificado por una acción solicitada, o por una lista de acciones. No todas las acciones pueden combinarse con cada una de las demás; para las combinaciones válidas, véase A.2.3.1. Las acciones, cuando se especifican, se codifican como una lista de palabras clave encerradas entre paréntesis y separadas por comas. Los códigos para las diversas acciones se indican en el cuadro A.9:

**Cuadro A.9/J.171 – Códigos de conexión**

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

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

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

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

La acción NotificationRequest insertada tiene el siguiente formato:

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

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

La acción ModifyConnection insertada tiene el siguiente formato:

C(M(<ConnectionMode<sub>1</sub>>( <ConnectionID<sub>1</sub>> )) , ... ,  
M(<ConnectionMode<sub>n</sub>>( <ConnectionID<sub>n</sub>> )) )

El siguiente ejemplo ilustra la utilización de ModifyConnection insertada:

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

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

### A.3.2.2.10 SignalRequests

El parámetro SignalRequests proporciona el nombre de las señales que han sido solicitadas. Las señales actualmente definidas se indican en el anexo A.A. Una determinada señal sólo puede aparecer una vez en la lista y, por definición, todas las señales se aplican al mismo tiempo.

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

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

donde signal-parameter-value puede ser una cadena, o una cadena entre comillas, esto es, una cadena encerrada entre dos caracteres comillas dobles. Para designar el carácter comillas dobles dentro de una cadena encerrada entre comillas dobles se utilizan dos caracteres comillas dobles consecutivos. Por ejemplo, la cadena entre comillas dobles "ab" "c" producirá la cadena ab"c.

Cada señal tiene asociado uno de los siguientes tipos de señal (véase A.2.3.1):

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

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

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

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

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

Señales individuales pueden definir parámetros de señal adicionales.

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

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

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

```
S: signal1, signal2
```

#### **A.3.2.2.11 ObservedEvents**

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

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

#### **A.3.2.2.12 RequestedInfo**

El parámetro RequestedInfo contiene una lista de códigos de parámetro separados por una coma, como se define en la sección "Líneas de parámetro"; en A.2.3.8 se indican los parámetros que pueden ser autorizados. Los siguientes valores indicados en el cuadro A.10 también son soportados:

**Cuadro A.10/J.171 – Valores de RequestedInfo soportados**

Parámetro RequestedInfo	Código
LocalConnectionDescriptor	LC
RemoteConnectionDescriptor	RC

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

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

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

F: A

#### **A.3.2.2.13 QuarantineHandling**

El parámetro tratamiento de cuarentena contiene la palabra clave "process" o "discard" para indicar el tratamiento de los eventos puestos en cuarentena, por ejemplo:

Q: process

#### **A.3.2.2.14 DetectEvents**

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

T: ft, mt

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

#### **A.3.2.2.15 EventStates**

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

ES: MO/rlc

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

#### **A.3.2.2.16 RestartMethod**

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

RM: restart

#### **A.3.2.2.17 VersionSupported**

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

VS: MGCP 1.0, MGCP 1.0 TGCP 1.0

### **A.3.3 Formatos de encabezamiento de respuesta**

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

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

200 1201 OK

En el cuadro A.11 se recapitulan los parámetros de respuesta cuya presencia es obligatoria o facultativa en un encabezamiento de respuesta, en función de la instrucción que ocasionó la respuesta, suponiendo que la instrucción tuvo éxito. Se invita al lector a examinar distintas definiciones de instrucciones, ya que en este cuadro sólo se ofrece una breve información. La letra M (*mandatory*) significa obligatorio, O (*optional*) significa facultativo, y F (*forbidden*) significa prohibido.

**Cuadro A.11/J.171 – Asociación de parámetros con respuesta de instrucción**

Nombre de parámetro	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
ResponseAck <sup>26</sup>	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>	O <sup>a)</sup>
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
ConnectionMode	F	F	F	F	F	F	O	F
RequestedEvents	F	F	F	F	F	O	F	F
SignalRequests	F	F	F	F	F	O	F	F
NotifiedEntity	F	F	F	F	F	O	O	O
ReasonCode	F	F	F	F	F	F	F	F
ObservedEvents	F	F	F	F	F	O	F	F
ConnectionParameters	F	F	O	F	F	F	O	F
Specific Endpoint ID	O	F	F	F	F	O	F	F
MaxEndPointIds	F	F	F	F	F	F	F	F
NumEndPoints	F	F	F	F	F	O	F	F
RequestedInfo	F	F	F	F	F	F	F	F
QuarantineHandling	F	F	F	F	F	F	F	F
DetectEvents	F	F	F	F	F	O	F	F
EventStates	F	F	F	F	F	O	F	F
RestartMethod	F	F	F	F	F	F	F	F
RestartDelay	F	F	F	F	F	F	F	F
Capabilities	F	F	F	F	F	O	F	F
VersionSupported	F	F	F	F	F	O	F	O
LocalConnection Descriptor	M	O	F	F	F	F	O	F
RemoteConnection Descriptor	F	F	F	F	F	F	O	F

<sup>a)</sup> El parámetro ResponseAck NO SE UTILIZARÁ con otra respuesta que no sea la respuesta final emitida después de una respuesta provisional para la transacción en cuestión. En ese caso, la presencia del parámetro ResponseAck DEBE ocasionar un mensaje de acuse de recibo de respuesta; todo valor de ResponseAck proporcionado será ignorado.

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

<sup>26</sup> El parámetro ResponseAck no se mostró en A.2.3 porque los identificadores de transacción no son visibles en el ejemplo de API. Los implementadores pueden elegir un método diferente.



### A.3.3.1 CreateConnection

En el caso de un mensaje CreateConnection, la línea de respuesta va seguida de un parámetro identificador de conexión con una respuesta exitosa (código 200). Se transmite además un LocalConnectionDescriptor con una respuesta positiva. El LocalConnectionDescriptor se codifica como una "descripción de sesión", como se define en A.3.4. 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
```

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

```
200 1204 OK
K:
I: FDE234C8

v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 96
a=rtpmap:96 G726-32/8000
```

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

```
000 1204
```

### A.3.3.2 ModifyConnection

En el caso de un mensaje ModifyConnection exitoso, la línea de respuesta va seguida de un LocalConnectionDescriptor, si la modificación culminó en una modificación de los parámetros de la sesión (por ejemplo, el solo cambio del modo de conexión no entraña una modificación de los parámetros de la sesión). El LocalConnectionDescriptor se codifica como una "descripción de sesión", como se define en A.3.4. 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
```

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

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

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

```
000 1207 OK
```

### A.3.3.3 DeleteConnection

En dependencia de la variante del mensaje DeleteConnection, la línea de respuesta puede ir seguida de una línea de parámetro de ConnectionParameters, como se define en A.3.2.2.5.

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

### A.3.3.4 NotificationRequest

Una respuesta NotificationRequest no incluye parámetros de respuesta adicionales.

### A.3.3.5 Notify

Una respuesta Notify no incluye parámetros de respuesta adicionales.

### A.3.3.6 AuditEndpoint

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

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

O:

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

### A.3.3.7 AuditConnection

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

```
200 1203 OK
C: A3C47F21456789F0
N: [128.96.41.12]
L: 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 proporciona un descriptor de conexión local y un descriptor de conexión distante, el descriptor de conexión local será el primero de los dos. Si se solicita un descriptor de conexión pero no existe ninguno para la conexión autorizada, dicho descriptor de conexión aparecerá con el campo versión de protocolo SDP solamente.

### A.3.3.8 RestartInProgress

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

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

### A.3.4 Codificación de descripción de sesión

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

La utilización de SDP depende del tipo de sesión, como se especifica en el parámetro "medios". Actualmente, la Recomendación sobre TGCP sólo soporta medios de tipo "audio".

#### A.3.4.1 Utilización del servicio de audio SDP

En una pasarela especializada en conexiones vocales, sólo es necesario describir sesiones que utilizan exactamente un medio, audio. Los parámetros SDP que son de interés para las aplicaciones de audio de tipo vocal se especifican más adelante. La pasarela de circuitos troncales DEBE soportar descripciones de sesión que cumplan estas reglas, y en el orden siguiente:

- 1) el perfil SDP presentado más adelante;
- 2) IETF RFC 2327 (*SDP: Session Description Protocol*).

El perfil SDP proporcionado describe la utilización del protocolo de descripción de sesión en TGCP. La descripción general y la explicación de los distintos parámetros pueden encontrarse en IETF RFC 2327; no obstante, lo cual se indican detalladamente a continuación los valores que los puntos extremos TGCP deben proporcionar para estos campos (send (enviar)) y lo que los puntos extremos TGCP deberían hacer con valores suministrados o no suministrados para estos campos (receive (recibir)).

Un punto extremo TGCP NO DEBERÍA proporcionar un parámetro no que no fuera uno de los especificados más adelante, y si recibiera tal parámetro DEBERÍA ignorarlo.

##### A.3.4.1.1 Protocol Version (v=) (versión de protocolo)

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

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

**Receive:** DEBE proporcionarse de acuerdo con IETF RFC 2327.

##### A.3.4.1.2 Origin (o=) (origen)

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

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

Username (nombre de usuario)

**Send:** El carácter guión DEBE utilizarse como nombre de usuario cuando se pide confidencialidad. El carácter guión DEBERÍA utilizarse en todo otro caso.<sup>27</sup>

**Receive:** Este campo DEBERÍA ignorarse.

Session-ID (identificador de sesión)

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

**Receive:** Este campo DEBERÍA ignorarse.

Version (versión)

**Send:** De conformidad con IETF RFC 2327.

**Receive:** Este campo DEBERÍA ignorarse.

Network Type (tipo de red)

**Send:** DEBE utilizarse el tipo 'IN'.

**Receive:** Este campo DEBERÍA ignorarse.

Address Type (tipo de dirección)

**Send:** DEBE utilizarse el tipo "IP4"

**Receive:** Este campo DEBERÍA ignorarse.

Address (dirección):

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

**Receive:** Este campo DEBE ignorarse.

#### **A.3.4.1.3 Session name (s=) (nombre de sesión)**

s=<session-name>

s=-

**Send:** El carácter guión DEBE utilizarse como nombre de sesión.

**Receive:** Este campo DEBE ignorarse.

#### **A.3.4.1.4 Session and Media Information (i=) (información de sesión y medios)**

i=<session-description>

**Send:** Para TGCP, el campo NO SE UTILIZARÁ.

**Receive:** Este campo DEBE ignorarse.

#### **A.3.4.1.5 URI (u=) (identificador de recurso universal)**

u=<URI>

**Send:** Para TGCP, el campo NO SE UTILIZARÁ.

**Receive:** Este campo DEBE ignorarse.

---

<sup>27</sup> Como los puntos extremos TGCP no saben cuándo se ha pedido confidencialidad, siempre se DEBERÍA utilizar un guión.

#### A.3.4.1.6 E-Mail Address and Phone Number (e=, p=) (dirección de correo electrónico y número de teléfono)

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

**Send:** Para TGCP, el campo NO SE UTILIZARÁ.

**Receive:** Este campo DEBE ignorarse.

#### A.3.4.1.7 Connection Data (c=) (datos de conexión)

El campo datos de conexión consta de 3 subcampos:

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

Network Type (tipo de red)

**Send:** DEBE utilizarse el tipo 'IN'.

**Receive:** El tipo "IN" DEBE estar presente.

Address Type (tipo de dirección)

**Send:** DEBE utilizarse el tipo "IP4".

**Receive:** El tipo "IP4" DEBE estar presente.

Connection Address (dirección de conexión)

**Send:** Este campo DEBE llenarse con una dirección IP unicast (unidifusión) en la que la aplicación recibirá el tren de medios. Por tanto NO ESTARÁ PRESENTE un valor TTL y NO ESTARÁ PRESENTE un valor "number of addresses" (número de direcciones). El campo NO SERÁ LLENADO con un nombre de dominio plenamente calificado en lugar de una dirección IP. Una dirección diferente de cero especifica la dirección de envío y la dirección de recepción para el tren o trenes de medios que dicha dirección abarca.

**Receive:** Una dirección IP unicast (unidifusión) o un nombre de dominio plenamente calificado DEBEN estar presentes. Una dirección diferente de cero especifica la dirección de envío y la dirección de recepción para el tren o trenes de medios que dicha dirección abarca.

#### A.3.4.1.8 Bandwidth (b=) (anchura de banda)

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

**Send:** La información de anchura de banda es facultativa en SDP pero siempre DEBERÍA incluirse<sup>28</sup>. Cuando se utiliza un rtpmap o un códec que no sea muy conocido<sup>29</sup>, DEBE utilizarse la información de anchura de banda.

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

Modifier (modificador)

**Send:** DEBE utilizarse el tipo 'AS'.

**Receive:** El tipo "AS" DEBE estar presente.

<sup>28</sup> Si no se utiliza este campo, el controlador de puerta podría no autorizar la anchura de banda adecuada.

<sup>29</sup> Por códec no muy conocido ha de entenderse un códec que no esté definido en la Rec. UIT-T J.161 sobre códecs IPCablecom.

Bandwidth Value (valor de anchura de banda)

**Send:** El campo DEBE llenarse con el valor que corresponda a la máxima anchura de banda requerida por el tren de medios, en kilobits/segundo.

**Receive:** La máxima anchura de banda requerida por el tren de medios, en kilobits/segundo DEBE estar presente.

#### A.3.4.1.9 Time, repeat times and time zones (t=, r=, z=) (tiempo, tiempos de repetición y zonas de tiempo (o zonas horarias))

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

**Send:** Time DEBE estar presente; start time PUEDE ser cero, pero DEBERÍA ser el tiempo (hora) actual, y stop time DEBERÍA ser cero. Repeat Times, y Time Zones NO DEBERÍAN utilizarse, y si se utilizan deben ser conformes con IETF RFC 2327.

**Receive:** Si cualesquiera de estos campos estuviera presente, DEBERÍA ignorarse.

#### A.3.4.1.10 Encryption keys (claves de encriptación)

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

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

**Send:** NO SE UTILIZARÁ.

**Receive:** Este campo DEBERÍA ignorarse.

#### A.3.4.1.11 Attributes (a=) (atributos)

```
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
```

**Send:** PUEDE incluirse una o más de las líneas de atributo "a" especificadas más adelante. NO DEBERÍA utilizarse una línea de atributo que no estuviera especificada más adelante.

**Receive:** PUEDE incluirse una o más líneas de atributo "a" especificadas más adelante y se DEBE actuar en consecuencia. Líneas de atributo "a" que no estén especificadas más adelante pueden estar presentes, pero DEBEN ignorarse.

## rtpmap

**Send:** Si se utiliza este campo, su utilización DEBE ser conforme con IETF RFC 2327. PUEDE utilizarse para códecs bien conocidos y para códecs no muy conocidos. Los nombres de codificación utilizados se indican en otras dos Recomendaciones sobre IPCablecom (véanse las Recs. UIT-T J.161 y J.170).

**Receive:** El campo DEBE utilizarse de acuerdo con IETF RFC 2327.

## X-pc-codecs

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

**Receive:** El campo contiene una lista de códecs que el punto extremo distante puede utilizar para esta conexión. Los códecs NO SE UTILIZARÁN hasta que hayan sido señalizados a través de una línea de medios (m=).

## X-pc-secret

**Send:** El campo contiene un secreto de extremo a extremo que habrá de utilizarse para la seguridad de RTP y RTCP. El secreto se codifica de manera similar al parámetro clave de encriptación (k=) de IETF RFC 2327, con las siguientes constricciones:

La clave de encriptación NO CONTENDRÁ una serie de texto cifrado, sino sólo una frase de paso.

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

**Receive:** El campo contiene un secreto de extremo a extremo que habrá de utilizarse para la seguridad de RTP y RTCP.

## X-pc-csuites-rtp

## X-pc-csuites-rtcp

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

ciphersuite = [AuthenticationAlgorithm] "/" [EncryptionAlgorithm]

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

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

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

62/51

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

**Receive:** Contiene una lista de series de texto cifrado que el punto extremo distante puede utilizar para esta conexión. Para poder utilizar una serie de texto cifrado que no sea la primera en la lista es necesario que la serie de texto cifrado que se va a utilizar haya sido señalizada mediante una nueva línea de serie de texto cifrado en la cual la serie de texto cifrado que se va a utilizar aparezca en la primera posición.

#### X-pc-spi-rtcp

**Send:** El campo contiene el índice de parámetros de seguridad (SPI, *security parameter index*) IPsec que habrá de utilizarse cuando se envíen 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. Se DEBE suministrar este campo cuando se utilice la seguridad RTCP.

**Receive:** Contiene el SPI IPsec que habrá de utilizarse cuando se envíen paquetes RTCP por IPsec. El campo DEBE estar presente cuando se utilice la seguridad RTCP.

#### X-pc-bridge

**Send:** Los puntos extremos TGCP NO UTILIZARÁN este atributo.

**Receive:** Los puntos extremos TGCP DEBEN ignorar este atributo, si se recibe.

#### recvonly

**Send:** El campo DEBE utilizarse de acuerdo con IETF RFC 2543.

**Receive:** El campo DEBE utilizarse de acuerdo con IETF RFC 2543.

#### sendrecv

**Send:** El campo DEBE utilizarse de acuerdo con IETF RFC 2543.

**Receive:** El campo DEBE utilizarse de acuerdo con IETF RFC 2543.

#### sendonly

**Send:** El campo DEBE utilizarse de acuerdo con IETF RFC 2543, excepto que la dirección IP y el número de puerto NO SE PONDRÁN A CERO.

**Receive:** El campo DEBE utilizarse de acuerdo con IETF RFC 2543.

#### ptime

**Send:** El ptime DEBERÍA proporcionarse siempre y, si se utiliza, su utilización debe ser conforme con IETF RFC 2327. Cuando se utiliza un rtpmap o un códec no muy conocido, se DEBE proporcionar el ptime.

**Receive:** El campo DEBE utilizarse de acuerdo con IETF RFC 2327. Si "ptime" está presente, el MTA DEBE utilizar el ptime en el cálculo de reservas de QoS. Si "ptime" no está presente, MTA DEBE suponer valores por defecto razonables para códecs bien conocidos.

#### A.3.4.1.12 Media Announcements (m=) (anuncios de medios)

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

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

#### Media

**Send:** DEBE utilizarse el tipo de medio 'audio'.

**Receive:** El tipo recibido DEBE ser 'audio'



Port

**Send:** DEBE llenarse de acuerdo con IETF RFC 2327. El puerto especificado es el puerto de recepción, independientemente de que el tren sea unidireccional o bidireccional. El puerto de envío puede ser diferente.

**Receive:** DEBE utilizarse de acuerdo con IETF RFC 2327. El puerto especificado es el puerto de recepción. El puerto de envío puede ser diferente.

Transport

**Send:** DEBE utilizarse el protocolo de transporte "RTP/AVP".

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

Media Formats

**Send:** DEBE utilizarse un tipo de medio adecuado, definido en IETF RFC 2327.

**Receive:** De acuerdo con IETF RFC 2327.

### **A.3.5 Transmisión por UDP**

#### **A.3.5.1 Entrega fiable de mensajes**

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

Cuando no se proporciona un puerto para el punto extremo<sup>30</sup>, las instrucciones deberían enviarse al puerto MGCP por defecto, 2427.

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

Incumbe a la entidad solicitante proporcionar periodos de temporización adecuados para todas las instrucciones pendientes, e intentar de nuevo las instrucciones cuando hayan expirado los periodos de temporización. En A.3.5.2 se especifica una estrategia de retransmisión.

Además, si después de repetidas las instrucciones no se ha obtenido respuesta, se supone que la entidad de destino no está disponible. Incumbe a la entidad solicitante buscar servicios redundantes y/o liberar conexiones existentes o pendientes, como se especifica en A.2.4.

---

<sup>30</sup> A cada punto extremo puede se le puede proporcionar una dirección MGC y un puerto MGC.

### A.3.5.2 Estrategia de retransmisión

En esta Recomendación se ha evitado especificar valores estáticos para los temporizadores de retransmisión, pues estos valores generalmente dependen de la red. Normalmente, los temporizadores de retransmisión deben determinar el periodo de temporización midiendo el tiempo que transcurre entre el instante en que se envía una instrucción y el instante en que se retorna una respuesta. Las pasarelas de circuitos troncales DEBEN, como mínimo, implementar una estrategia de retransmisión utilizando temporizadores de retransmisión cuya característica de reducción tiene valores inicial y máximo configurables.

Las pasarelas de circuitos troncales DEBERÍAN utilizar el algoritmo implementado en TCP-IP, que emplea dos variables (véase, por ejemplo, *TCP/IP Illustrated, Volume 1, The Protocols*):

El retardo promedio de respuesta (AAD, *average response delay*), calculado mediante un promedio alisado exponencialmente de los retardos de respuesta observados,

- la desviación promedio, (ADEV, *average deviation*) calculada mediante un promedio alisado exponencialmente del valor absoluto de la diferencia entre el retardo de respuesta observado y el promedio actual.

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

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

- multiplicar por dos el valor estimado del retardo promedio, AAD,
- calcular un valor aleatorio, uniformemente distribuido entre 0,5 AAD y AAD,
- fijar el temporizador de retransmisión (RTO) al mínimo de estos dos valores:
  - la suma del valor aleatorio y N veces la desviación promedio.
  - $RTO_{max}$ , donde el valor por defecto para  $RTO_{max}$  es 4 segundos.

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

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

### A.3.6 Remolque

Hay casos en que el MGC desea enviar varios mensajes simultáneamente a uno o más puntos extremos en una pasarela, y viceversa. Cuando hay que enviar varios mensajes en los mismos paquetes UDP, los mensajes se separan mediante una línea de texto que contiene un punto, como por ejemplo:

```
200 2005 OK
.
DLCX 1244 ds/ds1-2/2@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
```

Los mensajes remolcados DEBEN procesarse como si se hubieran recibido en distintos datagramas; sin embargo, si hay que retransmitir un mensaje (de instrucción o respuesta), se DEBE retransmitir el datagrama entero, y no sólo el mensaje que falta. Los mensajes individuales en el datagrama DEBEN procesarse en su orden, comenzando por el primer mensaje.

Los errores encontrados en un mensaje que fue remolcado NO AFECTARÁN a ningún otro mensaje recibido en ese paquete; cada mensaje se procesa independientemente.

### A.3.7 Identificadores de transacción y triple toma de contacto

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

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

Las pasarelas MGCP pueden optar por suprimir las copias de las respuestas a transacciones cuyo identificador esté incluido en "gamas de identificadores de transacción confirmada", recibidas en un mensaje; sin embargo, el hecho de que la transacción fue ejecutada debe retenerse durante  $T_{hist}$  segundos. Asimismo, cuando se recibe un mensaje de acuse de recibo de respuesta<sup>31</sup>, la respuesta de la cual se acusa recibo puede suprimirse. Las pasarelas deben descartar las ulteriores instrucciones que reciban de ese MGC, sin ejecutar ninguna otra acción, cuando el identificador de transacción esté comprendido en estas gamas y la respuesta se emitió dentro de los últimos  $T_{hist}$ .

Sean  $term_{new}$  y  $term_{old}$  el nombre de punto extremo en una nueva instrucción,  $cmd_{new}$ , y en una antigua instrucción,  $cmd_{old}$  respectivamente. Los identificadores de transacción que habrán de ser confirmados en  $cmd_{new}$  DEBERÍAN determinarse como sigue:

- 1) Si  $term_{new}$  no contiene comodines:
  - a) respuestas no confirmadas a instrucciones antiguas, donde  $term_{old}$  es igual a  $term_{new}$ .
  - b) Facultativamente, una o más respuestas no confirmadas, donde  $term_{old}$  contenía el comodín "any-of", y el nombre de punto extremo retornado en la respuesta era  $term_{new}$ .
  - c) Facultativamente, una o más respuestas no confirmadas, donde  $term_{old}$  contenía el comodín "all", y  $term_{new}$  está abarcado por el comodín en  $term_{old}$ .
  - d) Facultativamente, una o más respuestas no confirmadas, donde  $term_{old}$  contenía el comodín "any-of", no se retornó ningún nombre de punto extremo, y  $term_{new}$  está abarcado por el comodín en  $term_{old}$ .
- 2) Si  $term_{new}$  contiene el comodín "all":
  - a) Facultativamente, una o más respuestas no confirmadas, donde  $term_{old}$  contenía el comodín "all", y  $term_{new}$  está abarcado por el comodín en  $term_{old}$ .
- 3) Si  $term_{new}$  contiene el comodín "any of":
  - a) Facultativamente, una o más respuestas no confirmadas, donde  $term_{old}$  contenía el comodín "all", y  $term_{new}$  está abarcado por el comodín en  $term_{old}$  si el comodín "any of" en  $term_{new}$  había sido reemplazado por el comodín "all".

Una determinada respuesta NO DEBERÍA ser confirmada en dos mensajes distintos.

---

<sup>31</sup> Por oposición a una instrucción con un parámetro acuse de recibo de respuesta.

Los siguientes ejemplos ilustran la aplicación de estas reglas:

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

Los valores de "gammas de identificadores de transacción confirmada" NO DEBERÍAN utilizarse si han transcurrido más de  $T_{t_{hist}}$  segundos desde que la pasarela envió su última respuesta a ese MGC, o cuando una pasarela ha reanudado la operación. En esta situación, las instrucciones deben aceptarse y procesarse sin efectuar ninguna prueba sobre el identificador de transacción.

Asimismo, una respuesta NO DEBERÍA confirmarse si no se recibió dentro de los últimos  $T_{t_{hist}}$  segundos.

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

### A.3.8 Respuestas provisionales

En algunos casos, los tiempos de compleción de las transacciones son considerablemente más largos que en otros. TGCP utiliza UDP como el protocolo de transporte, y la fiabilidad se consigue mediante retransmisiones selectivas basadas en la expiración de un periodo de temporización, que se determina por una estimación de la suma del tiempo de propagación de ida y retorno de la red y el tiempo de compleción de las transacciones. Una varianza significativa del tiempo de compleción de las transacciones plantea por tanto problemas cuando se desea una rápida detección de la pérdida de mensajes sin una tara excesiva.

En consecuencia, para resolver este problema DEBE emitirse una respuesta provisional únicamente si el tiempo de compleción de las transacciones es mayor que algún corto periodo de tiempo. La respuesta provisional acusa recibo de la instrucción aunque todavía no se sepa el resultado de ésta, por ejemplo por estar pendiente una reserva de recurso. Como una directriz, una transacción que requiera la compleción de una comunicación externa, , por ejemplo, la reserva de un recurso de red, debe emitir una respuesta provisional. Además, si se recibe una instrucción CreateConnection o ModifyConnection duplicada, y todavía no ha terminado la ejecución de la transacción, se DEBE enviar en retorno una respuesta provisional.

Una semántica transaccional pura implicaría que las respuestas provisionales no debieran retornar otra información que la que comunica el hecho de que la transacción se encuentra en ejecución en ese momento; sin embargo, un enfoque optimista que retorne algunas otras informaciones permite reducir el retardo del sistema.

Las respuestas provisionales DEBEN enviarse solamente en respuesta a una instrucción CreateConnection o ModifyConnection. Para reducir el retardo del sistema, en la respuesta provisional a la instrucción CreateConnection se DEBE incluir un identificador de conexión y una descripción de sesión. Si la instrucción ModifyConnection habrá de retornar una descripción de sesión, la descripción de sesión se DEBE también incluir en la respuesta provisional, también en este caso. Si la transacción concluye con éxito, la información retornada en la respuesta provisional DEBE repetirse en la respuesta final. Se considera un error de protocolo no repetir esta información o cambiar cualquiera de las informaciones previamente suministradas en una respuesta exitosa. Si la transacción fracasa, se retorna un código de error; la información retornada anteriormente deja de ser válida.

Una transacción CreateConnection o ModifyConnection que esté en ejecución DEBE anularse si se recibe una instrucción DeleteConnection para el punto extremo. En ese caso, de todas formas, se DEBERÍA retornar automáticamente una respuesta para la transacción anulada, y se DEBE retornar una respuesta para la transacción anulada si se detecta una retransmisión de la transacción anulada.

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

Cuando termina la ejecución de la transacción, se envía la respuesta final, y se suprime la respuesta provisional, la cual ha perdido su vigencia. Para asegurar una rápida detección de una respuesta final perdida se DEBE acusar recibo de las respuestas finales emitidas después de respuestas provisionales. El punto extremo DEBE por tanto incluir un parámetro "ResponseAck" vacío en esas respuestas finales, y sólo en esas. La presencia del parámetro "ResponseAck" en la respuesta final ocasionará el envío de una respuesta "acuse de recibo de respuesta" en retorno, al punto extremo. La respuesta "acuse de recibo de respuesta" incluirá, en el encabezamiento de respuesta, el identificador de transacción de la respuesta de la que se acusa recibo. La recepción de esta respuesta "acuse de recibo de respuesta" está sometida al mismo periodo de temporización y a las mismas estrategias y procedimientos de retransmisión que las respuestas a las instrucciones (véase A.2.4), es decir, el emisor de la respuesta final la retransmitirá si el "acuse de recibo de respuesta" no se recibe en tiempo oportuno. No se acusa recibo de la respuesta "acuse de recibo de respuesta".

#### **A.4 Seguridad**

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

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

## Anexo A.A

### Lotes de eventos

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

Cada lote contiene un nombre de lote definido, para el lote, y códigos y definiciones de evento para cada uno de los eventos en el lote. Los cuadros de eventos/señales para cada lote tienen cinco columnas:

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

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

## A.A.1 Lote de circuitos troncales SUP

Nombre de lote: IT.

**Cuadro A.A.1/J.171 – Eventos y señales del lote de circuitos troncales SUP**

Código	Descripción	Evento	Señal	Información adicional
co1	Tono de continuidad 1	√	TO	Temporización = 3 segundos
co2	Tono de continuidad 2	√	TO	Temporización = 3 segundos
ft	Tono de facsímil	√	–	
ld	Conexión de larga duración	C	–	
ma	Comienzo de medio	C	–	
mt	Tono de módem	√	–	
oc	Operación completa	√	–	
of	Fracaso de operación	√	–	
ro	Tono de volver a llamar	–	TO	Temporización = 30 segundos
rt	Tono de llamada de retorno	–	C, TO	Temporización = 180 segundos
TDD	Tonos de dispositivos de telecomunicaciones para sordos (TDD, <i>telecommunications devices for the deaf</i> )	√		

Las definiciones de los distintos eventos y señales son las siguientes:

**Tono de continuidad 1 (co1):** Un tono de 2010 Hz de acuerdo con la Rec. UIT-T Q.724. Para ajustarse a las actuales prácticas en materia de pruebas de continuidad, el evento NO DEBERÍA generarse hasta que se hubiera suprimido el tono. El tono es de tipo TO; el tono de continuidad sólo se aplicará durante el periodo de tiempo especificado. El proceso de provisión puede modificar el valor por defecto.

**Tono de continuidad 2 (co2):** Un tono de 1780 Hz de acuerdo con la Rec. UIT-T Q.724. Para ajustarse a las actuales prácticas en materia de pruebas de continuidad, el evento NO DEBERÍA generarse hasta que se hubiera suprimido el tono. El tono es de tipo TO; el tono de continuidad sólo se aplicará durante el periodo de tiempo especificado. El proceso de provisión puede modificar el valor por defecto.

Los tonos de continuidad se utilizan cuando el MGC desea iniciar una prueba de continuidad. Hay dos tipos de prueba de continuidad: con un solo tono y con dos tonos. La parte que inicia la prueba de continuidad comprueba las señales y detecta los tonos adecuados para el circuito troncal en cuestión. Por ejemplo, para una prueba de continuidad de un circuito a cuatro hilos a un circuito a dos hilos podrían utilizarse los siguientes mensajes:

*Pasarela de origen*

```
RQNT 1234 ds/ds3-1/ds1-6/17@tgw1.example.net
X: AB123FE0
S: co2
R: co1
```

## *Pasarela de terminación*

```
CRCX 1234 ds/ds1-4/7@tgw2.example.net
C: A3C47F21456789F0
L: p:10, a:PCMU
M: conttest
```

La pasarela de origen envía la señal solicitada, y busca el retorno del tono adecuado para el circuito troncal en cuestión. Cuando detecta ese tono y si considera que la prueba de continuidad ha tenido éxito, genera el evento "co1" que en el ejemplo se notifica al MGC. Si la prueba no tiene éxito antes de la expiración del periodo de temporización, se genera un evento de operación completa y se envía en este caso al MGC. De manera similar, si se produce un error antes de la expiración del periodo de temporización, se genera un evento "fallo de operación". Los eventos "oc" y "of" se parametrizarán con el nombre del evento/señal que ellos informan, esto es, "co1" en este caso.

**Tono de facsímil (ft):** El evento tono de facsímil se genera cuando se detecta una comunicación de facsímil (véase, por ejemplo, la Rec. UIT-T T.30, o la Rec. UIT-T V.21).

**Conexión de larga duración (ld):** Se detecta la "conexión de larga duración" cuando se determina que una conexión ha estado establecida durante un periodo de tiempo mayor que un determinado periodo de tiempo. El valor por defecto es una hora; sin embargo, el proceso de provisión puede modificar este valor por defecto.

El evento puede detectarse en una conexión. Cuando no se especifica una conexión, el evento es aplicable a todas las conexiones para el punto extremo, cualquiera que sea el momento en que se crearon las conexiones.

**Comienzo de medio (ma):** El evento comienzo de medio se produce en una conexión cuando se recibe por ésta el primer paquete válido de medios RTP<sup>32</sup>. Este evento puede utilizarse para sincronizar una señal local, por ejemplo, la señal de llamada de retorno, con la llegada de medios procedentes de la otra parte.

El evento puede detectarse en una conexión. Cuando no se especifica una conexión, el evento es aplicable a todas las conexiones para el punto extremo, cualquiera que sea el momento en que se crearon las conexiones.

**Tonos de módem (mt):** El evento tonos de módem se genera cuando se detecta una comunicación en la que interviene un módem (véase por ejemplo, la Rec. UIT-T V.8).

**Operación completa (oc):** El evento operación completa se genera cuando se pide a la pasarela que aplique una o más señales de tipo TO en el punto extremo, y una o más de esas señales han concluido sin que hayan sido detenidas por la detección de un evento solicitado, como por ejemplo un "tono de continuidad 1". El informe de compleción puede contener como parámetro el nombre de la señal que llegó al final de su tiempo de vida, como en:

```
O: IT/oc(IT/co1)
```

Cuando la señal informada se aplica en una conexión, el parámetro suministrado incluirá también el nombre de la conexión, como por ejemplo:

```
O: IT/oc(IT/rt@0A3F58)
```

Cuando se solicita el evento operación completa, no se le puede añadir ningún parámetro de evento. Si se omite el nombre de lote, se supone que tiene el nombre de lote por defecto.

---

<sup>32</sup> Cuando se utilizan servicios de seguridad de autenticación e integridad, un paquete RTP no se considerará válido hasta que haya pasado las verificaciones de seguridad.



El evento operación completa se puede generar también como se define en el protocolo de base, por ejemplo, cuando una instrucción ModifyConnection insertada concluye con éxito, como en<sup>33</sup>:

O: IT/oc(B/C)

**Fracaso de la operación (of):** En general, el evento fallo de la operación puede generarse cuando se pide al punto extremo que aplique una o varias señales de tipo TO en el punto extremo, y una o más de esas señales fallan antes de la expiración del periodo de temporización. El informe de compleción puede contener como parámetro el nombre de la señal que falló, como en:

O: IT/of(IT/co2)

Cuando la señal informada se aplica en una conexión, el parámetro suministrado incluirá también el nombre de la conexión, como por ejemplo en:

O: IT/of(IT/rt@0A3F58)

Cuando se solicita el evento fallo de la operación, no se puede especificar parámetros de evento. Si se omite el nombre de lote, se supone que tiene el nombre de lote por defecto.

El evento fallo de operación se puede generar también como se define en el protocolo de base, por ejemplo, cuando una instrucción ModifyConnection insertada fracasa, como en<sup>33</sup>:

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

**Tono de volver a llamar (ro):** El tono de volver a llamar, conocido también por tono de congestión, se especifica en la Rec. UIT-T E.180/Q.35.

**Tono de llamada de retorno (rt):** El tono de llamada audible se especifica en la Rec. UIT-T E.180/Q.35. La definición del tono viene determinada por las características nacionales del tono de llamada de retorno y PUEDE establecerse en el proceso de provisión. La señal de llamada de retorno puede aplicarse a un punto extremo y a una conexión.

**Tonos de dispositivos de telecomunicaciones para sordos (TDD):** El evento TDD se genera cuando se detecta una comunicación TDD (véase por ejemplo, la Rec. UIT-T V.18).

## Apéndice A.I

### Interacciones de modos

Una conexión MGCP puede establecer uno o más trenes de medios. Estos trenes de medios son entrantes (procedentes de un punto extremo distante) o salientes (generados en el punto extremo del circuito). El parámetro "modo de conexión" determina el sentido de transmisión y la generación de estos trenes. Cuando un punto extremo tiene establecida una sola conexión, la correspondencia de estos trenes es directa; el punto extremo del circuito aplica el tren entrante al circuito y genera el tren saliente a partir de la señal del circuito, de acuerdo con el modo indicado por el parámetro modo.

En cambio, cuando un punto extremo tiene establecidas varias conexiones, puede haber muchos trenes entrantes y salientes. Según el modo de conexión utilizado, estos trenes pueden interactuar diferentemente unos con otros, y con los trenes que van hacia el punto extremo o que vienen del punto extremo.

---

<sup>33</sup> Obsérvese que en este ejemplo se utiliza "B" como el prefijo para el parámetro informado.

El cuadro A.I.1 describe la forma en que deben combinarse diferentes conexiones cuando una o más conexiones están concurrentemente "activas". En este contexto, por conexión activa ha de entenderse una conexión que está en uno de los modos siguientes:

- "envío/recepción";
- "sólo envío";
- "sólo recepción".

**Cuadro A.I.1/J.171 – Reagrupamiento de diferentes conexiones cuando una o más conexiones están simultáneamente activas**

		Modo de la conexión A					
		sendonly	recvonly	sendrecv	loopback/ conttest	inactive	netwloop/ netwtest
Modo de la conexión B	sendonly	$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=NA$	$A_{out}=NA$ $B_{out}=H_{in}$ $H_{out}=A_{in}$	$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=A_{in}$	$A_{out}=NA$ $B_{out}=NA$ $H_{out}=cot$	$A_{out}=NA$ $B_{out}=H_{in}$ $H_{out}=NA$	$A_{out}=A_{in}$ $B_{out}=H_{in}$ $H_{out}=NA$
	recvonly		$A_{out}=NA$ $B_{out}=NA$ $H_{out}=A_{in}+B_{in}$	$A_{out}=H_{in}$ $B_{out}=NA$ $H_{out}=A_{in}+B_{in}$	$A_{out}=NA$ $B_{out}=NA$ $H_{out}=cot$	$A_{out}=NA$ $B_{out}=NA$ $H_{out}=B_{in}$	$A_{out}=A_{in}$ $B_{out}=NA$ $H_{out}=B_{in}$
	sendrecv			$A_{out}=H_{in}$ $B_{out}=H_{in}$ $H_{out}=A_{in}+B_{in}$	$A_{out}=NA$ $B_{out}=NA$ $H_{out}=cot$	$A_{out}=NA$ $B_{out}=H_{in}$ $H_{out}=B_{in}$	$A_{out}=A_{in}$ $B_{out}=H_{in}$ $H_{out}=B_{in}$
	loopback/ conttest				$A_{out}=NA$ $B_{out}=NA$ $H_{out}=cot$	$A_{out}=NA$ $B_{out}=NA$ $H_{out}=cot$	$A_{out}=NA$ $B_{out}=NA$ $H_{out}=cot$
	inactive					$A_{out}=NA$ $B_{out}=NA$ $H_{out}=NA$	$A_{out}=A_{in}$ $B_{out}=NA$ $H_{out}=NA$
	netwloop/ netwtest						$A_{out}=A_{in}$ $B_{out}=B_{in}$ $H_{out}=NA$

Las conexiones que están en los modos "conexión en bucle de red", "prueba de continuidad de red", o "inactivo" no son afectadas por conexiones que se encuentran en el modo "activo". En el cuadro A.I.1 se han seguido los siguientes convenios:

- $A_{in}$  es el tren de medios entrante procedente de la conexión A;
- $B_{in}$  es el tren de medios entrante procedente de la conexión B;
- $H_{in}$  es el tren de medios entrante procedente del circuito troncal;
- $A_{out}$  es el tren de medios saliente que va a la conexión A;
- $B_{out}$  es el tren de medios saliente que va a la conexión B;
- $H_{out}$  es el tren de medios saliente que va al punto extremo, donde "cot" indica prueba de continuidad, sea en el modo "prueba de continuidad", sea en el modo "conexión en bucle";
- NA indica que no hay ningún tren.

## Apéndice A.II

### Ejemplos de codificación de instrucciones

En este apéndice se presentan ejemplos de instrucciones y respuestas, con las formas de codificación realmente utilizadas. Se presentan ejemplos para cada instrucción. Todos los comentarios que aparecen en las instrucciones y respuestas son facultativos.

#### A.II.1 NotificationRequest

El primer ejemplo muestra una NotificationRequest que inicia una prueba de continuidad y busca la verificación de la prueba. La "entidad notificada" para el punto extremo se fija a "ca@ca1.whatever.net:5678" y el RequestIdentifier se repite en la correspondiente instrucción Notify:

```
RQNT 1201 ds/ds1-1/2@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
N: mgc@mgc1.whatever.net:5678
X: 0123456789AC
R: col, oc(N), of(N)
S: col
```

La respuesta indica que la transacción tuvo éxito:

```
200 1201 OK
```

#### A.II.2 Notify

El ejemplo que sigue ilustra un mensaje Notify que notifica una prueba de continuidad exitosa como se indica por los eventos observados. Como en la NotificationRequest que ocasionó la activación se especificó una "entidad notificada", ésta se repite aquí. Además, se incluye el RequestIdentifier para correlacionar esta instrucción Notify con la instrucción NotificationRequest que ocasionó la activación:

```
NTFY 2002 ds/ds1-1/2@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
N: mgc@mgc1.whatever.net:5678
X: 0123456789AC
O: col
```

La respuesta Notify indica que la transacción tuvo éxito:

```
200 2002 OK
```

#### A.II.3 CreateConnection

El primer ejemplo ilustra una instrucción CreateConnection para crear una conexión en el punto extremo especificado. La conexión será parte del CallId especificado. Las LocalConnectionOptions especifican que el códec utilizará la ley  $\mu$  de la Rec. UIT-T G.711 y que el periodo de paquetización será de 10 ms. El modo de conexión será "receive only":

```
CRCX 1204 ds/ds1-1/17@tgw2.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: recvonly
```

La respuesta indica que la transacción tuvo éxito, por lo que se incluye un identificador de conexión para la conexión recientemente creada. También se incluye una descripción de sesión para la nueva conexión (obsérvese que va precedida de una línea vacía).

```
200 1204 OK
I: FDE234C8
```

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

El segundo ejemplo ilustra una instrucción CreateConnection que contiene una petición de notificación y un RemoteConnectionDescriptor:

```
CRCX 1205 ds/ds1-1/1@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: recvonly
X: 0123456789AD
R: MO/sup(addr(K0, 4,1,1, s2), id(K0,0,0,7,3,2,5,5,5,1,2,3,4,s0))
S: MO/ans
v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 0
```

La respuesta indica que la transacción fracasó porque el circuito troncal ya había sido tomado. En consecuencia, no se retorna un identificador de conexión ni una descripción de sesión:

```
401 2005 Circuit already seized
```

En el tercer ejemplo se ilustra la utilización de la respuesta provisional y la triple toma de contacto:

```
CRCX 1206 ds/ds1-1/1@tgw.whatever.net MGCP 1.0 TGCP 1.0
K: 1205
C: A3C47F21456789F0
L: p:10, a:PCMU
M: inactive
v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 0
```

Inicialmente se retorna una respuesta provisional:

```
100 1206 Pending
I: DFE233D1
```

```
v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 3456 RTP/AVP 0
```

Poco después se recibe la respuesta final:

```
200 1206 OK
K:
I: DFE233D1
```

```
v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 3456 RTP/AVP 0
```

El MGC acusa recibo de la respuesta final, como se solicitó:

```
000 1206
```

y la transacción está completa.

#### **A.II.4 ModifyConnection**

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

```
MDCX 1209 ds/ds1-1/21@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
N: mgc@mgc1.whatever.net
M: sendrecv
```

La respuesta indica que la transacción tuvo éxito:

```
200 1209 OK
```

En el segundo ejemplo, se transfiere una descripción de sesión y se incluye una petición de notificación con la instrucción ModifyConnection. El punto extremo comenzará a aplicar tonos de llamada de retorno a la red telefónica pública conmutada (RTPC) y los aplicará hasta que detecte audio en la conexión especificada para la señal de llamada de retorno:

```
MDCX 1210 ds/ds1-1/3@abc5.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
M: recvonly
X: 0123456789AE
R: ma@ FDE234C8
S: rt
```

```
v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 3456 RTP/AVP 0
```

La respuesta indica que la transacción tuvo éxito:

```
200 1206 OK
```

#### **A.II.5 DeleteConnection (procedente del controlador de pasarela de medios (MGC))**

En este ejemplo, el MGC ordena simplemente a la pasarela de circuitos troncales que suprima la conexión FDE234C8 en el punto extremo especificado:

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

La respuesta indica que la transacción tuvo éxito, y que se suprimió la conexión. Por tanto, también se incluyen parámetros (de conexión) para la conexión:

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

#### **A.II.6 DeleteConnection (procedente de la pasarela de circuitos troncales)**

En este ejemplo, la pasarela de circuitos troncales envía una instrucción DeleteConnection al MGC para comunicarle que se ha suprimido una conexión en el punto extremo especificado. El ReasonCode especifica el motivo para la supresión, y también se proporciona parámetros (de conexión) para la conexión:

```
DLCX 1210 ds/ds1-1/1@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
E: 900 - Hardware error
P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48
```

El MGC envía una respuesta de éxito a la pasarela:

```
200 1210 OK
```

#### **A.II.7 DeleteConnection (procedente del controlador de pasarela de medios, en caso de múltiples conexiones)**

En el primer ejemplo, el MGC ordena a la pasarela de circuitos troncales que suprima todas las conexiones relacionadas con la llamada "A3C47F21456789F0" en el punto extremo especificado:

```
DLCX 1210 ds/ds1-1/6@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
```

La respuesta indica éxito y que la conexión o conexiones fueron suprimidas:

```
250 1210 OK
```

En el segundo ejemplo, el MGC ordena a la pasarela de circuitos troncales que suprima todas las conexiones relacionadas con todos los puntos extremos especificados:

```
DLCX 1210 ds/ds1-1/*@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
```

La respuesta indica éxito:

```
250 1210 OK
```

#### **A.II.8 AuditEndpoint**

En el primer ejemplo, el MGC desea averiguar qué puntos extremos están presentes en la pasarela de circuitos troncales especificada, por lo que utiliza el comodín "all of" para la porción local del nombre de punto extremo. El MGC sólo desea dos nombres de punto extremo:

```
AUEP 1200 *@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
ZM: 2
```

La pasarela de circuitos troncales indica éxito e incluye una lista de hasta dos nombres de punto extremo. Un total de 24 nombres de punto extremo concordaron con el comodín especificado:

```
200 1200 OK
Z: ds/ds1-1/1@tgw-2567.whatever.net
Z: ds/ds1-1/2@tgw-2567.whatever.net
ZN: 24
```

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

```
AUEP 1201 ds/ds1-1/1@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
F: A
```

La respuesta indica éxito, y también las capacidades. Dos códigos están soportados, pero con capacidades diferentes. En consecuencia se retornan dos conjuntos distintos de capacidades:

```
200 1201 OK
A: a:PCMU, p:10-100, e:on, s:off, v:IT, m:sendonly;recvonly;sendrecv;
    inactive;loopback;conttest;netwloop;netwtest
A: a:G728, p:30-90, e:on, s:on, v:IT, m: sendonly;recvonly;sendrecv;
    inactive;loopback;conttest;netwloop
```

En el tercer ejemplo, el MGC auditoriza todas las informaciones posibles relacionadas con el punto extremo:

```
AUEP 2002 ds/ds1-1/1@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
F: R, S,X,N,I,T,O,ES
```

La respuesta indica éxito:

```
200 2002 OK
R: IT/ft,mt(N)
S:
X: 0123456789B1
N: [128.96.41.12]
I: 32F345E2
T: ft
O:
ES:
```

La lista de eventos solicitados contiene dos eventos. Cuando no se especifica ningún nombre de lote, se supone el lote por defecto. La misma regla es aplicable a las acciones, por lo que se debe suponer la acción por defecto (Notify) para el evento "IT/ft". El hecho de que no se haya dado ningún valor para las "SignalRequests" significa que no hay señales activas en ese momento. La "entidad notificada" actual hace referencia a una dirección IP y existe una sola conexión para el punto extremo. El valor actual de DetectEvents es "ft", y la lista de ObservedEvents está vacía, como también lo está la de EventStates.

### A.II.9 AuditConnection

El primer ejemplo muestra una instrucción AuditConnection en la que se auditorizan los parámetros CallId, NotifiedEntity, LocalConnectionOptions, ConnectionMode, LocalConnectionDescriptor, y Connection Parameters:

```
AUCX 2003 ds/ds1-1/18@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
I: 32F345E2
F: C,N,L,M,LC,P
```

La respuesta indica éxito e incluye información para la RequestedInfo:

```
200 2003 OK
C: A3C47F21456789F0
N: mgc@mgc1.whatever.net
L: p:10, a:PCMU
M: sendrecv
P: PS=395, OS=22850, PR=615, OR=30937, PL=7, JI=26, LA=47

v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
```

```
t=0 0
m=audio 1296 RTP/AVP 0
```

En el segundo ejemplo se solicita auditar RemoteConnectionDescriptor y LocalConnectionDescriptor:

```
AUCX 1203 ds/ds1-1/2@tgw.whatever.net MGCP 1.0 TGCP 1.0
I: FDE234C8
F: RC,LC
```

La respuesta indica éxito, e incluye información para la RequestedInfo. En este caso no existe RemoteConnectionDescriptor, por lo que sólo se incluye el campo versión de protocolo para el RemoteConnectionDescriptor:

```
200 1203 OK
```

```
v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 1296 RTP/AVP 0
```

```
v=0
```

### **A.II.10 RestartInProgress**

El primer ejemplo ilustra un mensaje RestartInProgress enviado por una pasarela de circuitos troncales para informar al MGC que el punto extremo será retirado del servicio en 300 segundos:

```
RSIP 1200 ds/ds1-1/1@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
RM: graceful
RD: 300
```

La respuesta del MGC indica que la transacción tuvo éxito:

```
200 1200 OK
```

En el segundo ejemplo, el mensaje RestartInProgress enviado por la pasarela de circuitos troncales informa al MGC que todos los puntos extremos de la pasarela de circuitos troncales se ponen en servicio en 0 segundos, esto es, que están restablecidos al servicio. También se hubiera podido omitir el plazo:

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

La respuesta del MGC indica éxito y, además, proporciona a los puntos extremos en cuestión una nueva "entidad notificada":

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

Como otra posibilidad, la respuesta podría haber indicado que la transacción ha fracasado, y proporcionar una nueva "entidad notificada", como en:

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

En tal caso, hubiera sido necesario reintentar la instrucción para cumplimentar el "procedimiento de reenganche" (véase A.2.4.3.5), esta vez yendo a MGC "MGC-1@whatever.net".



## Apéndice A.III

### Ejemplo de flujo de llamadas

En este apéndice se presenta un ejemplo de flujo de llamadas entre un usuario on-net que utiliza un MTA y un protocolo de señalización<sup>34</sup> no especificados y un usuario off-net al que se gana acceso a través de una pasarela de circuitos troncales que utiliza el protocolo TGCP y una pasarela de señalización que soporta la señalización de PU de la RDSI del SS7. Debe hacerse observar que este flujo de llamadas, aunque es válido, no representa más que un ejemplo, que puede o no seguirse en la práctica.

En el flujo de llamadas que se muestra en la figura A.III.1, CMS (*call management server*) designa el servidor de gestión de llamadas, MGC (*media gateway controller*) designa el controlador de pasarela de medios, TGW (*trunking gateway*) designa la pasarela de circuitos troncales, y SG (*signaling gateway*) designa la pasarela de señalización:

MTA	CMS	MGC	TGW	SG
<i>Efectuar llamada (E.164)</i>	→			
	<i>Establecimiento de llamada</i>	→		
		Crear Conexión(SDP1) + Petición de notificación	→	
		←	Ack(SDP2)	
		IAM	———	→
		←	Notificar	
		Ack + Modificar conexión + Petición de notificación	→	
		COT	———	→
		←	———	ACM
		←	<i>Aviso</i>	
←	<i>Aviso</i>	←	———	ANM
		Modificar conexión	→	
		←	Ack	
		←	<i>Respuesta</i>	
←	<i>Respuesta</i>			
		(Llamada establecida)		
<i>Colgar</i>	→			
	<i>Liberación</i>	→		
		REL	———	→
		Suprimir conexión	→	
		←	Ack(Perf data)	
		←	———	RLC
		(Fin de llamada )		

**Figura A.III.1/J.171 – Ejemplo de flujo de llamada**

<sup>34</sup> Este protocolo podría ser NCS o DCS.

Durante estos intercambios, el MGC utiliza el perfil TGCP del MGCP para controlar la pasarela de circuitos troncales. Se supone que entre el MTA, CMS MGC se aplica un protocolo no identificado.

Se ha partido del supuesto de que el MTA indica (directa o indirectamente) al MGC su deseo de establecer una comunicación vocal con un número de teléfono E.164 y que ha añadido una descripción de sesión a su petición. El CMS busca el número E.164 solicitado y determina que necesita hacer una llamada off-net, por lo que establece contacto con el MGC apropiado. El MGC determina que debe efectuar la llamada a través de la pasarela de circuitos troncales `tgw.whatever.net`. Además, el MGC determina que se debe realizar una prueba de continuidad para esta llamada.

La primera instrucción es una instrucción combinada `CreateConnection` y `NotificationRequest` enviada a la pasarela de circuitos troncales:

```
CRCX 2001 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: inactive
X: 0123456789B0
R: co2, oc, of
S: col
```

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

En ese punto, se ordena a la pasarela de circuitos troncales que inicie la prueba de continuidad, que busque el resultado de la prueba, y que lo informe. La generación de la señal de prueba de continuidad y la detección de su éxito (o fracaso) mediante el mecanismo de eventos están sincronizadas, de manera que cuando se produce el evento "co2", se detiene la prueba "co1". La parte (de la instrucción) relativa a la creación de conexión ordena crear una conexión inactiva en el punto extremo especificado, de acuerdo con la Rec. UIT-T G.711, con un periodo de paquetización de 10 ms. Además, la instrucción incluye la descripción de sesión recibida del MTA de origen.

La pasarela de circuitos troncales de egreso acusará recibo de la instrucción, y enviará en la descripción de sesión sus propios parámetros, como la dirección, puertos y perfil RTP, así como el identificador de conexión para la nueva conexión:

```
200 2001 OK
I: 32F345E2
```

```
v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 1297 RTP/AVP 0
```

El MGC envía un mensaje IAM SS7 a través de la pasarela de señalización al conmutador conectado al circuito troncal por el que se efectúa la llamada. En este mensaje se incluye una indicación de que se ha realizado la prueba de continuidad.

Se ha supuesto también que la prueba de continuidad tiene éxito, por lo que se genera el evento "co2" y se notifica al MGC:

```
NTFY 3001 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
X: 0123456789B0
O: co2
```

El MGC envía un mensaje COT SS7 que indica "éxito de la prueba de continuidad" al conmutador distante y acusa recibo de la instrucción Notify recibida. Asimismo, envía en remolque una instrucción combinada ModifyConnection y NotificationRequest por la que ordena a la pasarela que efectúe la conexión en modo "sólo recepción" y comience la búsqueda tonos de facsímil y de módem:

```
200 3001 OK
```

```
.
MDCX 2006 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: 32F345E2
M: recvonly
X: 0123456789B0
R: ft,mt
```

En esta fase, el MGC ha establecido un trayecto de transmisión semidúplex. El teléfono asociado al MTA de ingreso podrá recibir las señales, tales como tonos y anuncios, que podrían generarse en caso de error, así como el material hablado que muy probablemente se generará cuando el usuario de egreso conteste al teléfono.

El MGC recibe entonces un mensaje ACM SS7 que indica que se está avisando a la parte llamada, y seguidamente un mensaje ANM SS7 que indica que la parte llamada ha constado. El MGC efectúa la conexión en modo dúplex enviando la siguiente instrucción ModifyConnection a la pasarela de circuitos troncales:

```
MDCX 2007 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: 32F345E2
M: sendrecv
```

La pasarela de circuitos troncales responde inmediatamente a la instrucción con:

```
200 2007 OK
```

En paralelo, el MGC informa al MTA de origen sobre el evento contestación a la llamada y registra la hora de la contestación a la llamada.

En este momento, la llamada está totalmente establecida.

En cierto momento ulterior, en el escenario que se ha presentado, se cuelga el teléfono asociado al MTA de origen, y se transmite al MGC (directamente, o a través del CMS como se muestra en este ejemplo) un evento de colgar, con lo que se indica al MGC que se debe terminar la llamada.

El MGC verifica que la llamada debe realmente desconectarse, por ejemplo por el hecho de que no existe un dispositivo de retención de llamada, y envía entonces un mensaje REL SS7 al conmutador distante, y también una instrucción DeleteConnection a la pasarela de circuitos troncales:

```
DLCX 2009 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: 32F345E2
```

La pasarela de circuitos troncales responderá con un acuse de recibo que incluirá los parámetros de conexión para la conexión:

```
250 2009 OK
P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48
```

El MGC recibe también una confirmación de la desconexión de la llamada en forma de un mensaje RLC SS7 y, por último, registra la terminación de la llamada.

## Apéndice A.IV

### Requisitos de los puntos extremos

En este apéndice se define un conjunto de requisitos específicos de los puntos extremos TGCP.

#### A.IV.1 Modos de conexión soportados

En el cuadro A.IV.1 se indican los modos de conexión que un determinado punto extremo TGCP DEBE soportar:

**Cuadro A.IV.1/.171 – Lista de modos de conexión que deben estar soportados por un punto extremo TGCP**

Tipo de punto extremo	Información adicional de punto extremo	sendonly	recvonly	sendrecv	inactive	loopback	conttest	netwloop	netwtest
DS-0	Circuito troncal PU-RDSI	√	√	√	√	√	√	√	√
DS-0	Circuito troncal MF	√	√	√	√	–	–	√	√

## Apéndice A.V

### Información sobre compatibilidad

Este apéndice proporciona información de compatibilidad relativa al protocolo TGCP.

#### A.V.1 Compatibilidad con NCS

Esta versión de TGCP se basa en la Rec. UIT-T J.162, con la que se ha alineado lo más posible. Puesto que TGCP y NCS están destinados a dos tipos diferentes de pasarelas, presentan ciertas diferencias, que se recapitulan a continuación:

- **Modos de conexión:** NCS y TGCP tienen un conjunto de modos de conexión que son comunes a ambos protocolos, pero cada uno de ellos tiene un conjunto de modos de conexión que no están soportados por el otro:
  - NCS soporta los modos de conexión "conferencia" y "replicación", que no están soportados por TGCP.
  - TGCP soporta los modos de conexión "prueba de continuidad" y "conexión en bucle", que no están soportados por NCS.
- **Mapas de dígitos:** TGCP no soporta mapas de dígitos; en cambio, NCS sí los soporta. Esto tiene algunas implicaciones, por ejemplo:
  - Ninguna instrucción TGCP puede aceptar un mapa de dígitos como parámetro.
  - La acción "acumular de acuerdo con mapa de dígitos" no está soportada en TGCP.

- El "mapa de dígitos" no puede ser auditorizado.
- **Calidad de servicio dinámica:** NCS soporta la señalización de calidad de servicio dinámica IPCablecom, la cual no está soportada en TGCP.

Además de las diferencias antes mencionadas relativas a los modos de conexión de los protocolos, entre NCS y TGCP existen también las siguientes diferencias que no se relacionan con los protocolos.

- **Lotes de eventos:** Los lotes de eventos iniciales en TGCP NCS son diferentes.
- **Esquema de denominación de puntos extremos:** Los esquemas de denominación de los puntos extremos TGCP y de los puntos extremos NCS son algo diferentes.

#### A.V.2 Compatibilidad con MGCP

TGCP (y NCS) es además un perfil de IETF RFC 2705 MGCP 1.0; sin embargo, TGCP también ha introducido algunas adiciones. A continuación se indican las adiciones introducidas TGCP que no están incluidas actualmente en MGCP:

- **Esquema de denominación de puntos extremos:** Se ha introducido un esquema de denominación de puntos extremos específico para puntos extremos DS-0. Las reglas para uso de comodín son más estrictas que en MGCP; también se ha introducido el concepto de "gama" para puntos extremos DS-0.
- **ModifyConnection insertada:** Se ha introducido una nueva acción ModifyConnection insertada.
- **Seguridad:** Los servicios de seguridad IPCablecom están soportados en TGCP. Esto influye en las LocalConnectionOptions, Capabilities, y SDP.
- **Recuperación de nombres de punto extremo:** La instrucción AuditEndpoint se ha ampliado con una capacidad para retornar el número de puntos extremos que son abarcados por un comodín, y con un mecanismo para la recuperación, bloque por bloque, de estos nombres de punto extremo. Además de ampliar la instrucción AuditEndpoint, esto implica la introducción de dos nuevos nombres de parámetros: MaxEndPointIds y NumEndPoints
- **Versiónes soportadas:** La respuesta RestartInProgress y la instrucción AuditEndpoint se han ampliado con un parámetro VersionSupported para permitir que los MGC y pasarelas determinen las versiones de protocolo que cada uno soporta.
- **Códigos de error:** Se han introducido dos nuevos códigos de error: 532 y 533.
- **Utilización de SDP:** Se ha incluido en TGCP un nuevo perfil de utilización de SDP. Un aspecto digno de mención es que el perfil y todos los ejemplos de utilización requieren específicamente una estricta observancia de SDP, independientemente de la utilidad que ofrezcan los campos incluidos. Además, se han añadido a SDP extensiones específicas de IPCablecom.
- **Respuesta provisional:** Se ha incluido en TGCP detalles adicionales y la recomendación de un mecanismo de respuesta provisional. Se ha introducido una respuesta Acuse de recibo de respuesta (000), se ha permitido un parámetro ResponseAck vacío en respuestas las finales que sigan a respuestas provisionales, y se ha especificado un procedimiento para el mecanismo.
- **Parámetros de señal:** La sintaxis de parámetro de señal se ha ampliado para que permita la utilización de paréntesis equilibrados dentro de parámetros de señal. El valor del periodo de temporización de todas las señales de temporización puede ser modificado por un parámetro de señal.
- **Lotes de eventos:** TGCP introduce un conjunto de nuevos lotes de eventos.

Por último, debe señalarse que TGCP proporciona interpretaciones del comportamiento del protocolo MGCP de base, y en algunos casos una recomendación y una aclaración adicionales de dicho comportamiento, que pueden o no reflejar el comportamiento deseado del protocolo MGCP.

## Apéndice A.VI

### Ejemplos de lotes de eventos

#### A.VI.1 Lote de servicios de operador MF FGD

Nombre de lote: MO

Los códigos del cuadro A.VI.1 suelen utilizarse para identificar eventos y señales del lote "MO" lote en el caso de circuitos troncales de servicios de operador MF unidireccionales salientes "señalización de servicios de operador". La señalización de servicios de operador MF FGC también está soportada. Este lote se utilizará para circuitos troncales de servicios de operador generales, así como también para circuitos troncales de emergencia especializados:

**Cuadro A.VI.1/J.171 – Códigos utilizados para identificar eventos y señales de lote MO**

Código	Descripción	Evento	Señal	Información adicional
ans	Contestación de llamada	P	–	
ft	Tono de facsímil	√	–	
ld	Conexión de larga duración	C	–	
mt	Tono de módem	√	–	
orbk	Señal de llamada de retorno de operador	√	–	
rbz	Marca de ocupación	P	–	
rcl	Rellamada de operador	–	BR	
rel	Liberar llamada	P	BR	
res	Reanudar llamada	–	BR	
rlc	Liberación completa	P, S	BR	
sup(<addr>, <id>)	Establecimiento de llamada	–	TO	Temporización variable
sus	Suspender llamada	–	BR	
swk	Start wink	√	–	
TDD	Tonos de dispositivos de telecomunicaciones para sordos (TDD)	√		
oc	Operación completa	√		
of	Fallo de operación	√		

Siguen las definiciones de los distintos eventos y señales:

**Contestación de llamada (ans):** La contestación de llamada se produce al tiempo de la petición ANI del sistema de soporte de operaciones (OSS, *operations support system*), esto es, la llamada puede no haber sido pasada necesariamente a un operador. Después de producirse la contestación de llamada, se establecerá el dispositivo de retención de llamada, es decir, a partir de este momento sólo el OSS puede liberar el circuito troncal.

**Tono de facsímil (ft):** Se genera el evento tono de facsímil cuando se detecta una llamada de facsímil (véase por ejemplo, la Rec. UIT-T T.30, o V.21).

**Conexión de larga duración (ld):** Se detecta "conexión de larga duración" cuando una conexión ha estado establecida durante un periodo de tiempo mayor que cierto valor dado. El valor por defecto es una hora, pero puede ser modificado por el proceso de provisión.

Este evento puede detectarse en una conexión. Cuando no se especifica una conexión, el evento es aplicable a todas las conexiones del punto extremo, cualquiera que sea el momento en que se crearon.

**Tonos de módem (mt):** Se genera el evento tono de módem cuando se detecta una llamada de módem (véase por ejemplo, la Rec. UIT-T V.8).

**Señal de llamada de retorno de operador (orbk):** Se genera este evento cuando el OSS pide que se avise a la parte llamante<sup>35</sup>.

**Marca de ocupación (rbz):** Se produce este evento cuando el OSS marca el circuito troncal. Se generará un evento liberación cuando el circuito troncal deja de estar ocupado.

**Rellamada de operador (rcl):** Se puede aplicar esta señal para invocar una rellamada de operador, por ejemplo, debida a una señal de gancho conmutador hecha por el cliente para obtener el retorno del operador.

**Liberar llamada (rel):** Esta señal puede enviarse a la pasarela de medios; sin embargo, si está establecido un dispositivo de retención de llamada, la llamada no será desconectada hasta que el OSS la libere. La pasarela de medios genera un evento "liberar llamada" cuando se considera que el OSS ha liberado el circuito troncal. En este caso, el evento puede parametrizarse con uno de los códigos de causa del cuadro A.VI.2, que indican el motivo de la liberación:

**Cuadro A.VI.2/J.171 – Códigos de motivos de liberación de llamada**

Código de causa	Motivo
0	Liberación normal
3	No hay ruta al destino
8	Preapropiación
19	Ausencia de respuesta
21	Llamada rechazada
27	Destino fuera de servicio
28	Formato de número no válido (por ejemplo, dirección incompleta)
38	Red fuera de servicio
111	Error de protocolo/señalización, no especificado (por ejemplo, expiración de periodo de temporización)

**Reanudar llamada (res):** Esta señal indica que la otra parte reanudó la llamada, es decir, descolgó.

**Liberación completa (rlc):** El punto extremo y el MGC utilizan este evento/señal para confirmar que la llamada ha sido liberada y que el circuito troncal está disponible para una nueva llamada.

<sup>35</sup> Si el teléfono de la parte llamante está colgado, normalmente se aplicará una corriente de timbre; en cambio, si está descolgado, normalmente se aplicará el tono de volver a llamar.

**Establecimiento de llamada (sup(<addr>, <id>)):** Establece una llamada al sistema de servicios de operador utilizando la información de dirección y de identificación proporcionada. La información de dirección será de la forma siguiente:

addr(MF<sub>1</sub>, MF<sub>2</sub>, ..., MF<sub>n</sub>)

y la información de identificación será de la forma siguiente

id(MF<sub>1</sub>, MF<sub>2</sub>, ..., MF<sub>n</sub>)

donde cada MF<sub>i</sub> será uno de los siguientes símbolos de dígito MF en el cuadro A.VI.3:

**Cuadro A.VI.3/J.171 – Símbolos de dígito MF**

Símbolo	Dígito MF	Símbolo	Dígito MF
0	MF 0	K0	MF K0 or KP
1	MF 1	K1	MF K1
2	MF 2	K2	MF K2
3	MF 3	S0	MF S0 or ST
4	MF 4	S1	MF S1
5	MF 5	S2	MF S2
6	MF 6	S3	MF S3
7	MF 7	K0	MF K0 or KP
8	MF 8		
9	MF 9		

Por tanto, un ejemplo de señal de establecimiento de llamada podría ser:

sup(addr(K0, 5,5,5,1,2,1,2, SO), id(K0, 5,5,5,1,2,3,4, SO))

**Suspender llamada (sus):** Esta señal indica que la otra parte ha suspendido la llamada, es decir, colgó.

**Start Wink (swk):** Un controlador de pasarela de medios puede pedir a la pasarela de medios que le notifique cuándo se produce esta señal.

**Tonos de dispositivos de telecomunicaciones para sordos (TDD):** El evento TDD se genera cuando se detecta una llamada TDD (véase por ejemplo, la Rec. UIT-T V.18).

**Operación completa (oc):** El evento de operación completa se genera cuando se ha pedido a la pasarela que aplique una o más señales de tipo TO en el punto de extremo, y una o más de estas señales fueron completadas sin que hubieran sido detenidas por la detección de un evento solicitado, como la transición de descolgado o la marcación de una cifra. En el informe de finalización se puede indicar como un parámetro el nombre de la señal que llegó hasta el final de su ciclo, por ejemplo:

O: MO/oc(MO/sup)

Si se solicita el evento operación completa, no puede parametrarse con ningún parámetro de evento. Cuando no se incluye el nombre del paquete, se supone que es el nombre de paquete implícito.

**Fallo de operación (of):** En general, el evento fallo de operación puede ser generado cuando se ha solicitado al punto de extremo que aplique una o más señales de tipo TO en el punto de extremo, y una o más de estas señales fueron interrumpidas antes de la temporización. En el informe de finalización puede incluirse como un parámetro el nombre de la señal no realizada, por ejemplo:

O: MO/of(MO/sup)



Cuando se solicita el evento fallo de operación, no pueden especificarse parámetros de evento. Cuando no se incluye el nombre del paquete, se supone que es el nombre de paquete implícito.

### A.VI.2 Lote protocolo de terminación MF

*Nombre de lote: MT*

En esta versión de la Recomendación sobre TGCP, el lote sólo puede utilizarse para verificación de línea ocupada (BLV, *busy-line verification*) e interrupción de operador (OI, *operator interrupt*) en circuitos troncales de terminación MF unidireccionales entrantes dedicados a BLV y OI<sup>36</sup>.

Los códigos del cuadro A.VI.4 se utilizan para identificar eventos y señales del lote "MT" en el caso de circuitos troncales de terminación MF unidireccionales entrantes utilizados para BLV y OI:

**Cuadro A.VI.4/J.171 – Códigos utilizados para identificar eventos y señales del lote MT**

Código	Descripción	Evento	Señal	Información adicional
ans	Contestación de llamada	–	BR	
bz	Tono de ocupado	–	TO	Periodo de temporización = 30 segundos
hf	Señal de gancho conmutador	–	BR	
inf	Dígitos de información	√		
oc	Operación completa	√	–	
of	Fallo de operación	√	–	
oi	Interrupción de operación	√	–	
pst	Tono de señal permanente	–	TO	Periodo de temporización = infinito
rel	Liberar llamada	P	BR	
res	Reanudar llamada	–	BR	
rlc	Liberación completa	P, S	BR	
ro	Tono de volver a llamar	–	TO	Periodo de temporización = 30 segundos
sup	Establecimiento de llamada	P	–	
sus	Suspender llamada	–	BR	

Siguen las definiciones de los distintos eventos y señales:

NOTA – Para los detalles técnicos concretos de los tonos utilizados, véase la Rec. UIT-T E.180/Q.35.

**Contestación de llamada (ans):** La señal de contestación de llamada informa al punto extremo que la parte verificada ha contestado. Incluye el caso en que la parte verificada estaba ya descolgada. Cabe esperar que el punto extremo pase al OSS la supervisión de la contestación de llamada.

**Tono de ocupado (bz):** Estación ocupada.

**Señal de gancho conmutador (hf):** Esta señal indica que la parte verificada produjo una señal de gancho conmutador.

**Dígitos de información (inf (<inf-digits>):** Se utilizan en un circuito troncal MF entrante para indicar los dígitos recibidos. El valor del parámetro <inf-digits> comprende todos los dígitos acumulados hasta el delimitador de dígitos inclusive, es decir, ST, ST', ST'', o ST'''.

<sup>36</sup> Obsérvese que cuando los servicios de operador son proporcionados por un proveedor off-net, es posible que el OSS no tenga acceso a bases de datos de abonados para determinar si se debe o no permitir BLV y OI.

El valor de <inf-digits> es una lista de dígitos MF separados por una coma:

MF<sub>1</sub>, MF<sub>2</sub>, ..., MF<sub>n</sub>

donde cada MF<sub>1</sub> será uno de los siguientes símbolos de dígito MF en el cuadro A.VI.5:

**Cuadro A.VI.5/J.171 – Símbolos de dígito MF**

Símbolo	Dígito MF	Símbolo	Dígito MF
0	MF 0	K0	MF K0 or KP
1	MF 1	K1	MF K1
2	MF 2	K2	MF K2
3	MF 3	S0	MF S0 or ST
4	MF 4	S1	MF S1
5	MF 5	S2	MF S2
6	MF 6	S3	MF S3
7	MF 7	K0	MF K0 or KP
8	MF 8		
9	MF 9		

Por tanto, un ejemplo de señal o evento podría ser de la forma siguiente:

inf(k0, 5,5,5,1,2,3,4, s0)

Un ejemplo en el que el temporizador entre dígitos expiró después de 5,5,5 podría ser de la forma siguiente:

inf(k0, 5,5,5)

**Operación completa (oc):** Véase la definición de "operación completa" en el lote circuito troncal de PU-RDSI.

**Fallo de operación (of):** Véase la definición de "fallo de operación" en el lote circuito troncal de PU-RDSI.

**Interrupción de operador (oi):** El evento interrupción de operador se produce cuando el operador intenta interrumpir la llamada y genera el tono "interrupción de operador". Como no hay un tono normalizado definido para este evento, en el presente apéndice se define que el evento se produce cuando en el circuito troncal se detecta un nivel de energía que corresponde a una transición de ruido de línea a voz o tonos. Debe señalarse que, según esto, no es posible detectar una transición inversa, es decir, de voz o tonos a ruido de línea.

**Tono de señal permanente (pst): Liberar llamada (rel):** El MGC puede utilizar la señal de liberación para liberar la llamada<sup>37</sup>. En este caso, la señal de liberación no puede estar parametrizada.

A su vez, el punto extremo puede utilizar el evento para informar al MGC que ha liberado la llamada; en este caso, el evento puede estar parametrizado con uno de los códigos de causa del cuadro A.VI.6, que indican el motivo de la liberación:

<sup>37</sup> Obsérvese que el operador que efectúa la verificación normalmente controla la liberación de conexión completadas, no de conexiones para pruebas, y que en este caso lo usual es que se utilice la señal de suspensión.

### Cuadro A.VI.6/J.171 – Códigos de motivos de liberación de llamada

Código de causa	Motivo
0	Liberación normal
3	No hay ruta al destino
8	Preapropiación
19	Ausencia de respuesta
21	Llamada rechazada
27	Destino fuera de servicio
28	Formato de número no válido (por ejemplo, dirección incompleta)
38	Red fuera de servicio
111	Error de protocolo/señalización, no especificado (por ejemplo, expiración de periodo de temporización)

**Reanudar llamada (res):** Esta señal indica que la parte verificada reanudó la llamada, es decir, descolgó.

**Liberación completa (rlc):** El punto extremo y el MGC utilizan el evento/señal liberación completa para confirmar que la llamada ha sido liberada y que el circuito troncal está disponible para una nueva llamada.

**Tono de volver a llamar (ro) Establecimiento de llamada (sup):** Se utiliza un evento "sup" para indicar cuándo llega una llamada entrante (lo que corresponde al evento descolgar entrante). El evento se proporciona sin parámetros.

**Suspender llamada (sus):** Esta señal indica que la parte verificada suspendió la llamada, es decir, colgó.

## Apéndice A.VII

### Bibliografía

- *Bellcore Notes on the Networks*, Bellcore, SR-2275.
- *Compatibility Information for Feature Group D Switched Access Service*, Bellcore, TR-NPL-000258, Issue 1, octubre 1985.
- *Interoffice LATA Switching Systems Generic Requirements (LSSGR): Verification Connections (25-05-0903)*, Bellcore, TR-TSY-000531, Issue 2, julio 1987.
- *Signalling for Analog Interfaces*, Bellcore, LSSGR GR-506-CORE, Issue 1, junio 1996.
- *Switching System Generic Requirements for Call Control Using the Integrated Services Digital Network User Part (ISDNUP)*, Bellcore, LSSGR GR-317-CORE, Issue 2, diciembre 1997.
- *Custom Call Handling Features (FSD 80 Series)*, Bellcore, OSSGR GR-1176-CORE, Issue 1, marzo 1999.
- IETF RFC 1827 (1995), *IP Encapsulating Security Payload (ESP)*.
- IETF RFC 2974 (Experimental), *Session Announcement Protocol*.
- *RTP Parameters*, <http://www.iana.org/assignments/rtp-parameters>.

### Anexo A.B

#### Perfil 2 de TGCP

Necesita más estudio, basado en la Rec. UIT-T H.248.



## 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
<b>Serie J</b>	<b>Redes de cable y transmisión de programas radiofónicos y televisivos, y de otras señales multimedia</b>
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