UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T H.225.0

(09/99)

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

SERIE H: SISTEMAS AUDIOVISUALES Y MULTIMEDIOS

Infraestructura de los servicios audiovisuales – Multiplexación y sincronización en transmisión

---

Protocolos de señalización de llamada y paquetización de trenes de medios para sistemas de comunicación multimedios por paquetes

Recomendación UIT-T H.225.0

(Anteriormente Recomendación del CCITT)

RECOMENDACIONES UIT-T DE LA SERIE H

SISTEMAS AUDIOVISUALES Y MULTIMEDIOS

Características de los canales de transmisión para usos distintos de los telefónicos	H.10–H.19
Utilización de circuitos de tipo telefónico para telegrafía armónica	H.20–H.29
Utilización de circuitos o cables telefónicos para transmisiones telegráficas de diversos tipos o transmisiones simultáneas	H.30–H.39
Utilización de circuitos de tipo telefónico para telegrafía facsímil	H.40–H.49
Características de las señales de datos	H.50–H.99
CARACTERÍSTICAS DE LOS SISTEMAS VIDEOTELEFÓNICOS	H.100–H.199
INFRAESTRUCTURA DE LOS SERVICIOS AUDIOVISUALES	
Generalidades	H.200–H.219
Multiplexación y sincronización en transmisión	**H.220–H.229**
Aspectos de los sistemas	H.230–H.239
Procedimientos de comunicación	H.240–H.259
Codificación de imágenes vídeo en movimiento	H.260–H.279
Aspectos relacionados con los sistemas	H.280–H.299
Sistemas y equipos terminales para los servicios audiovisuales	H.300–H.399
Servicios suplementarios para multimedios	H.450–H.499

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

RECOMENDACIÓN UIT-T H.225.0

PROTOCOLOS DE SEÑALIZACIÓN DE LLAMADA Y PAQUETIZACIÓN DE TRENES DE MEDIOS PARA SISTEMAS DE COMUNICACIÓN MULTIMEDIOS POR PAQUETES

Resumen

Esta Recomendación trata los requisitos técnicos de los servicios videotelefónicos de banda estrecha definidos en las Recomendaciones de la serie H.200/AV.120, en aquellas situaciones en las que el trayecto de transmisión incluye una o más redes de paquetes, cada una de las cuales está configurada y gestionada para ofrecer una calidad de servicio (QOS) no garantizada que no es equivalente a la de la RDSI de banda estrecha, de manera que los mecanismos de protección o recuperación adicionales que van más allá de los que dispone la Recomendación H.320 han de proporcionarse en los terminales. Hay que señalar que la Recomendación H.322 trata el tema de la utilización de algunas otras LAN que pueden proporcionar las prestaciones exigibles no asumidas por las Recomendaciones H.323/H.225.0.

Esta Recomendación describe cómo puede gestionarse la información de audio, vídeo, datos y control en una red de paquetes de calidad de servicio no garantizada para proporcionar servicios conversacionales en equipos conformes con la Recomendación H.323.

Orígenes

La Recomendación UIT-T H.225.0, ha sido revisada por la Comisión de Estudio 16 (1997-2000) del UIT-T y fue aprobada por el procedimiento de la Resolución N.° 1 de la CMNT el 30 de septiembre de 1999.

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 Conferencia Mundial de Normalización de las Telecomunicaciones (CMNT), 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 N.° 1 de la CMNT.

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 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 2000

Es propiedad. Ninguna parte de esta publicación puede reproducirse o utilizarse, de ninguna forma o por ningún medio, sea éste electrónico o mecánico, de fotocopia o de microfilm, sin previa autorización escrita por parte de la UIT.

ÍNDICE

Página

Página

Recomendación H.225.0

PROTOCOLOS DE SEÑALIZACIÓN DE LLAMADA Y PAQUETIZACIÓN DE TRENES DE MEDIOS PARA SISTEMAS DE COMUNICACIÓN MULTIMEDIOS POR PAQUETES

(revisada en 1999)

El UIT-T,

considerando

la extendida adopción y el creciente uso de la Recomendación H.320 para los servicios de videotelefonía y de videoconferencia por redes conformes con las características de la RDSI de banda estrecha especificadas en las Recomendaciones de la serie I,

reconociendo

la conveniencia y ventajas de permitir el transporte de los servicios indicados, total o parcialmente, por redes de área local, pero manteniendo también la capacidad de interfuncionamiento con terminales H.320,

y advirtiendo

las características y prestaciones de los muchos tipos de red de área local que son de interés potencial,

recomienda

que se utilicen sistemas y equipos que cumplen los requisitos de las Recomendaciones H.322 o H.323 para proporcionar estas facilidades.

1 Alcance

Esta Recomendación describe los métodos por los que se asocian, codifican y paquetizan las señales de audio, vídeo, datos y control para su transporte entre equipos H.323 por una red de paquetes. Esto incluye la utilización de una pasarela H.323, que a su vez puede conectarse a terminales H.320, H.324 o H.310/H.321 por la RDSI de banda estrecha, RTPC o RDSI de banda ancha, respectivamente. Las descripciones de equipos, y los procedimientos se describen en la Recomendación H.323, mientras que la presente Recomendación trata los protocolos y formatos de mensaje. Es también posible la comunicación a través de una pasarela H.323 hacia una pasarela H.322 para las LAN de calidad de servicio (QOS, *quality of service*) garantizada, y por tanto a puntos extremos H.322.

La presente Recomendación está destinada a operar con una amplia variedad de redes de paquetes diferentes, inclusive IEEE 802.3, Token Ring, etc. De este modo, la presente Recomendación se define como algo que está por encima de la capa de transporte tal como TCP/IP/UDP, SPX/IPX, etc. En el apéndice IV se incluyen perfiles específicos para determinadas sucesiones de protocolos de transporte. ***Así, el alcance de la comunicación H.225.0 se halla entre entidades H.323 en la misma red de paquetes, utilizando el mismo protocolo de transporte***. Esta red de paquetes puede ser un único segmento o anillo, o podría lógicamente ser una red de datos empresarial que comprenda múltiples redes de paquetes puenteadas o encaminadas para crear una red interconectada. Debe destacarse que el funcionamiento de los terminales H.323 en Internet completa, o incluso varias redes de paquetes conectadas, pueden dar lugar a prestaciones mediocres. El posible medio por el que la calidad de servicio podría ser asegurada en esta red de paquetes, o en Internet en general cae fuera del alcance de esta Recomendación. Sin embargo, esta Recomendación proporciona un medio

al usuario de equipo H.323 de determinar que los problemas de calidad son resultado de la congestión de las redes de paquetes, así como procedimientos para acciones correctivas. Se señala también que el uso de múltiples pasarelas H.323 conectadas por la red RDSI pública es un método directo para aumentar la calidad de servicio.

La Recomendación H.323 y está Recomendación están destinadas a extender las conferencias/conexiones H.320 y H.221 al entorno de la red de paquetes con QOS no garantizada. Como tal, el modelo de conferencia primario[1] es un modelo de tamaño comprendido entre algunos participantes y algunos miles, a diferencia de las operaciones de difusión en gran escala, con riguroso control de admisión y estricto control de la conferencia.

Esta Recomendación hace uso del protocolo de transporte en tiempo real/protocolo de control en tiempo real (RTP/RTCP, *real-time transport protocol/real-time transport control protocol*) para la paquetización y sincronización de medios de todas las redes de paquetes subyacentes (véanse los anexos A, B y C). Adviértase que la utilización de RTP/RTCP especificada en esta Recomendación no está vinculada en modo alguno a la utilización de TCP/IP/UDP. La presente Recomendación supone un modelo de llamada en el que se utiliza señalización inicial en una dirección de transporte no RTP para establecimiento de llamadas y negociación de capacidad (véanse las Recomendaciones H.323 y H.245), seguida por el establecimiento de una o más conexiones RTP/RTCP. Esta Recomendación contiene detalles de la utilización de RTP/RTCP.

En la Recomendación H.221, las señales de audio, vídeo, datos y control se multiplexan en una o más llamadas RCC físicas sincronizadas. En el lado red de paquetes de una llamada H.323, no se aplica ninguno de estos conceptos. No hay necesidad de trasladar desde el lado RCC el concepto H.221 de una llamada P*64 kbit/s, por ejemplo 2 por 64 kbit/s, 3 por 64 kbit/s, etc. Así, en el lado red de paquetes, por ejemplo hay llamadas de una sola "conexión" con una velocidad máxima limitada a 128 kbit/s, y no llamadas a velocidad fija 2*64 kbit/s. Otro ejemplo tiene llamadas red de paquetes de una sola "conexión" con una velocidad máxima limitada a 384 kbit/s interfuncionando con 6*64 kbit/s en el lado WAN[2]. La principal justificación de este planteamiento es añadir complejidad en la pasarela y no en el terminal y evitar extenderse a las características de la red de paquetes de la Recomendación H.320 que están estrechamente vinculadas a la RDSI, a menos que sea necesario.

En general, los terminales H.323 no conocen directamente la velocidad de transferencia H.320, aunque interfuncionan a través de una pasarela H.323; en su lugar, la pasarela utiliza mensajes **FlowControlCommand** H.245 para limitar la velocidad de los medios en cada canal lógico en uso a la permitida por el múltiplex H.221. La pasarela puede permitir que las velocidades de vídeo lado red de paquetes estén substancialmente por debajo de las velocidades del lado WAN (o al contrario) mediante la utilización de una función reductora de velocidad y tramas de relleno H.261; los detalles de dichas operaciones caen fuera del alcance de la Recomendación H.323 y esta Recomendación. Adviértase que el terminal H.323 está indirectamente al corriente de las velocidades de transferencia

---

[1] Hay en estudio un modelo opcional de difusión sólo conferencia; necesariamente el modelo de difusión no permite un riguroso control de admisiones ni un estricto control de conferencia.

[2] Adviértase que las velocidades de vídeo y de datos en el lado LAN deben concordar con las velocidades de vídeo y de datos del lado RCC del múltiplex H.320; las velocidades de audio y de control no necesitan concordar. Dicho de otro modo, esperaríamos normalmente que, utilizando control de flujo H.245, la pasarela LAN/RCC obligará a las velocidades de vídeo y de datos a encajar en el múltiplex RCC de la Recomendación H.221. Sin embargo, dado que el audio puede transcodificarse a menudo en la pasarela, encontraremos frecuentemente que la velocidad de audio LAN y la velocidad RCC no concuerdan. Además, no habría ninguna esperanza de que la velocidad binaria H.221 para control (800 bit/s) concuerde en general con la velocidad binaria H.245 en el lado LAN. Adviértase también que la velocidad LAN puede quedar por debajo de la velocidad RCC para vídeo o/y datos, pero no puede rebasar la cantidad máxima que encaja en el múltiplex del lado RCC.

H.320 por medio de los campos vídeo de máxima velocidad binaria de la Recomendación H.245, y no deberá transmitir a velocidades que excedan de éstas.

Esta Recomendación está concebida de manera que, con una pasarela H.323, es posible la interoperabilidad con terminales H.320 (1990), H.320 (1993) y H.320 (1996). Sin embargo, algunas características de esta Recomendación pueden orientarse a permitir operaciones mejoradas con futuras versiones de la Recomendación H.320. Es también posible que la calidad de servicio en el lado H.320 pueda variar en base a las características y capacidades de la pasarela H.323 (véase la figura 1).

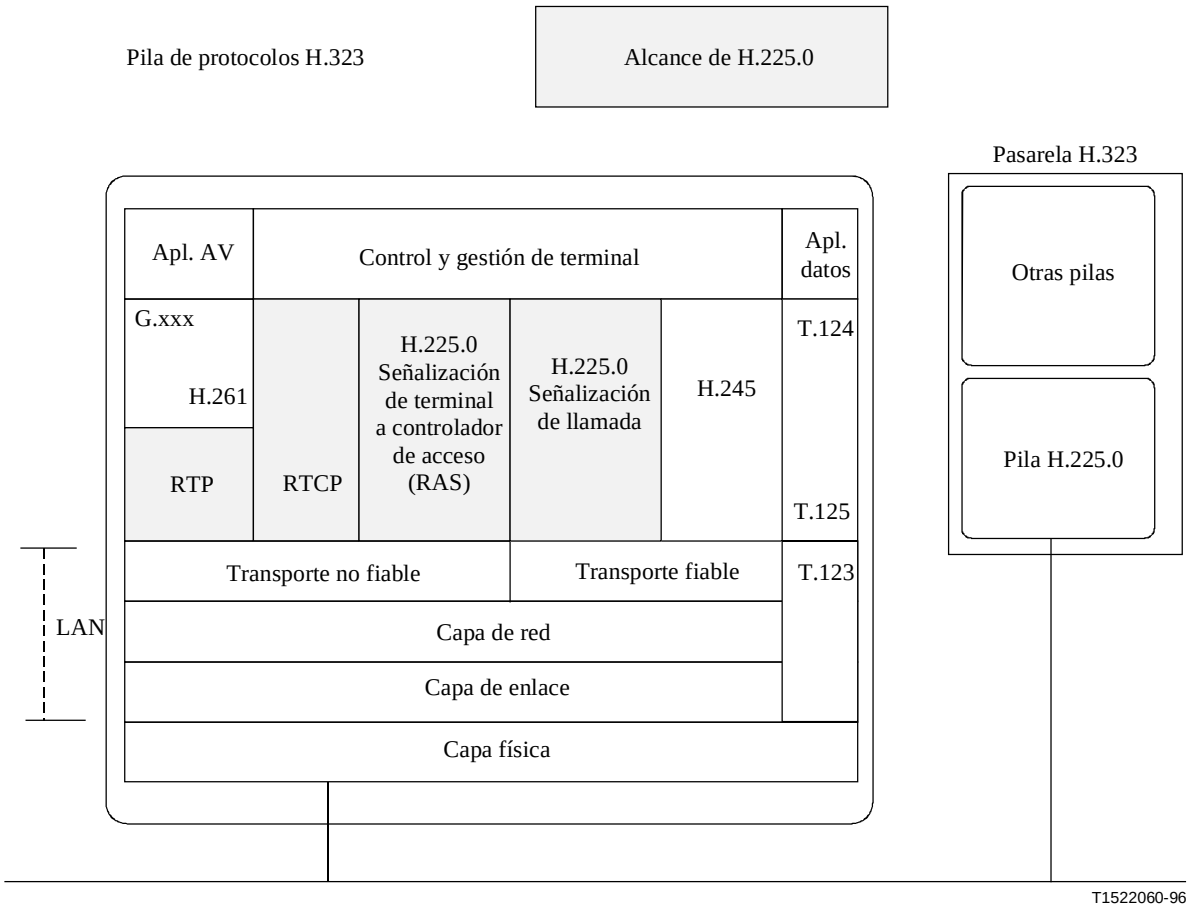


Figura 1/H.225.0 – Alcance de la Recomendación H.225.0

El planteamiento general de esta Recomendación consiste en proporcionar un medio de sincronizar paquetes que haga uso de las facilidades de la red de paquetes/de transporte subyacentes. Esta Recomendación no exige que todos los medios y el control se mezclen en un solo tren, que es luego paquetizado. Los mecanismos de trama de la Recomendación H.221 no se utilizan por las siguientes razones:

- No utilizar H.221 permite a cada medio recibir diferente tratamiento de errores, si así conviene.
- H.221 es relativamente sensible a la pérdida de grupos aleatorios de bits; la paquetización permite mayor solidez en el entorno de la red de paquetes.
- H.245 y Q.931 pueden enviarse por enlaces fiables proporcionados por la red de paquetes.
- La flexibilidad y la potencia de H.245 comparada con H.242.

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.

[1] Recomendación CCITT G.711 (1988), *Modulación por impulsos codificados (MIC) de frecuencias vocales*.

[2] Recomendación CCITT G.722 (1988), *Codificación de audio de 7 kHz dentro de 64 kbit/s*.

[3] Recomendación CCITT G.728 (1992), *Codificadores locales: Codificación de señales vocales a 16 kbit/s utilizando predicción lineal con excitación por código de bajo retardo*.

[4] Recomendación UIT-T G.723.1 (1996), *Codificadores vocales: Codificador de voz de doble velocidad para transmisión en comunicaciones multimedios a 5,3 y 6,3 kbit/s*.

[5] Recomendación UIT-T G.729 (1996), *Codificación de la voz a 8 kbit/s mediante predicción lineal con excitación por código algebraico de estructura conjugada*.

[6] Recomendación UIT-T H.221 (1997), *Estructura de trama para un canal de 64 a 1920 kbit/s en teleservicios audiovisuales*.

[7] Recomendación UIT-T H.230 (1997), *Señales de control e indicación con sincronismo de trama para sistemas audiovisuales*.

[8] Recomendación UIT-T H.233 (1995), *Sistemas con confidencialidad para servicios audiovisuales*.

[9] Recomendación UIT-T H.242 (1997), *Sistema para el establecimiento de comunicaciones entre terminales audiovisuales con utilización de canales digitales de hasta 2 Mbit/s*.

[10] Recomendación UIT-T H.243 (1997), *Procedimientos para el establecimiento de comunicaciones entre tres o más terminales audiovisuales con utilización de canales digitales de hasta 1920 kbit/s*.

[11] Recomendación UIT-T H.245 (1998), *Protocolo de control para comunicaciones multimedios*.

[12] Recomendación UIT-T H.261 (1993), *Códec vídeo para servicios audiovisuales a* $p \times 64$ *kbit/s*.

[13] Recomendación UIT-T H.263 (1998), *Codificación de vídeo para comunicación a baja velocidad binaria*.

[14] Recomendación UIT-T H.320 (1997), *Sistemas y equipos terminales videotelefónicos de banda estrecha*.

[15] Recomendación UIT-T T.122 (1998), *Servicio de comunicación multipunto – Definición de los servicios*.

[16] Recomendación UIT-T T.123 (1996), *Pilas de protocolos de datos específicos de la red para conferencias multimedios*.

[17] Recomendación UIT-T T.125 (1998), *Especificación de protocolo del servicio de comunicación multipunto*.

[18] Recomendación UIT-T H.321 (1998), *Adaptación de los terminales videotelefónicos H.320 a entornos de la red digital de servicios integrados de banda ancha (RDSI-BA)*.

[19] Recomendación UIT-T H.322 (1996), *Sistemas y equipos terminales videotelefónicos para redes de área local que proporcionan una calidad de servicio garantizada*.

[20] Recomendación UIT-T H.324 (1998), *Terminal para comunicación multimedios a baja velocidad binaria*.

[21] Recomendación UIT-T H.310 (1996), *Sistemas y terminales para comunicaciones audiovisuales de banda ancha*.

[22] Recomendación UIT-T Q.931 (1998), *Especificación de la capa 3 de la interfaz usuario-red de la red digital de servicios integrados para el control de la llamada básica*.

[23] Recomendación UIT-T Q.932 (1998), *Sistema de señalización de abonado digital N.° 1 – Procedimientos genéricos para el control de los servicios suplementarios de RDSI*.

[24] Recomendación UIT-T X.680 (1994), *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica*.

[25] Recomendación UIT-T X.680/enm.1 (1995), *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica – Enmienda 1: Reglas de extensibilidad*.

[26] Recomendación UIT-T X.681/enm.1 (1995), *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de objetos de información – Enmienda 1: Reglas de extensibilidad*.

[27] Recomendación UIT-T X.691 (1995), *Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno – Especificación de las reglas de codificación compactada*.

[28] Recomendación UIT-T E.164 (1997), *Plan internacional de numeración de telecomunicaciones públicas*.

[29] ISO/CEI 10646-1:1993, *Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane*.

[30] Recomendación UIT-T Q.850 (1998), *Utilización de los elementos de información causa y ubicación en el sistema de señalización de abonado digital N.° 1 y en la parte usuario de RDSI del sistema de señalización N.° 7*.

[31] Recomendación UIT-T Q.950 (1997), *Protocolos de servicios suplementarios, estructura y principios generales*.

[32] Recomendación UIT-T H.235 (1998), *Seguridad y criptado para terminales multimedios de la serie H (basados en las Recomendaciones H.323 y H.245)*.

3 Definiciones

Véanse las definiciones de la Recomendación H.323. En la Recomendación H.323, el término "punto extremo" se utiliza para referirse a los terminales, pasarelas y unidades de control multipunto como elementos capaces de recibir o iniciar llamadas. En la presente Recomendación, el término terminal se utiliza a menudo de manera general en descripciones de establecimiento de la llamada y debe entenderse que se refiere a un elemento que puede tomar parte en el establecimiento de la llamada, incluida una pasarela o unidad de control multipunto.

4 Abreviaturas

En esta Recomendación se utilizan las siguientes siglas.

4.1 Abreviaturas generales

BAS Señal de asignación de velocidad binaria (*bit rate allocation signal*)

CIF Formato intermedio común (*common intermediate format*)

CRV Valor de referencia de llamada (*call reference value*)

ECS Señal de control de criptación (*encryption control signal*)

GOB Grupo de bloques (*group of blocks*)

H-MLP Protocolo multicapa de alta velocidad (*high speed multi-layer protocol*)

HSD Datos de alta velocidad (*high speed data*)

IA5 Alfabeto Internacional N.° 5 (*international alphabet No. 5*)

IE Elemento de información (*information element*)

IETF Grupo de tareas especiales de ingeniería en Internet (*Internet engineering task force*)

IP Protocolo Internet (*Internet protocol*)

LAN Red de área local (*local area network*)

LD-CELP Predicción lineal con excitación por código de bajo retardo (*low delay – code excited linear prediction*)

LSB Bit menos significativo (*least significant bit*)

LSD Datos de baja velocidad (*low speed data*)

MB Macrobloque (véase la Recomendación H.261)

MBE Extensión de múltiples bytes (*multi-byte extension*)

MCC Instrucción multipunto de conferencia (*multipoint command conference*)

MCN Instrucción multipunto de negación (*multipoint command negating*)

MCS Instrucción multipunto de transmisión simétrica de datos (*multipoint command symmetrical data transmission*)

MCS Servicio de comunicación multipunto (*multipoint communication service)*

MCU Unidad de control multipunto (*multipoint control unit*)

MF Multitrama (*multiframe*)

MIC Modulación por impulsos codificados

MLP Protocolo multicapa (*multi-layer protocol*)

MPI Intervalo de imagen mínimo (*minimum picture interval*)

MSB Bit más significativo (*most significant bit*)

NA No aplicable (*not applicable*)

NS No normalizado (*non-standard*)

NSAP Punto de acceso al servicio de red (*network service access point*)

PDU Unidad de datos de protocolo (*protocol data unit*)

QCIF Cuarto de formato intermedio común (*quarter common intermediate format*)

QOS Calidad de servicio (*quality of service*)

RAS Registro, admisión y situación (*registration, admission and status*)

RCC Red con conmutación de circuitos

RTCP Protocolo de control de transporte en tiempo real (*real-time transport control protocol*)

RTP Protocolo de transporte en tiempo real (*real-time transport protocol*)

SBE Extensión de un solo byte (*single byte extension*)

SC Canal de servicio (*service channel*)

SCM Modo de comunicaciones seleccionado (*selected communications mode*)

TCP Protocolo de control de transporte (*transport control protocol*)

TSAP Punto de acceso al servicio de transporte *(transport service access point*)

UDP Protocolo de datagrama de usuario (*user datagram protocol*)

VCF Instrucción de vídeo de "petición de congelación de imagen" (*video command "freeze picture request"*)

VCU Instrucción de vídeo de "petición de actualización rápida" (*video command "fast update request"*)

4.2 Abreviaturas de mensajes RAS

ACF Confirmación de admisión (*admission confirm*)

ARJ Rechazo de admisión (*admission reject*)

ARQ Petición de admisión (*admission request*)

BCF Confirmación de ancho de banda (*bandwidth confirm*)

BRJ Rechazo de ancho de banda (*bandwidth reject*)

BRQ Petición de ancho de banda (*bandwidth request*)

DCF Confirmación de desligamiento (*disengage confirm*)

DRJ Rechazo de desligamiento (*disengage reject*)

DRQ Petición de desligamiento (*disengage request*)

GCF Confirmación de controlador de acceso (*gatekeeper confirm*)

GRJ Rechazo de controlador de acceso (*gatekeeper reject*)

GRQ Petición de controlador de acceso (*gatekeeper request*)

IACK Acuse de recibo de petición de información (*information request acknowledgement*)

INAK Acuse de recibo negativo de petición de información (*information request negative acknowledgement*)

IRQ Petición de información (*information request*)

IRR Respuesta a petición de información (*information request response*)

LCF Confirmación de localización (*location confirm*)

LRJ Rechazo de localización (*location reject*)

LRQ Petición de localización (*location request*)

RAC Confirmación de disponibilidad de recurso (*resource availability confirmation*)

RAI Indicación de disponibilidad de recurso (*resource availability indication*)

RCF Confirmación de registro (*registration confirm*)

RIP Petición en curso (*request in progress*)

RRJ Rechazo de registro (*registration reject*)

RRQ Petición de registro (*registration request*)

UCF Confirmación de desregistro (*unregistration confirm*)

URJ Rechazo de desregistro (*unregistration reject*)

URQ Petición de desregistro (*unregistration request*)

5 Convenios

En esta Recomendación los verbos en futuro indican un requisito obligatorio mientras que en condicional indican un procedimiento o característica sugeridos pero opcionales. El poder modal se refiere a un desarrollo opcional sin expresar una preferencia.

Cuando se utiliza un término tal como "MCU", se alude a un MCU H.323. Si se desea aludir a un MCU H.231, así se hará explícitamente.

En esta Recomendación, los kilobits/segundo se abrevian kbit/s y se miden en unidades de 1000. Así, 64 kbit/s es exactamente 64 000 bit/s.

A menos que se indique otra cosa, la codificación PER de la ASN.1 se utilizará para todas las ASN.1 especificadas en esta Recomendación.

Los mensajes de la Recomendación Q.931 aparecerán todos en LETRAS MAYÚSCULAS; la ASN.1 en **negritas**.

6 Mecanismo de paquetización y de sincronización

6.1 Planteamiento general

Antes de que se efectúen llamadas, un punto extremo puede descubrir/registrarse en un controlador de acceso. Si así ocurre, es conveniente que el punto extremo conozca la antigüedad del controlador de acceso en el que se registra. También es conveniente que el controlador de acceso conozca la antigüedad de los puntos extremos que se registran en él. Por estas razones, el *descubrimiento* y las secuencias de registro contienen un IDENTIFICADOR DE OBJETO de estilo H.245 que permite determinar la antigüedad en términos de la versión de la Recomendación H.323 implementada. Esta secuencia puede también contener partes opcionales de mensaje no normalizadas para permitir que los puntos extremos establezcan relaciones no normalizadas. Al final de esta secuencia, los controladores de acceso y los puntos extremos conocen mutuamente los números de versión y la situación normalizada de sus correspondientes interlocutores.

El número de versión es obligatorio y la información no normalizada es opcional en la secuencia de establecimiento/conexión descrita a continuación, para permitir a los dos puntos extremos e informarse entre sí de su antigüedad y situación no normalizada. Adviértase, sin embargo, que todos los mensajes Q.931 tienen un campo para un mensaje opcional no normalizado en el elemento de información de usuario a usuario, y que todos los mensajes de canal RAS tienen un campo opcional para información no normalizada. Además, se ha definido un mensaje RAS no normalizado que puede enviarse en cualquier momento.

El canal no fiable para registro, admisiones y mensajería de situación se denomina el canal RAS. El procedimiento general para iniciar una llamada es enviar una petición de admisión obligatoria por el

canal RAS[3], seguida por un mensaje **Establecimiento** (Setup) inicial en una dirección de transporte de canal fiable (esta dirección puede haber sido devuelta en el mensaje de confirmación o admisión o puede haberle resultado conocida al terminal llamante). De resultas de este mensaje inicial, comienza un secuencia de establecimiento de llamada sobre la base de operaciones Q.931 con las mejoras descritas más adelante. La secuencia está completa cuando el terminal recibe en el mensaje **Conexión** una dirección de transporte fiable en la cual enviar mensajes de control H.245[4].

Cuando los mensajes se envían por el canal de señalización de llamada H.225.0 fiable, se enviará únicamente un mensaje completo dentro de los límites definidos por el transporte fiable; no habrá ninguna fragmentación de mensajes H.225.0 en las PDU de transporte. (En las implementaciones IP descritas en el apéndice IV, esta PDU es definida por TPKT.)

Una vez que se ha establecido el canal de control H.245 fiable, pueden establecerse canales adicionales para audio, vídeo y datos a tenor del resultado de intercambio de capacidades utilizando procedimientos de canal lógico H.245. Además, la naturaleza de la conferencia multimedios en el lado red de paquetes (centralizada o bien distribuida/multidifusión) es negociada conexión por conexión[5]. Esta negociación se efectúa según el medio, en el sentido de que, por ejemplo, el audio/vídeo puede ser distribuido, mientras que los datos y el control son centralizados.

Cuando los mensajes se envían por el canal de control H.245 fiable, puede enviarse más de un mensaje dentro de los límites definidos por la PDU de transporte fiable mientras se envíen mensajes completos; no habrá ninguna fragmentación de mensajes H.245 en las PDU de transporte. (En las implementaciones IP descritas en el apéndice IV, esta PDU es definida por TPKT.)

Los terminales H.225.0 serán capaces de enviar audio y vídeo utilizando RTP a través de canales no fiables para minimizar el retardo. Puede aplicarse ocultación de errores u otra acción de recuperación para superar la pérdida de paquetes; en general, los paquetes de audio/vídeo no se retransmiten, pues se originaría así un retardo excesivo en el entorno de la red de paquetes[6]. Se supone que los errores de bit son detectados en las capas inferiores, y que en los paquetes con error no son enviados hasta H.225.0. Adviértase que el audio/vídeo y la señalización de llamada/control H.245 nunca se envían por el mismo canal, y no comparten una estructura de mensaje común. Los terminales H.225.0 serán capaces de enviar y recibir audio y vídeo en direcciones de transporte separadas utilizando ejemplares separados de RTP para permitir números de secuencia de trama específicos de los medios y tratamiento separado de calidad de servicio para cada medio. Sin embargo, queda para estudio ulterior un modo opcional en el que se mezclen paquetes de audio y vídeo en una sola trama que se envía a una única dirección de transporte.

Las capacidades T.120 se negocian utilizando la H.245, y al recibo de mensajes apropiados, se establecen conferencias T.120 utilizando las pilas de transporte/red de paquetes de la Recomendación T.123, si así conviene. T.120 se transportará por la red de paquetes entre puntos extremos en otra dirección de transporte. El cuadro 1 muestra el número de identificadores de TSAP utilizados para cada medio en una llamada punto a punto. Es también cierto que un determinado terminal H.323 puede conseguir participar en más de una conferencia a un tiempo, lo que da lugar al

---

[3] Un terminal que no se ha registrado en un controlador de acceso no necesita enviar una petición de admisión.

[4] Adviértase que la dirección H.245 puede enviarse en el mensaje de AVISO o LLAMADA EN CURSO para acortar el tiempo de establecimiento de llamada. Obsérvese que el canal H.245 se puede abrir inmediatamente después de la recepción de la dirección H.245 en el mensaje ESTABLECIMIENTO.

[5] La conferencia en el lado LAN puede ser en parte centralizada y en parte distribuida, según decida la MC que controla la conferencia. Sin embargo, el terminal no conoce este dato. Generalmente, por supuesto, todos los terminales verán el mismo modo de comunicaciones seleccionado (SCM, *selected communications mode*) (véase en la Recomendación H.243 una definición).

[6] La actualización rápida de tramas completas MB o GOB puede solicitarse mediante señalización H.245.

uso de identificadores de TSAP adicionales. Todos los canales lógicos H.245 son unidireccionales, excepto los asociados con T.120, que son bidireccionales.

Cuadro 1/H.225.0 – ID de TSAP utilizados por H.225.0 por llamada unidifusión punto a punto

Utilización de ID de TSAP	Fiable o no fiable	Conocidos o dinámicos
Audio/RTP	No fiable	Dinámico
Audio/RTCP	No fiable	Dinámico
Vídeo/RTP	No fiable	Dinámico
Vídeo/RTCP	No fiable	Dinámico
Señalización de llamada	Fiable	Conocido o dinámico
H.245	Fiable	Dinámico
Datos (T.120)	Fiable	Conocido o dinámico
RAS	No fiable	Conocido o dinámico
NOTA – Si se utilizan identificadores de TSAP conocidos, puede haber sólo un único punto extremo por dirección de red. Además, en el modelo de llamada directa, el llamante requiere un identificador de TSAP conocido para que el canal de señalización de llamada inicie la llamada.		

Aunque la dirección de transporte para, por ejemplo, audio y vídeo, puede compartir la misma dirección de la red de paquetes y diferir sólo en el identificador de TSAP, algunos fabricantes pueden decidir utilizar diferentes direcciones de la red de paquetes para audio y vídeo. El único requisito es que se siga el convenio de los anexos A y B en la numeración de identificadores TSAP en la sesión RTP[7].

El cuadro 1 describe el caso básico de operaciones punto a punto entre dos terminales. Para facilitar la construcción de pasarelas, MCU, y controladores de acceso, se pueden utilizar los ID de TSAP dinámicos en vez de los ID de TSAP conocidos. Los cuadros 2 y 3, ilustran un ejemplo de la utilización de los ID de TSAP en el caso de pasarela/MCU y en el caso de controladores de acceso.

Cuadro 2/H.225.0 – ID de TSAP utilizados en una MCU/un puerto de pasarela unidifusión

Utilización de ID de TSAP	Fiable o no fiable	Conocidos o dinámicos
Audio/RTP	No fiable	Dinámico
Audio/RTCP	No fiable	Dinámico
Vídeo/RTP	No fiable	Dinámico
Vídeo/RTCP	No fiable	Dinámico
Señalización de llamada	Fiable	Dinámico (nota)
H.245	Fiable	Dinámico
Datos (T.120)	Fiable	Dinámico
RAS	No fiable	Dinámico (nota)
NOTA – Véase la nota 1 al cuadro 3.		

[7] Adviértase que puede utilizarse cualquier ID de TSAP para la sesión RTP inicial; la razón principal de seguir el convenio RTP es para una posible interoperabilidad IETF RTP.

Cuadro 3/H.225.0 – Uso de ID de TSAP por un controlador de acceso H.225.0 por punto extremo que soporte el modelo de llamada por mediación de un controlador de acceso de la figura 11/H.323 para una llamada punto a punto

Utilización de ID de TSAP	Fiable o no fiable	Conocidos o dinámicos	Número de canales
Señalización de llamada	Fiable	Dinámico o conocido (nota 1)	2 por llamada (nota 2)
H.245	Fiable	Dinámico	2 por llamada (nota 2)
RAS	No fiable	Conocido	1
NOTA 1 – Si se utiliza el ID de TSAP conocido, el controlador de acceso puede limitarse a un único punto extremo por dispositivo; por tanto, deben utilizarse ID de TSAP dinámicos. NOTA 2 – 0 para modelo de llamada directa; 2 para modelo de llamada por mediación de un controlador de acceso.			

Adviértase que se utiliza una dirección de transporte fiable conocida para el establecimiento de llamada en el caso de terminal a terminal, y también para el caso de mediación de un controlador de acceso. La conexión de señalización de llamada fiable se mantendrá activa de acuerdo con las siguientes reglas:

1) Para señalización de llamada de terminal a terminal (véase la figura 9/H.323), uno u otro terminal puede decidir cerrar el canal de señalización de llamada fiable, o dejarlo abierto.

2) En el caso de señalización de llamada con mediación de un controlador de acceso (véase la figura 8/H.323), los terminales mantendrán activo el puerto fiable a lo largo de toda la llamada. Sin embargo, el controlador de acceso puede decidir cerrar el canal de señalización, pero debe mantener el canal abierto para llamadas en las que intervienen pasarelas, lo cual permite la transmisión de extremo a extremo de elementos de información Q.931 tales como información de visualización.

3) Si por algún motivo, el enlace fiable queda inactivo por un fallo a nivel de transporte u otro problema, el enlace será reabierto, y la llamada no se abandonará. El estado de llamada y el uso de CRV (valor de referencia de llamada de la Recomendación Q.931) no es afectado por el cierre del enlace fiable a menos que se cierre también el canal H.245, indicando el fin de la llamada.

Adviértase que puede haber abierto en un determinado momento más de un canal H.245, es decir, un punto extremo puede estar en más de una llamada/conferencia al mismo tiempo. Adviértase también que dentro de una determinada llamada, un terminal puede tener abierto más de un canal del mismo tipo, por ejemplo, dos canales de audio para audio estéreo. La única limitación es que habrá exclusivamente un canal de control H.245 en cada sentido por llamada punto a punto.

La señalización de canal lógico H.245 se utiliza para comenzar y detener la utilización de protocolos de vídeo, audio y datos. Este proceso exige el cierre del canal abierto, y la posterior reapertura con un nuevo modo de operación. Como parte del proceso de apertura del canal, antes de enviar el acuse de canal lógico abierto, el punto extremo utiliza la frecuencia ARQ/ACF o BRQ/BCF para asegurar que hay disponible suficiente anchura de banda para el nuevo canal (a menos que haya disponible suficiente anchura de banda de una secuencia ARQ/ACF o BRQ/BCF anterior). En algunos casos, la pasarela puede encontrar que el cambio de modo en el lado RCC se produce mucho más rápidamente que el cambio de modo en el lado red de paquetes, lo que introduce la posibilidad de pérdida de información de audio. La pasarela podría adoptar varios procedimientos a discreción del fabricante:

a) la pasarela puede transcodificar audio, ocultando así los cambios de modo en el lado RCC;

b) la pasarela puede simplemente desechar la información de audio; o

c) la pasarela puede funcionar como una MCU H.231, obteniendo así control sobre todos los cambios de modo en el lado RCC.

No existe una regla general para saber si los procedimientos H.245 o RTP (véanse los anexos A, B y C) tienen precedencia; cada conflicto y su resolución se menciona específicamente en esta Recomendación.

Obsérvese también que no hay ninguna asociación fija entre los SSRC y los canales lógicos. La Recomendación H.245 proporciona esta asociación que puede utilizar para la sincronización de audio/vídeo.

En general, son posibles dos tipos de modos de operación conferencia en el lado red de paquetes: distribuido y centralizado. Es también posible que puedan hacerse elecciones diferentes para diferentes medios, por ejemplo, audio/vídeo distribuido y datos centralizados. Los procedimientos para determinar qué clase de conferencia por establecer figuran en la Recomendación H.323; los mensajes de esta Recomendación se destinan a soportar todas las combinaciones permitidas, señalándose que el control y datos distribuidos quedan en estudio aunque son soportados por la señalización de capacidades H.245.

6.2 Utilización de RTP/RTCP

El punto extremo H.225.0 deberá poder utilizar los ID de TSAP distintos para audio y vídeo y los canales RTCP asociados descritos en los anexos A y B. Opcionalmente, los puntos extremos pueden decidir utilizar diferentes direcciones de la red de paquetes para audio y vídeo, pero para cada dirección de la red de paquetes se debe seguir el convenio de los anexos A y B en el uso de ID de TSAP. Utilizando señalización H.245 pueden establecerse canales de audio y de vídeo adicionales si el terminal soporta esta capacidad.

Sigue en estudio una capacidad opcional para utilizar una sola dirección de transporte para audio y vídeo.

A menos que se mencione específicamente aquí una excepción, las implementaciones seguirán las del RTP contenidas en el anexo A, a menos que sean modificadas por texto en esta Recomendación. Las implementaciones seguirán el perfil RTP (véase el anexo B) únicamente, como se menciona específicamente en esta Recomendación.

Los traductores y mezcladores de RTP no son elementos del sistema H.323, y toda información sobre ellos que figure en los anexos A y B deberá considerarse informativa. Se señala que tanto las pasarelas como las MCU tienen algunos aspectos de los mezcladores y de los traductores, y la información de los anexos A y B puede ser de utilidad en la implementación de pasarelas y MCU. Sin embargo, las MCU no son mezcladores, y los mezcladores no son MCU. Adviértase que las pasarelas, por ejemplo, en una llamada de red de paquetes a red de paquetes a través de la pasarela, pueden actuar como traductores.

Versión (V): Se utilizará la versión 2 del RTP.

Cuenta de CSRC (CC): El uso de la cuenta de CSRC en esta Recomendación es opcional. Cuando no se utiliza, el valor de CC será cero (0). El CSRC puede ser utilizado por las MCU para proporcionar información sobre contribuyentes a la suma de audio cuando se produce procesamiento de audio distribuido. Adviértase que no hay capacidades asociadas con la aptitud para entender la cuenta de CSRC, por lo que la MCU/MC no tiene ningún modo de conocer si y cómo el terminal de la conferencia hace uso de la información.

CNAME: En el caso más simple de una conexión punto a punto por la red de paquetes, el SSRC se utiliza para identificar una fuente de audio/vídeo desde un terminal, y los dos trenes están asociados por un CNAME suministrado por el mismo punto extremo que se especifica en el anexo A.

Cuando se utiliza RTCP, los paquetes RR o SR se enviarán periódicamente como se describe en el anexo A. Se utilizará el mensaje CNAME SDES. Otros mensajes SDES (véase el anexo A) son opcionales, pero no se utilizarán para control de conferencia o información de conferencia cuando se

utilizan funciones de control H.245 y/o T.120. La información proporcionada por las Recomendaciones H.245 y/o T.120 se considerará la información correcta.

No se dependerá del mensaje RTCP BYE para la terminación de la sesión RTP. El terminal H.323 determina cuándo es desconectada una llamada mediante los procedimientos de H.323. La única utilización obligatoria del paquete RTCP BYE es para la resolución de colisiones de SSRC.

El terminal H.323, cuando interviene en cualquier conferencia, sea punto a punto o multipunto, restringirá la velocidad binaria del canal lógico promediada en un periodo definido en la Recomendación H.245 a la señalizada en las **instrucciones de control de flujo** H.245 **(FlowControlCommands** H.245**)**, instrucciones de canal lógico H.245, y el mecanismo de control de flujo T.120.

Cuando el terminal H.323 está conectado a una pasarela H.323, la pasarela utilizará los medios de las Recomendaciones H.245 y T.120 para obligar al terminal H.323 a transmitir a una velocidad inferior o igual a las velocidades de medios del lado RCC y recibirá a una velocidad igual o superior a la velocidad RCC, con las siguientes excepciones:

- La anchura de banda de control en la red de paquetes no necesita concordar con la de la Recomendación H.221.
- La anchura de banda de audio en la red de paquetes puede concordar con la de la Recomendación H.221 en la WAN, pero con la transcodificación de pasarela, no se necesita concordancia.
- En el caso de que la pasarela esté utilizando un reductor de velocidad, el terminal H.323 del lado red de paquetes no rebasará la velocidad señalizada H.245, que probablemente será inferior a la velocidad que se envía por la WAN.

La criptación para los puntos extremos H.323 queda en estudio.

6.2.1 Audio

Antes de considerar cómo se paquetiza el audio utilizando el RTP, debemos considerar cómo se señaliza mediante H.245, y la relación de esta señalización con el RTP. En general, cuando se abre el canal de audio, se abre un canal lógico H.245. La señalización H.245 en la estructura **capacidad de audio** (**AudioCapability**) se expresa en forma de máximo número de tramas por paquete. El tamaño de trama para esta Recomendación varía con la codificación de audio en uso.

Todos los terminales H.323 que ofrecen comunicación de audio cumplirán la Recomendación G.711. Para todos los códecs de audio orientados a las tramas, los receptores señalizarán el máximo número de tramas de audio que son capaces de aceptar en un único paquete de audio. Los transmisores pueden enviar cualquier número entero de tramas de audio en cada paquete, hasta el máximo especificado por el receptor. Los transmisores no dividirán las tramas de audio a lo largo de los paquetes, y enviarán números completos de octetos en cada paquete de audio.

Los códecs basados en muestras, tales como los códecs G.711 y G.722, se considerarán orientados a las tramas, con un tamaño de trama de ocho muestras. (Para más información sobre las directrices relativas a la codificación audio basada en muestras, véase el anexo B.) Con los algoritmos de audio tales como el G.723, que utilizan más de un tamaño de trama de audio, las fronteras de trama de audio dentro de cada paquete serán señalizadas dentro de banda al canal de audio.

Con los algoritmos de audio que utilizan un tamaño de trama fijo (véanse en las Recomendaciones G.728 y G.729 el tamaño de trama utilizado por cada uno) los límites de trama de audio vendrán determinados por la relación tamaño de paquete/tamaño de trama de audio; en otras palabras, sólo se pondrán tramas de audio completas en el paquete RTP.

Tipo de cabida útil (PT, *payload type*): Sólo se utilizarán tipos de cabida útil UIT-T tales como (0)[PCMU], (8)[PCMA], (9)[G722], y (15)[G728] para los códecs del UIT-T señalizados en la

Recomendación H.245. Los tipos de cabida útil dinámica intercambiados mediante la señalización H.245 se utilizarán para cualesquiera tipos de cabida útil UIT-T no enumerados en el anexo B.

Se recomienda que si se observa una interrupción en los números de secuencia, el receptor puede repetir los sonidos recibidos más recientes de modo que la amplitud del sonido repetido caiga a silencio; pueden utilizarse otros procedimientos similares a discreción del fabricante.

Cada octeto G.711 estará alineado en un paquete RTP. El bit de signo de cada octeto G.711 corresponderá al bit más significativo del octeto en el paquete RTP (es decir, suponiendo que las muestras G.711 son tratadas como octetos en el computador central, el bit de signo será el bit más significativo del octeto definido por el formato del computador central).

Cuando se envía MIC a 48/56 kbit/s hacia la red de paquetes, la pasarela H.323 rellenará los 1 ó 2 bits extra de cada octeto de conformidad con la nota 2 del cuadro 1b/G.711, y utilizará los valores RTP para PCMA o PCMU (8 ó 0). En ley μ, el relleno consiste en un "1" en los séptimo y octavo bits. En ley A el séptimo bit será 0 y el octavo bit 1. En sentido opuesto, la pasarela H.323 truncará 64 kbit/s G.711 en el lado red de paquetes para ajustarse a la velocidad G.711 utilizada en H.320. Así, en el lado red de paquetes sólo se utilizará 64 kbit/s G.711.

Cuando se envíe 48/56 kbit/s G.722 hacia la red de paquetes, la pasarela H.323 rellenará los 1 ó 2 bits extra de cada octeto, y utilizará tipos de cabida útil RTP dinámica señalizados por la Recomendación H.245 para diferenciar entre 64 kbit/s (que utiliza PT = 9) y los casos de velocidad reducida. En el sentido opuesto, la pasarela H.323 truncará 64 kbit/s G.722 en el lado red de paquetes para ajustarse a la velocidad G.711 utilizada en H.320. Así, en el lado red de paquetes sólo se utilizará 64 kbit/s G.722.

Si es posible, el terminal H.323 debe hacer uso de la característica de supresión de silencio del RTP, especialmente cuando la conferencia es multidifusión. El terminal H.323 podrá recibir trenes RTP comprimidos de silencio. Los codificadores pueden omitir el envío de señales de audio durante periodos de silencio después de enviar una sola trama de silencio, o pueden enviar tramas de relleno de fondo de silencio si estas técnicas son especificadas por la Recomendación sobre códecs de audio en uso.

6.2.2 Mensajes de vídeo

Tipo de cabida útil (PT): Sólo se utilizarán tipos de cabida útil UIT-T tales como el que se utiliza en las Recomendaciones H.261 o H.263 para los códecs del UIT-T señalizados en la Recomendación H.245. Pueden utilizarse tipos de cabida útil dinámica en códecs que pueden ser señalizados por la Recomendación H.245 y para los cuales no se han definido formatos de paquetización.

Marcador (M): El bit marcador se debe fijar según los procedimientos descritos en el anexo A salvo en los casos en que aumente el retardo de extremo a extremo.

A fin de recuperarse de la pérdida de paquetes de vídeo, se utilizarán H.245 **VideoFastUpdatePicture**, **VideoFastUpdateMB** y **VideoFastUpdateGOB**. La utilización de los paquetes de control RTCP petición de trama completa (FIR, *full intra request*) [envíeme una trama completa] y acuse de recibo negativo (NACK, *negative acknowledgment*) [envíeme ciertos paquetes] es facultativa y se señaliza en las capacidades H.245.

En C.3.3 el método de recuperación tras error 3) puede no ser práctico si NACK no llega dentro de un periodo de trama.

H.261 está paquetizada en el lado red de paquetes como en el anexo C. Mientras se disponga de paquetes RTP suficientemente grandes, no se requiere fragmentación en las fronteras de MB por el transmisor. Sin embargo, si el terminal H.323 fragmenta paquetes H.261 en el nivel RTP, esta fragmentación ocurrirá en las fronteras MB. Todos los terminales H.323 podrán recibir paquetes fragmentados MB así como paquetes fragmentados GOB, o paquetes con una combinación de MB y

GOB. Adviértase que de no conseguir soportar la fragmentación de MB en el transmisor puede dar lugar a la pérdida de un GOB completo, y puede también rebajar la velocidad de paquetes. Los paquetes RTP utilizados no deberán rebasar el tamaño de la máxima unidad de transferencia (MTU, *maximun transfer unit*) en una determinada red de paquetes para maximizar la solidez de la operación, pero si el elemento más pequeño del esquema de codificación codificado independientemente (por ejemplo, un macrobloque) es mayor que el tamaño de la MTU, no es necesario fragmentar el paquete en unidades MTU. Los MB no se separarán a lo largo de los paquetes; todos los paquetes terminarán en una frontera de GOB o de MB. El transmisor H.323 puede decidir rellenar un paquete que contenga un pequeño GOB con MB adicionales, pero esto no es necesario.

Para excluir la posibilidad de corrupción en múltiples imágenes causadas por la pérdida de un paquete RTP, el paquetizador RTP en un punto extremo H.323 no incluirá vídeo de más de una imagen en un paquete RTP.

SBIT es el número de bits más significativos que serán ignorados en el primer octeto de datos y EBIT es el número de bits menos significativos que serán ignorados en el último octeto de datos.

El paquetizador RTP no alineará vídeo en octetos intencionalmente al principio de los paquetes RTP. En otras palabras, si EBIT = n en un paquete RTP, SBIT en el siguiente paquete RTP será igual 8 – n, 0 < n < 8, y si EBIT = 0 en un paquete RTP, SBIT en el siguiente paquete RTP será igual a 0. Este requisito evita posible retardo adicional de extremo a extremo causado por el desplazamiento de bits. Este requisito se aplicará a través de las fronteras de imagen.

El anexo D especifica una extensión H.323 del encabezamiento de paquete de vídeo que contiene una cuenta de octetos. La utilización de esta extensión facultativa se describe en el anexo D.

Véase en el apéndice IV asesoramiento específico para la red de paquetes sobre la paquetización de vídeo.

6.2.3 Mensajes de datos

No hay mensajes ni formatos de datos especiales; T.120 se utiliza en la red de paquetes como en la Recomendación T.123. La comparación entre la conferencia de datos centralizada y distribuida por la red de paquetes se describe en la Recomendación H.323, y se negocia mediante H.245.

El control de flujo T.120 en la red de paquetes es gestionado utilizando protocolos de red de paquetes cuando son solicitados por **FlowControlCommands (Instrucciones de control de flujo)** y límites **de velocidad binaria máxima (maxBitRate)** H.245.

Véanse en la Recomendación H.323 los procedimientos utilizados para conectar una conferencia T.120 en curso con una conferencia H.323, o para añadir una llamada H.323 a una conferencia T.120.

El protocolo a utilizar por H.224 en la red de paquetes seguirá en estudio.

7 Definición de mensajes H.225.0

En esta cláusula se trata la definición de los mensajes para el establecimiento de llamada, control de llamada y las comunicaciones entre terminales, pasarelas, controladores de acceso y MCU.

Las definiciones ASN.1 para todos los mensajes H.225.0 figuran en el anexo H.

7.1 Utilización de mensajes Q.931

Las implementaciones seguirán la Recomendación Q.931 como se especifica en esta Recomendación. Los terminales pueden también soportar mensajes Q.931 y H.450 opcionales. Los mensajes contendrán todos los elementos de información obligatorios y pueden contener cualquiera de los elementos de información opcionales definidos en la Recomendación Q.931 que se describen

en esta Recomendación. Adviértase que el punto extremo H.225.0 puede, según la Recomendación Q.931, ignorar todos los mensajes opcionales que no soportan sin dañar la interoperabilidad, pero responderá a un mensaje desconocido con un mensaje SITUACIÓN (STATUS).

Cada punto extremo H.225.0 será capaz de recibir e identificar un mensaje Q.931 o H.450 entrante como tal. Será capaz de procesar los mensajes Q.931 que sean de su mandato; puede ser capaz de procesar los mensajes Q.931 opcionales. En cualquier caso, cada punto extremo H.225.0 será capaz de ignorar mensajes que le resulten desconocidos sin perturbar el funcionamiento.

Cada punto extremo H.225.0 será capaz de interpretar y generar los elementos de información que sean de su mandato en lo sucesivo para los respectivos mensajes Q.931 o H.450. Podría interpretar y generar también los elementos de información opcionales definidos a continuación. Puede también interpretar otros elementos de información de Q.931 y otros protocolos de la serie Q o protocolos H.450. Los puntos extremos serán capaces de ignorar los elementos de información desconocidos contenidos en un mensaje Q.931 o H.450 sin perturbar el funcionamiento. Los procedimientos para recibir elementos de información "se requiere comprensión" no reconocidos se aplicarán conforme a 5.8.7.1/Q.931.

Los sistemas intermedios (pasarelas y controladores de acceso) seguirán las reglas siguientes en relación con los mensajes opcionales y elementos de información de Q.931:

1) La pasarela debe remitir y el controlador de acceso remitirá todos los elementos de información (opcionales u obligatorios) después de la modificación apropiada asociados con mensajes Q.931 obligatorios sea desde el terminal a la pasarela/terminal o en sentido opuesto. Esto incluye elementos de información tales como información de usuario a usuario y la información de visualización.

2) Una pasarela debe remitir todos los mensajes opcionales Q.931 o H.450 y elementos de información en ambos sentidos. Si el canal de señalización de llamada no es mantenido activo por el controlador de acceso, esto no es posible.

3) Mientras el canal de señalización de llamada Q.931 esté activo, un controlador de acceso remitirá todos los mensajes opcionales Q.931 o H.450 y elementos de información en ambos sentidos después de la modificación apropiada. Si el canal de señalización de llamada no es mantenido activo por el controlador de acceso, esto no es posible. Obsérvese que es posible que el controlador de acceso actúe como un elemento de señalización que puede proporcionar características (tales como las características de los servicios suplementarios) y, por lo tanto, modificar, terminar u originar mensajes Q.931.

Las pasarelas H.323 pueden convertir la serie H.450 de servicios suplementarios al servicio suplementario ISO/CEI 11582 correspondiente y viceversa. Las pasarelas H.323 pueden transmitir la señalización del servicio suplementario ISO/CEI 11582 sin modificaciones por el entorno H.323 dentro del elemento de información usuario a usuario. Los detalles de estos procedimientos quedan en estudio.

En esta versión de esta Recomendación, todas las referencias corresponden a la versión 1993 de la Recomendación Q.931. Se siguen los procedimientos de 3.1/Q.931 para el establecimiento de conexión en modo circuito. Sin embargo, se recuerda al implementador que aunque el "portador" está siendo señalizado al efecto, no existen "canales B" efectivos del tipo RDSI en el lado red de paquetes. La "llamada" realizada con éxito da lugar a un canal fiable de extremo a extremo que soporta la mensajería H.245. Realmente, el establecimiento del "portador" se efectúa aplicando H.245. Sin embargo, la utilización de Q.931 en el lado red de paquetes permite el interfuncionamiento con Q.931 en el lado WAN así como la provisión de un marco verificado para determinar las características generales de llamada orientadas a la conexión.

En general, se utilizan los procedimientos simétricos del anexo D/Q.931, lo cual implica que la máquina de estados Q.931 va seguida como se indica en el anexo D/Q.931 con la excepción de que

el procedimiento de D.3/Q.931 (Colisión de llamadas) no se aplicará; la recuperación tras esta condición se deja a la capa de aplicación.

Los puntos extremos que no soporten juegos de códigos Q.931 con cambio a otros juegos ignorarán todos los mensajes Q.931 que utilicen dichos métodos.

El cuadro 4 muestra qué mensajes son obligatorios y opcionales para el establecimiento de llamada H.323 y H.225.0 utilizando Q.931 en la red de paquetes.

Cuadro 4/H.225.0 – Utilización de mensajes Q.931/Q.932 en la Recomendación H.225.0

	Transmisión (M, F, O, CM)[a)]	**Recepción y acción (M, F, O[b)], CM)**
Mensajes de establecimiento de llamada		
Aviso	M	M
Llamada en curso	O	CM[c)]
Conexión	M	M
Acuse de conexión	F	F
Progresión	O	O
Establecimiento	M	M
Acuse de establecimiento	O	O
Mensajes de liberación de llamada		
Desconexión	F	F
Liberación	F	F
Liberación completa	M[d)]	M
Mensajes de la fase de información de llamada		
Reanudación	F	F
Acuse de reanudación	F	F
Rechazo de reanudación	F	F
Suspensión	F	F
Acuse de suspensión	F	F
Rechazo de suspensión	F	F
Información de usuario	O	O
Mensajes varios		
Control de congestión	F	F
Información	O	O
Notificación	O	O
Situación	M[e)]	M
Consulta de situación	O	M

Cuadro 4/H.225.0 – Utilización de mensajes Q.931/Q.932 en la Recomendación H.225.0 *(fin)*

	Transmisión (M, F, O, CM)[a)]	Recepción y acción (M, F, O[b)], CM)
Mensajes Q.932/H.450		
Facilidad	M	M
Retención	F	F
Acuse de retención	F	F
Rechazo de retención	F	F
Recuperación	F	F
Acuse de recuperación	F	F
Rechazo de recuperación	F	F

a) M: Obligatorio (*mandatory*), F: Prohibido (*forbidden*), O: Opcional, CM: Condicionalmente obligatorio (*conditionally mandatory*).
Algo es CM si se requiere una vez que se soporte una opción.

b) Obsérvese que no se enviará el mensaje SITUACIÓN en respuesta a un mensaje indicado aquí como "O". El receptor simplemente pasará por alto el mensaje si no lo soporta.

c) Los terminales que han de utilizar pasarelas recibirán y actuarán al recibir LLAMADA EN CURSO.

d) Liberación completa se necesita para cualquier situación en la que el canal de señalización de llamada fiable H.225.0 esté abierto. Si este canal no está abierto, puede utilizarse fin de sesión H.245 para terminar la conferencia.

e) El punto extremo responderá a un mensaje desconocido con un mensaje SITUACIÓN; es también obligatoria la respuesta a INDAGACIÓN DE SITUACIÓN. Sin embargo, un punto extremo no tiene que enviar INDAGACIÓN DE SITUACIÓN. Como un asunto práctico, el punto extremo debe ser capaz de comprender un mensaje SITUACIÓN recibido en respuesta a un mensaje enviado que no es conocido para el receptor.

7.2 Elementos de información Q.931 comunes

7.2.1 Elementos de información de encabezamiento

Para todos los mensajes Q.931, hay tres campos comunes que son obligatorios, además del tipo de mensaje, que se describe en esta subcláusula.

7.2.1.1 Discriminador de protocolo

Se define en 4.2/Q.931.

Se pondrá a 08H – esto identifica el mensaje como mensaje usuario-red Q.931/I.451 (codificado según la figura 4-2/Q.931). Si un controlador de acceso está actuando como una red para suministrar servicios suplementarios, puede ser adecuado utilizar otro valor. Este asunto seguirá en estudio.

7.2.1.2 Referencia de llamada

Se define en 4.3/Q.931.

Se soportará una longitud de valor de referencia de llamada de dos octetos por cualquier punto extremo H.323.

El valor de referencia de llamada se elige en el lado que origina la llamada y tiene que ser localmente exclusivo. En una comunicación posterior, el lado llamante y el lado llamado utilizarán este valor de referencia de llamada en todos los mensajes pertenecientes a esta llamada determinada.

El valor se codifica según la figura 4-5/Q.931 para un valor de referencia de llamada de dos octetos. El octeto más significativo del valor de referencia se codifica siempre en el octeto N.° 2.

Nótese que el CRV es sólo exclusivo en una determinada parte de una llamada, por ejemplo, entre los terminales, o entre un terminal y un controlador de acceso. Si un determinado terminal tiene dos llamadas en la misma conferencia, cada uno tendrá el mismo ID de conferencia, pero diferentes CRV.

La bandera de referencia de llamada se fijará de acuerdo con los procedimientos descritos en la Recomendación Q.931.

Nótese que los valores CRV enviados en mensajes RAS se ajustarán a la estructura indicada en la Recomendación Q.931. Concretamente, la bandera de referencia de llamada se incluirá como el bit más significativo del valor de referencia de llamada (CallReferenceValue). Esto limita el CRV real a la gama de 0 a 32 767, inclusive.

7.2.1.3 Tipo de mensaje

El tipo de mensaje se codifica según la figura 4-6/Q.931 utilizando los valores especificados en el cuadro 4-2/Q.931. Seguirán en estudio extensiones específicas de H.225.0.

7.2.2 Elementos de información específicos del mensaje

Las reglas de codificación generales para los elementos de información siguientes se definen en 4.5.1/Q.931 y en el cuadro 4-3/Q.931. Se seguirán estas reglas. El mecanismo de escape (véase la figura 4-8/Q.931) es opcional.

7.2.2.1 Capacidad portadora

Este elemento de información se codifica de acuerdo con la figura 4-11/Q.931 y el cuadro 4-6/Q.931. Si este elemento de información se recibe en una llamada de red de paquetes a red de paquetes puede ser ignorada por el receptor. Si este elemento de información aparece en un mensaje de establecimiento de llamada para una conexión de señalización independiente de la llamada, definida en la Recomendación H.450.1 la codificación se ajustará a 7.2/H.450.1. En todos los demás casos, se aplica lo siguiente al uso de diversos campos de este elemento de información (las referencias de números de octeto remiten a la figura 4-11/Q.931):

Capacidad de transferencia de información (octeto N.º 3)

– El bit de extensión (bit 8) se pondrá a '1'.
– La norma de codificación (bits 6, 7) se pondrá a '00' indicando 'UIT-T'.
– Capacidad de transferencia de información (bits 0-5):
    - Para llamadas originadas desde un punto extremo de RDSI, se remitirá la información indicada por la pasarela.

      NOTA – Esto permite obtener alguna información adelantada sobre la naturaleza de la conexión que ha de remitirse al punto extremo H.323, por ejemplo, voz solamente *versus* datos *versus* vídeo; esto tendría repercusión en el ancho de banda requerido así como en la aptitud/voluntad de aceptar o no la llamada.

- Las llamadas que se originan en un punto extremo H.323 utilizarán este campo para indicar su deseo de efectuar una llamada audiovisual. Por tanto, el campo se pondrá a 'información digital sin restricciones', es decir, '01000' o a 'información digital restringida' es decir "01001". Si ha de efectuarse una llamada sólo vocal, el terminal H.323 pondrá la capacidad de transferencia de información a "conversación" (es decir "00000") o a "audio a 3,1 kHz" (es decir "10000").

Bit de extensión para el octeto N.º 4 (bit 8)

– Se pondrá a '0' si la velocidad de transferencia de información se pone a 'multivelocidad'; se pondrá a '1' en otro caso.

Modo de transferencia – octeto abreviado N.º 4 (bits 6, 7)

– Especificará 'modo circuito', valor '00'.

Velocidad de transferencia de información

– Se codificará siguiendo el cuadro 4-6/Q.931, salvo que el valor '00000' (para el modo paquete) no se permite a menos que la pasarela se conecte a una red de paquetes.

Multiplicador de velocidad – octeto N.º 4.1

– Estará presente si la velocidad de transferencia de información se pone a 'multivelocidad'.

– El bit de extensión (bit 8) se pondrá a '1'.

– Los bits 1 a 7 indicarán la anchura de banda necesaria para la llamada definida a continuación (nótese que, contrariamente a la Recomendación Q.931, se permite aquí un valor de '0000001') .

– Para una llamada originada en un punto extremo de RDSI, la pasarela pasará simplemente la información que recibe de la RDSI.

– Para una llamada procedente de un punto extremo H.324, la pasarela fijará el multiplicador de velocidad a 01H.

– Para una llamada procedente de una RDSI-BA, es necesario efectuar cierta traducción de las Recomendaciones Q.2931 a Q.931. Este asunto seguirá en estudio.

– Para una llamada originada en un punto extremo H.323, éste se utilizará para indicar la anchura de banda a utilizar para esta llamada. Si el sistema llamado es otro punto extremo H.323, este valor puede reflejar la anchura de banda a utilizar en la red de paquetes, pero no es necesario que el terminal de recepción siga esta información. Si interviene una pasarela, este valor reflejará entonces el número de conexiones externas a establecer. La anchura de banda necesaria para la llamada es la anchura de banda requerida en el lado RCC y puede o no concordar con la anchura de banda permitida en la red de paquetes por los mensajes ACF/BCF.

Protocolo de capa 1 – octeto N.º 5

– El bit de extensión (bit 8) se pondrá a '1'.

– Los bits 6 y 7 indicarán el identificador de capa 1, es decir, '01'.

– Los bits 1 a 5 indicarán el protocolo de capa 1.

– Los valores permitidos son G.711 (ley A '00011' y ley μ '00010') para indicar una llamada sólo voz y H.221 y H.242 ('00101') para indicar una llamada videotelefónica H.323.

Los octetos N.º 5a, 5b, 5c, 5d no estarán presentes.

Identificador de protocolo de capa 2 – octeto N.º 6

– No estará presente.

Identificador de protocolo de capa 3 – octeto N.º 7

– No estará presente.

7.2.2.2 Identidad de la llamada

El posible uso del elemento de información identidad de llamada seguirá en estudio. Este estudio debe considerar marcación multietapas incluidas terminal → controlador de acceso → terminal y terminal → pasarela → terminal y, encaminamiento de fuente indeterminada.

7.2.2.3 Estado de la llamada

Este elemento de información se codifica según la figura 4-13/Q.931.

Octeto N.º 3 Norma de codificación (bits 8-7)

– Se pone a '00' para codificación normalizada indicando UIT-T.

Valor de estado de la llamada (octeto N.° 3, bits 1-6)

– Fijado como en el cuadro 4-8/Q.931, pero no se utilizan los valores globales de estado de la interfaz. Los valores se interpretan como estado de usuario tal como se usa en el anexo D/Q.931. Adviértase que la mayoría de los códigos enumerados no serán generados por un terminal H.323.

7.2.2.4 Número de la parte llamada

Este elemento de información se codifica según la figura 4-14/Q.931 y el cuadro 4-9/Q.931.

Octeto N.º 3 Extensión (bit 8)

– Puesto a '1'.

Tipo de número (octeto N.º 3, bits 5-7)

– Codificado según los valores y reglas del cuadro 4-9/Q.931.

Identificación del plan de numeración (octeto N.º 3, bits 1-4)

– Codificado según los valores y reglas del cuadro 4-9/Q.931. Si está puesto a '1001' (Plan de numeración privado) en una llamada originada en una red de paquetes, esto indica que:

  1) la dirección E.164 no está presente en ESTABLECIMIENTO; y
  2) la llamada se encaminará mediante una dirección de alias en la información de usuario a usuario.

"Dígitos" de número

– Cualquier número de caracteres IA5, según los formatos especificados en el plan de numeración/marcación apropiado.

7.2.2.5 Subdirección de la parte llamada

El mismo uso que en la Recomendación Q.931.

7.2.2.6 Número de la parte llamante

Este elemento de información se codifica según la figura 4-16/Q.931 y el cuadro 4-11/Q.931.

Octeto N.º 3 Extensión (bit 8)

– Puesto a '1'.

Tipo de número (octeto N.º 3, bits 5-7)

– Codificado según los valores y reglas del cuadro 4-9/Q.931.

Identificación del plan de numeración (octeto N.º 3, bits 1-4)

– Codificado según los valores y reglas del cuadro 4-9/Q.931. Si está puesto a '1001' (Plan de numeración privado) en una llamada originada en una red de paquetes, esto indica que:

  1) la dirección E.164 no está presente en ESTABLECIMIENTO; y

  2) la llamada se encaminará mediante una dirección de alias en la información de usuario a usuario.

Octeto N.º 3a

– Codificado según los valores y reglas del cuadro 4-11/Q.931.

"Dígitos" de número

– Cualquier número de caracteres IA5, según los formatos especificados en el plan de numeración/marcación apropiado.

7.2.2.7 Subdirección de la parte llamante

Se utiliza como en la Recomendación Q.931.

7.2.2.8 Causa

Si se recibe, se aplican las reglas definidas en la Recomendación Q.850. Obsérvese que Causa o **RelCompReason** no son obligatorias para LIBERACIÓN COMPLETA; el IE Causa es facultativo en cualquier otra parte. El IE Causa y el motivo de liberación completa (ReleaseCompleteReason) (una parte del mensaje Liberación completa) se excluyen mutuamente. Las pasarelas establecerán una correspondencia entre ReleaseCompleteReason y el IE Causa cuando se envíe un mensaje Liberación completa al lado conmutado del circuito desde el lado red de paquetes (véase el cuadro 5). (No se requiere la correspondencia inversa ya que las entidades de red de paquetes tienen que decodificar el IE Causa.)

Cuadro 5/H.225.0 – Correspondencia entre ReleaseCompleteReason y el IE Causa

Código ReleaseCompleteReason	Valor de causa Q.931/Q.850 correspondiente
noBandwidth	34 – Sin circuito/canal disponible
gatekeeperResources	47 – Recurso no disponible
unreachableDestination	3 – Sin ruta al destino
destinationRejection	16 – Liberación de llamada normal
invalidRevision	88 – Destino incompatible
noPermission	111 – Error de protocolo, no especificado
unreachableGatekeeper	38 – Red deteriorada
gatewayResources	42 – Congestión del equipo de conmutación
badFormatAddress	28 – Formato de número no válido
adaptiveBusy	41 – Fallo temporal
inConf	17 – Usuario ocupado
undefinedReason	31 – Normal, no especificado
facilityCallDeflection	16 – Liberación de llamada normal
securityDenied	31 – Normal, no especificado
calledPartyNotRegistered	20 – Abonado ausente
callerNotRegistered	31 – Normal, no especificado

7.2.2.9 Identificación de canal

La utilización seguirá en estudio; puede utilizarse para proporcionar realimentación en múltiples intentos de llamada.

7.2.2.10 Número conectado

Codificado conforme a 5.4.1/Q.951

7.2.2.11 Subdirección conectada

Codificada conforme a 5.4.2/Q.951

7.2.2.12 Nivel de congestión

No se utilizará.

7.2.2.13 Fecha/hora

Codificado según la figura 4-21/Q.931.

7.2.2.14 Visualización

Codificado según la figura 4-22/Q.931. La longitud máxima del elemento de información completo es 82 octetos.

7.2.2.15 Elemento de información Facilidad ampliada

Cualquier IE Facilidad ampliada utilizado para indicar una semántica sin modificaciones, tal como se define en las Recomendaciones de la serie Q.95x, se codificará de acuerdo a 8.2.4/Q.932. En este caso, las ADU de servicio se formarán de acuerdo con el ROSE [utiliza las Recomendaciones X.208 (Especificación de la ASN.1) y X.209 (Especificación de las reglas básicas de codificación de la ASN.1)] como se define en la Recomendación X.229.

7.2.2.16 Facilidad

Para señalizar la redirección de llamada específica de los procedimientos H.323 (reenvío de llamada, redireccionamiento de una llamada al MC, o reencaminamiento forzado de una llamada hacia el controlador de acceso) o, en el caso de servicios suplementarios, señalización según las Recomendaciones de la serie H.450x, se utiliza el elemento de información usuario a usuario de la facilidad. Este caso particular se indicará codificando un IE Facilidad de longitud cero, es decir, el elemento de información Facilidad constará exactamente de 2 octetos, como sigue:

- Octeto N.° 1 (identificador del elemento de información) se pondrá a '00011100' ('1C'H) para indicar el IE Facilidad.
- Octeto N.° 2 (longitud del elemento de información) se pondrá a '0' para indicar que no siguen más octetos pertenecientes a este elemento de información.

Para indicar el reenvío de llamada, el IE Facilidad estará vacío y en el **UUIE Facilidad (Facility-UUIE)** se indicará en **dirección alternativa (alternativeAddress)** o **dirección alias alternativa (alternativeAliasAddress)** el terminal al que será redirigida la llamada. En este caso, **motivo de la facilidad (facilityReason)** se fijará en **llamada reenviada (callForwarded)**.

Para ordenar a un punto extremo que llame a un punto extremo diferente porque el punto extremo llamante desea incorporarse a una conferencia y el punto extremo llamado no tiene el MC, el IE Facilidad se podría también dejar vacío. El **ID de conferencia (conferenceID)** indicará la conferencia a la que se ha de incorporar y el motivo en el **UUIE Facilidad** será **encaminar llamada a MC (routeCallToMC)**.

Además, para ordenar al punto extremo llamante que señalice al punto extremo llamado a través del controlador de acceso del punto extremo llamado, el IE Facilidad se deja vacío. El **ID de**

conferencia en el **UUIE Facilidad** indicará la conferencia a la que se ha de incorporar y el motivo en el **UUIE Facilidad** será **encaminar llamada a controlador de acceso (routeCallToGatekeeper)**.

Cualquier IE Facilidad ampliada utilizado para indicar una semántica sin modificaciones, tal como se define en las Recomendaciones de la serie Q.95x, se codificará de acuerdo a 8.2.3/Q.932. En este caso, las ADU de servicio se formarán de acuerdo con el ROSE [utiliza las Recomendaciones X.208 (Especificación de la ASN.1) y X.209 (Especificación de las reglas básicas de codificación de la ASN.1)] como se define en la Recomendación X.229.

7.2.2.17 Compatibilidad de capa alta

Queda en estudio.

7.2.2.18 Facilidad de teclado

Codificado según la figura 4-24/Q.931.

7.2.2.19 Compatibilidad de capa baja

Queda en estudio.

7.2.2.20 Más datos

No se utilizará.

7.2.2.21 Facilidades específicas de la red

No se utilizará.

7.2.2.22 Indicador de notificación

Codificado según 4.5.22/Q.931.

7.2.2.23 Indicador de progresión

Codificado según la figura 4-29/Q.931 y el cuadro 4-20/Q.931.

Este elemento de información sólo se requiere para hacer de interfaz desde un terminal H.323 a un terminal basado en la RDSI y el ATM cuando hay disponible información de llamada en curso detallada. En este caso, la pasarela remitirá esta información al terminal H.323. El sistema extremo H.323 no necesita interpretar este elemento de información.

Si este elemento de información es generado por un terminal H.323, se aplican las siguientes restricciones:

Norma de codificación (octeto N.º 3, bits 6, 7)

– Indicará 'UIT-T' ('00').

Ubicación

– Según el cuadro 4-20/Q.931.

– Los valores 'usuario' ('0000'), 'red privada que sirve al usuario local' ('0001'), y 'red privada que sirve al usuario distante' ('0101') están permitidos.

Descripción de progresión

– Según el cuadro 4-20/Q.931.

7.2.2.24 Indicador de repetición

No se utilizará.

7.2.2.25 Indicador de rearranque

No se utilizará.

7.2.2.26 Mensaje segmentado

No se utilizará. Adviértase que no hay ningún límite superior crítico al tamaño de mensaje en la Recomendación H.323 y esta Recomendación.

7.2.2.27 Envío completo

Codificado según la figura 4-33/Q.931.

No se aplican restricciones.

7.2.2.28 Señal

Codificado según la figura 4-34/Q.931 y el cuadro 4-24/Q.931.

No se aplican restricciones.

7.2.2.29 Selección de la red de tránsito

No se utilizará.

7.2.2.30 Usuario a usuario

Codificado según la figura 4-36/Q.931 y el cuadro 4-26/Q.931, con las modificaciones que aquí se efectúan.

El elemento de información usuario a usuario será utilizado por todas las entidades H.323 para transportar información relacionada con H.323. La información usuario a usuario efectiva a intercambiar solamente entre los terminales participantes está anidada en la PDU H.323-UserInformation (a la cual no se aplican restricciones).

Se aplican las siguientes restricciones:

Longitud de contenido de usuario a usuario

– Será 2 octetos en vez de 1 como se indica en la figura 4-36/Q.931.

Discriminador de protocolo

– Indicará información de usuario codificada ('00000101') X.208 y X.209 (ASN.1).

NOTA – Esto se toma de la revisión 1993 de la Recomendación Q.931, que hace referencia a las anteriores revisiones de ASN.1. Las referencias correctas a ASN.1 son la Recomendación X.680 (sintaxis) y la Recomendación X.691 (PER).

Información de usuario

– Contendrá una estructura ASN.1 que, además de la información pertinente H.323, incluya los datos de usuario efectivos, por ejemplo, como sigue. Obsérvese que la estructura ASN.1 comienza con **H323-UserInformation**. La ASN.1 se codifica utilizando la variante alineada básica de las reglas de codificación compactada especificadas en la Recomendación X.691.

– Para el campo de información de usuario, se aplican las reglas especificadas en 4.5.30/Q.931.

7.3 Detalles de un mensaje Q.931

Adviértase que las longitudes de los elementos de información especificados en los cuadros que siguen no se refieren a mensajes que son generados únicamente por terminales H.323. Se entiende que el tamaño mostrado del elemento de información usuario a usuario es el tamaño de la estructura **datos de usuario** en **H323-UserInformation** y no incluye **h323-UU-PDU**. El tamaño total de

H323-UserInformation está limitado a 65 536 octetos. Independientemente de los tamaños especificados, los mensajes remitidos desde el lado RCC pueden tener diferentes tamaños (más grandes).

Se señala también que un elemento de información especificado más abajo como obligatorio, opcional o prohibido, sólo indica si los terminales H.323 pueden o no originar dicho elemento de información.

7.3.1 Aviso (Alerting)

Este mensaje puede ser enviado por el usuario llamado para indicar que se ha iniciado el aviso del usuario llamado. En lenguaje corriente, "el teléfono está sonando".

Seguir el cuadro 3-2/Q.931 (versión 1993) con las modificaciones del cuadro 6.

Cuadro 6/H.225.0 – Aviso

Elemento de información	Situación H.225.0 (M/F/O)	Longitud en H.225.0
Discriminador de protocolo	M	1
Referencia de llamada	M	3
Tipo de mensaje	M	1
Capacidad portadora	O	5-6
Facilidad ampliada	O	8-*
Identificación de canal	En estudio	No aplicable
Facilidad	O	8-*
Indicador de progresión	O	2-4
Indicador de notificación	O	2-*
Visualización	O	2-82
Señal	O	2-3
Compatibilidad de capa alta	En estudio	No aplicable
Usuario a usuario	M	2-131

El elemento de información usuario a usuario contiene el UUIE Aviso definido en la sintaxis de mensaje H.225.0. UUIE Aviso incluye lo siguiente:

protocolIdentifier (identificador de protocolo) – Fijado según la versión de la Recomendación H.225.0 soportada.

destinationInfo (información de destino) – Contiene un **tipo de punto extremo (EndpointType)** para permitir al llamante determinar si en la llamada interviene o no una pasarela.

h245Address (dirección h245) – Es una dirección de transporte específica en la cual el punto extremo llamado o el controlador de acceso que trata la llamada desearía establecer señalización H.245. Esta dirección se enviará en los mensajes Llamada en curso, Progresión o Conexión.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

h245SecurityMode (modo de seguridad h245) – Una entidad H.323 que recibe un mensaje Establecimiento con el conjunto **capacidad de seguridad h245 (h245SecurityCapability)** responderá con **h245SecurityMode** aceptable correspondiente en Llamada en curso, Aviso, Progresión o Conexión.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

fastStart – Utilizado únicamente en el procedimiento de conexión rápida, **fastStart** soporta la señalización necesaria para abrir un canal lógico. Utiliza la estructura canal lógico abierto (OpenLogicalChannel) definida en la Recomendación H.245, pero el emisor indica los modos en que prefiere recibir y transmitir, y las direcciones de transporte en las que espera recibir trenes de medios.

multipleCalls – Si es VERDADERO indica que el emisor del mensaje puede señalizar múltiples llamadas a través de una sola conexión de señalización de llamadas.

maintainConnection – Si es VERDADERO indica que el emisor del mensaje puede soportar una conexión de señalización cuando ninguna llamada está actualmente señalizada a través de la conexión.

alertingAddress – Contiene las direcciones de alias para la parte con aviso.

presentationIndicator – Indica si se debe permitir o restringir la presentación de las direcciones alertingAddress.

screeningIndicator – Indica si el punto extremo o la red (controlador de acceso) suministró el elemento alertingAddress, y si este elemento fue verificado por un controlador de acceso.

7.3.2 Llamada en curso (Call Proceeding)

Este mensaje puede ser enviado por el usuario llamado para indicar que se ha iniciado el establecimiento de llamada solicitado y que no se aceptará ninguna información más de establecimiento de llamada. Véase el cuadro 7.

Cuadro 7/H.225.0 – Llamada en curso

Elemento de información	Situación H.225.0 (M/F/O)	Longitud en H.225.0
Discriminador de protocolo	M	1
Referencia de llamada	M	3
Tipo de mensaje	M	1
Capacidad portadora	O	5-6
Facilidad ampliada	O	8-*
Identificación de canal	En estudio	No aplicable
Facilidad	O	8-*
Indicador de progresión	O	2-4
Indicador de notificación	O	2-*
Visualización	O	2-82
Compatibilidad de capa alta	En estudio	No aplicable
Usuario a usuario	M	2-131

El elemento de información usuario a usuario contiene el UUIE Llamada en curso definido en la sintaxis de mensaje H.225.0. UUIE Llamada en curso incluye lo siguiente:

protocolIdentifier (identificador de protocolo) – Fijado según la versión de la Recomendación H.225.0 soportada.

destinationInfo (información de destino) – Contiene un **EndpointType** para permitir al llamante determinar si en la llamada interviene o no una pasarela.

h245Address (dirección h245) – Es una dirección de transporte específica en la cual el punto extremo llamado o el controlador de acceso que trata la llamada desearía establecer señalización H.245.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

h245SecurityMode (modo de seguridad h245) – Una entidad H.323 que recibe un mensaje Establecimiento con el conjunto **h245SecurityCapability** responderá con **h245SecurityMode** aceptable correspondiente en Llamada en curso, Aviso, Progresión o Conexión.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

fastStart – Utilizado únicamente en el procedimiento de conexión rápida, **fastStart** soporta la señalización necesaria para abrir un canal lógico. Utiliza la estructura OpenLogicalChannel definida en la Recomendación H.245, pero el emisor indica los modos en que prefiere recibir y transmitir, y las direcciones de transporte en las que espera recibir trenes de medios.

multipleCalls – Si es VERDADERO indica que el emisor del mensaje puede señalizar múltiples llamadas a través de una sola conexión de señalización de llamadas.

maintainConnection – Si es VERDADERO indica que el emisor del mensaje puede soportar una conexión de señalización cuando actualmente no está señalizada ninguna llamada a través de la conexión.

7.3.3 Conexión (Connect)

Este mensaje será enviado por la entidad llamada a la entidad llamante (controlador de acceso, pasarela o terminal llamante) para indicar aceptación de la llamada por la entidad llamada. Seguir el cuadro 3-4/Q.931 con las modificaciones del cuadro 8.

Cuadro 8/H.225.0 – Conexión

Elemento de información	Situación H.225.0 (M/F/O)	Longitud en H.225.0
Discriminador de protocolo	M	1
Referencia de llamada	M	3
Tipo de mensaje	M	1
Capacidad portadora	O (nota)	5-6
Facilidad ampliada	O	8-*
Identificación de canal	En estudio	No aplicable
Facilidad	O	8-*
Indicador de progresión	O	2-4
Indicador de notificación	O	2-*
Visualización	O	2-82
Fecha/hora	O	8
Compatibilidad de capa alta	En estudio	No aplicable

Cuadro 8/H.225.0 – Conexión *(fin)*

Elemento de información	Situación H.225.0 (M/F/O)	Longitud en H.225.0
Compatibilidad de capa baja	En estudio	No aplicable
Usuario a usuario	M	2-131
Número conectado	O	2-*
Subdirección conectada	O	2-23
NOTA – Capacidad portadora es obligatorio si el mensaje es entre un terminal y una pasarela.		

El elemento de información usuario a usuario contiene el UUIE Conexión definido en la sintaxis de mensaje H.225.0. UUIE Conexión incluye lo siguiente:

protocolIdentifier (identificador de protocolo) – Fijado según la versión de la Recomendación H.225.0 soportada.

h245Address (dirección h245) – Es una dirección de transporte específica en la cual el punto extremo llamado o el controlador de acceso que trata la llamada desearía establecer señalización H.245. Esta dirección se enviará si se envió antes en los mensajes Aviso, Progresión, o Llamada en curso.

destinationInfo (información de destino) – Contiene un **EndpointType** para permitir al llamante determinar si en la llamada interviene o no una pasarela.

conferenceID (ID de conferencia) – Contendrá un número exclusivo para permitir a la conferencia identificarse inequívocamente de las otras recibidas en el mensaje establecimiento.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

h245SecurityMode (modo de seguridad h245) – Una entidad H.323 que recibe un mensaje Establecimiento con el conjunto **h245SecurityCapability** responderá con **h245SecurityMode** aceptable correspondiente en Llamada en curso, Aviso, Progresión o Conexión.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

fastStart – Utilizado únicamente en el procedimiento de conexión rápida, **fastStart** soporte la señalización necesaria para abrir un canal lógico. Utiliza la estructura OpenLogicalChannel definida en la Recomendación H.245, pero el emisor indica los modos en que prefiere recibir y transmitir, y las direcciones de transporte en las que espera recibir trenes de medios.

multipleCalls – Si es VERDADERO indica que el emisor del mensaje puede señalizar múltiples llamadas a través de una sola conexión de señalización de llamadas.

maintainConnection – Si es VERDADERO indica que el emisor del mensaje puede soportar una conexión de señalización cuando no hay llamadas actualmente señalizadas sobre la conexión.

language – Indica el o los idiomas en que el usuario prefiere recibir anuncios y avisos. El campo contiene uno o más rótulos de lenguaje que satisfacen la norma RFC 1766.

connectedAddress – Contiene las direcciones de alias para la parte conectada (que responde); el número de la parte conectada se encuentra en el elemento de información (IE) número conectado.

presentationIndicator – Indica si se debe permitir o restringir la presentación del elemento connectedAddress. Si están presentes el elemento **presentationIndicator** y el indicador de

presentación del IE número conectado y están en conflicto, se utilizará el indicador de presentación del IE número conectado.

screeningIndicator – Indica si el punto extremo o la red (controlador de acceso) suministró el elemento connectedAddress, y si este elemento fue verificado por un controlador de acceso. Si el **screeningIndicator** y el indicador de verificación del IE número conectado están presentes y en conflicto, se utilizará el indicador de verificación del IE número conectado.

7.3.4 Acuse de conexión (Connect Acknowledge)

Este mensaje no será enviado.

7.3.5 Desconexión (Disconnect)

Este mensaje no será enviado por una entidad H.323.

El contenido y la semántica de un mensaje DESCONEXIÓN recibido de la red se definen en el cuadro 3-6/Q.931 y en 10.5 de ISO/CEI 11582.

7.3.6 Información

Este mensaje puede enviarse para proporcionar información adicional. Puede utilizarse para proporcionar información para el establecimiento de la llamada (por ejemplo, envío con superposición) o información diversa relacionada con la llamada. Puede utilizarse para entregar características propietarias.

Este mensaje puede ser enviado por una entidad H.323; su procesamiento al ser recibido es opcional.

Este mensaje sigue el cuadro 3-7/Q.931 con las modificaciones del cuadro 9.

Cuadro 9/H.225.0 – Información de usuario

Elemento de información	Situación H.225.0 (M/F/O)	Longitud en H.225.0
Discriminador de protocolo	M	1
Referencia de llamada	M	3
Tipo de mensaje	M	1
Envío completo	O	1
Visualización	O	2-82
Facilidad de teclado	O	2-34
Señal	O	2-3
Número de la parte llamada	O	2-35
Usuario a usuario	M	2-131

El elemento de información usuario a usuario contiene el Information-UUIE definido en la sintaxis de mensaje H.225.0. El Information-UUIE incluye lo siguiente:

protocolIdentifier (**identificador de protocolo**) – Fijado según la versión de la Recomendación H.225.0 soportada.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 que se utiliza en esta Recomendación.

7.3.7 Progresión (Progress)

Este mensaje puede ser enviado por una pasarela H.323 para indicar la progresión de una llamada en caso de interfuncionamiento con RCC. Este mensaje puede ser enviado también por un punto

extremo H.323 antes del mensaje Conexión dependiendo de la interacción del servicio suplementario.

Seguir el cuadro 3-9/Q.931 y 10.10 de ISO/CEI 11502 con las siguientes modificaciones, en el cuadro 10.

Cuadro 10/H.225.0 – Progresión

Elemento de información	Situación H.225.0 (M/F/O)	Longitud en H.225.0
Discriminador de protocolo	M	1
Referencia de llamada	M	3
Tipo de mensaje	M	1
Capacidad portadora	O (nota)	5-6
Causa	O	2-32
Facilidad ampliada	O	8-*
Identificación de canal	En estudio	No aplicable
Facilidad	O	8-*
Indicador de progresión	O	2-4
Indicador de notificación	O	2-*
Visualización	O	2-82
Compatibilidad de capa alta	En estudio	No aplicable
Usuario a usuario	M	2-131
NOTA – El elemento de información capacidad portadora es obligatorio si el mensaje circula entre un terminal y una pasarela.		

El elemento de información usuario a usuario contiene el UUIE Progresión definido en la sintaxis de mensaje H.225.0. El UUIE Progresión incluye lo siguiente:

protocolIdentifier (identificador de protocolo) – Fijado según la versión de la Recomendación H.225.0 soportada.

destinationInfo (información de destino) – Contiene un **EndpointType** para permitir al llamante determinar si en la llamada interviene o no una pasarela.

h245Address (dirección h245) – Es una dirección de transporte específica en la cual el punto extremo llamado o el controlador de acceso que trata la llamada desearía establecer señalización H.245. Esta dirección se enviará si se envió antes en los mensajes Llamada en curso, Aviso o Conexión.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

h245SecurityMode (modo de seguridad h245) – Una entidad H.323 que recibe un mensaje Establecimiento con el conjunto **h245SecurityCapability** responderá con **h245SecurityMode** aceptable correspondiente en Llamada en curso, Aviso, Progresión o Conexión.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

fastStart – Utilizado únicamente en el procedimiento de conexión rápida, **fastStart** soporta la señalización necesaria para abrir un canal lógico. Utiliza la estructura OpenLogicalChannel definida en la Recomendación H.245, pero el emisor indica los modos en que prefiere recibir y transmitir, y las direcciones de transporte en las que espera recibir trenes de medios.

multipleCalls – Si es VERDADERO indica que el emisor del mensaje puede señalizar múltiples llamadas a través de una sola conexión de señalización de llamadas.

maintainConnection – Si es VERDADERA indica que el emisor del mensaje puede soportar una conexión de señalización cuando no hay llamadas actualmente señalizadas en la conexión.

7.3.8 Liberación (Release)

Este mensaje no será enviado por una entidad H.323.

El contenido y la semántica de un mensaje LIBERACIÓN recibido de la red se definen en el cuadro 3-10/Q.931 y en 10.5 de ISO/CEI 11582.

7.3.9 Liberación completa

Este mensaje será enviado por un terminal para indicar liberación de la llamada si el canal de señalización de llamada fiable está abierto. Después, el valor de referencia de llamada (CRV, *call reference value*) está disponible para su reutilización.

La secuencia desconexión/liberación/liberación completa no se utiliza, ya que el único valor añadido es que puede agregarse un elemento de información red a usuario al mensaje liberación. Como esto no se aplica al entorno de la red de paquetes, se utiliza el método del paso único de enviar sólo liberación completa.

Seguir el cuadro 3-11/Q.931 con las modificaciones del cuadro 11.

Cuadro 11/H.225.0 – Liberación completa

Elemento de información	Situación H.225.0 (M/F/O)	Longitud en H.225.0
Discriminador de protocolo	M	1
Referencia de llamada	M	3
Tipo de mensaje	M	1
Causa	CM (nota)	1
Facilidad	O	8-*
Indicador de notificación	O	2-*
Visualización	O	2-82
Señal	O	2-3
Usuario a usuario	M	2-131
NOTA – Estará presente el IE Causa o **ReleaseCompleteReason**.		

Si este mensaje es enviado en respuesta a un mensaje Facilidad con un IE Facilidad vacío, ReleaseCompleteReason se fijará en **facilidad de reflexión de llamada (facilityCallDeflection)**.

Si este mensaje es remitido desde una RCC por una pasarela, el valor de causa se fijará como se especifica en la Recomendación Q.931.

El elemento de información usuario a usuario contiene el UUIE Liberación completa definida en la sintaxis de mensaje H.225.0. El UUIE Liberación completa incluye lo siguiente:

protocolIdentifier (identificador de protocolo) – Fijado según la versión de la Recomendación H.225.0 soportada.

reason (**motivo**) – Más información sobre por qué se liberó la llamada.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

tokens – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Tokens** criptados.

busyAddress – Contiene las direcciones de alias para la parte ocupada.

presentationIndicator – Indica si se debe permitir o restringir la presentación del elemento busyAddress.

screeningIndicator – Indica si el punto extremo o la red (controlador de acceso) suministró el elemento busyAddress, y si este elemento fue verificado por un controlador de acceso.

7.3.10 Establecimiento (Setup)

Este mensaje será enviado por una entidad H.323 llamante para indicar su deseo de establecer una conexión hacia la entidad llamada.

Seguir el cuadro 3-16/Q.931 con las modificaciones del cuadro 12.

Cuadro 12/H.225.0 – Establecimiento

Elemento de información	Situación H.225.0 (M/F/O/CM)	Longitud en H.225.0
Discriminador de protocolo	M	1
Referencia de llamada	M (nota 2)	3
Tipo de mensaje	M	1
Envío completo	O	1
Indicador de repetición	F	No aplicable
Capacidad portadora	M	5-6
Facilidad ampliada	O	8-*
Identificación de canal	En estudio	No aplicable
Facilidad	O	8-*
Indicador de progresión	F	No aplicable
Facilidades específicas de la red	F	No aplicable
Indicador de notificación	O	2-*
Visualización	O	2-82
Facilidad de teclado	O	2-34
Señal	O	2-3
Número de la parte llamante	O	2-131
Subdirección de la parte llamante	CM (nota 1)	No aplicable
Número de la parte llamada	O	2-131
Subdirección de la parte llamada	CM (nota 1)	No aplicable
Selección de red de tránsito	F	No aplicable
Indicador de recepción	F	No aplicable

Cuadro 12/H.225.0 – Establecimiento *(fin)*

Elemento de información	Situación H.225.0 (M/F/O/CM)	Longitud en H.225.0
Compatibilidad de capa baja	En estudio	No aplicable
Compatibilidad de capa alta	En estudio	No aplicable
Usuario a usuario	M	2-131
NOTA 1 – Las subdirecciones se necesitan para algunos casos de llamadas RCC; no deberían utilizarse para llamadas sólo lado red de paquetes. NOTA 2 – Si se envió previamente ARQ, el CRV utilizado aquí será el mismo.		

El elemento de información usuario a usuario contiene el UUIE Establecimiento definido en la sintaxis de mensaje H.225.0. El UUIE Establecimiento incluye lo siguiente:

protocolIdentifier (identificador de protocolo) – Fijado según la versión de la Recomendación H.225.0 soportada.

h245Address (dirección h245) – Es una dirección de transporte específica en la cual el punto extremo llamante o el controlador de acceso que trata la llamada desearía establecer la señalización H.245. Sólo debe ser proporcionada por el emisor si es capaz de tratar los procedimientos H.245 antes de recibir un mensaje CONEXIÓN por el canal de señalización de llamada.

sourceAddress (dirección de origen) – Contiene las direcciones de alias para el origen; el número E.164 del origen está en el IE número de la parte llamante Q.931. La dirección primaria será la primera.

sourceInfo (información de origen) – Contiene un **tipo de punto extremo** para que la parte llamada pueda determinar si la llamada comprende o no una pasarela.

destinationAddress (dirección de destino) – Es la dirección a la que se desea conectar el punto extremo. La dirección primaria será la primera. Cuando se llama a un punto extremo utilizando solamente una dirección E.164, esta dirección se colocará en el IE número de la parte llamante Q.931. La dirección de destino, si está disponible, será incluida en el mensaje Establecimiento por los terminales de la versión 2.

destCallSignalAddress (dirección de señalización de llamada de destino) – Necesario para informar al controlador de acceso de la dirección de transporte de señalización de llamada del terminal de destino; redundante en el caso directo de terminal a terminal. En todos los casos en que la información esté a disposición del emisor del mensaje Establecimiento, se rellenará este campo.

destExtraCallInfo (información de llamada extra de destino) – Necesario para efectuar posibles llamadas de canal adicional, es decir, para una llamada 2*64 kbit/s en el lado WAN. Sólo contendrá las direcciones E.l64, y no contendrá el número del canal inicial (véase la nota).

destExtraCRV (CRV extra de destino) – Los CRV para llamadas RCC adicionales especificados por **destExtraCallInfo** que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada utilizada en esta Recomendación. Su uso seguirá en estudio.

activeMC (MC activo) – Indica que el punto extremo llamante está bajo la influencia de un MC activo.

conferenceID (ID de conferencia) – Identificador de conferencia exclusivo.

conferenceGoal (objeto de la conferencia):

creación – Comenzar una nueva conferencia;

invitación – Invitar a una parte a una conferencia existente;

incorporación – Incorporarse a una conferencia existente;

negociación de capacidad – Negociar capacidades para una conferencia posterior menos estrictamente acoplada.

CallIndependentSupplementaryService (servicios suplementarios independientes de la llamada) – Transporte de las APDU de servicios suplementarios de una manera no relacionada con la llamada.

callServices (**servicios de llamada**) – Proporciona información sobre el soporte de protocolos opcionales de la serie Q para el controlador de acceso y el terminal llamado.

callType (tipo de llamada) – Mediante este valor, el controlador de acceso de la parte llamada puede tratar de determinar la utilización de anchura de banda 'real'. El valor por defecto es **punto a punto** para todas las llamadas; se debe reconocer que el tipo de llamada puede cambiar dinámicamente durante la llamada, y que el tipo de llamada final puede no ser conocido cuando se envía el mensaje establecimiento.

sourceCallSignalAddress (dirección de señalización de llamada de origen) – Contiene la dirección de transporte para la llamada de origen; este valor será utilizado en el mensaje ARQ por el receptor del mensaje Establecimiento. En todos los casos en que la información esté a disposición del emisor del mensaje Establecimiento, se rellenará este campo. El valor de sourceCallSignalAddress será igual al valor que utilizó en ARQ el emisor del mensaje Establecimiento, repetido en eco por el punto extremo que recibe ese mensaje en su ARQ.

remoteExtensionAddress (dirección de extensión distante) – Contiene la dirección de alias de un punto extremo llamado en los casos en que esta información es necesaria para atravesar múltiples pasarelas. En todos los casos en que la información esté a disposición del emisor del mensaje Establecimiento, se rellenará este campo.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

h245SecurityCapability (capacidad de seguridad h245) – Conjunto de capacidades que puede utilizar el emisor para asegurar el canal H.245.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

fastStart – Utilizado únicamente en el procedimiento de conexión rápida, **fastStart** soporta la señalización necesaria para abrir un canal lógico. Utiliza la estructura OpenLogicalChannel definida en la Recomendación H.245, pero el emisor indica los modos en que prefiere recibir y transmitir, y las direcciones de transporte en las que espera recibir trenes de medios.

mediaWaitForConnect (los medios esperan la conexión) – Si es VERDADERO indica que el recibiente del mensaje Establecimiento no transmitirá medios hasta el envío del mensaje Conexión.

canOverlapSend (posible superposición de emisiones) – Si es VERDADERO, indica que el emisor del mensaje Establecimiento soportará la superposición de emisiones.

endpointIdentifier – Es un identificador de punto extremo que fue asignado al terminal en el mensaje RCF. Este campo se presentará cuando se envía ESTABLECIMIENTO hacia el controlador de acceso donde se registra el punto extremo, y no estará presente cuando la señal de establecimiento se envía a otra entidad.

multipleCalls – Si es VERDADERO indica que el emisor del mensaje puede señalizar múltiples llamadas a través de una sola conexión de señalización de llamadas.

maintainConnection – Si es VERDADERO indica que el emisor del mensaje puede soportar una conexión de señalización cuando no hay llamadas actualmente señalizadas en la conexión.

ConnectionParameters – Permite la especificación de parámetros necesarios por pasarelas que proporcionan tipos de conexión múltiples y/o agregación (por ejemplo, una pasarela H.323/H.320):

– **scnConnectionType** – Proporciona información a una pasarela sobre el tipo de conexión particular utilizada para producir la llamada RCC completa. Los puntos extremos o controladores de acceso ocuparán este campo si la información es pertinente a los mismos. Si se indica la opción 'velocidad múltiple', el octeto de velocidad de transferencia de información en la capacidad portadora también deberá indicar 'velocidad múltiple' y el octeto multiplicador de velocidad indicará el número de conexiones. En todos los otros casos, si el campo scnConnectionType está presente, invalida cualquier indicación referente al tipo de conexión individual requerida en la velocidad de transferencia (octeto # 4) y en el multiplicador de velocidad (octeto # 4.1) del IE de capacidad portadora.

– **numberOfSCNConnections** – Indica el número de conexiones del tipo scnConnectionType que se añaden en conjunto para producir la llamada RCC. Este campo, cuando se multiplica por la anchura de banda de la conexión particular especificada en scnConnectionType, señala la anchura de banda para toda la llamada en la RCC. Los puntos extremos o controladores de acceso se insertarán en este campo si la información es pertinente a los mismos. Se debe señalar que si el campo scnConnectionType está fijado en desconocido, se supone entonces una unidad de 64 kbit/s de anchura de banda. Si tanto este campo como los campos scnConnectionType están presentes, la anchura de banda total indicada estará en conformidad con la anchura de banda RCC total indicada por la velocidad de transferencia (octeto # 4) y el multiplicador de velocidad (octeto # 4.1) del IE de capacidad portadora.

– **scnConnectionAggregation** – Indica cómo se añaden conjuntamente las conexiones particulares para producir la llamada RCC completa. Los puntos extremos o controladores de acceso se insertarán en este campo si la información es pertinente a los mismos. La opción por defecto que se utilizará cuando el mecanismo real de agregación es desconocido, es 'automático'. Cuando se tiene conocimiento que se utiliza la técnica de adhesión, pero se desconoce el modo preciso propiamente dicho, se utilizará la opción 'modelo aglomerado'.

language – Indica el o los idiomas en que el usuario prefiere recibir anuncios y avisos. El campo contiene uno o más rótulos de idioma que satisfacen RFC 1766.

presentationIndicator – Indica si se permite o restringe la presentación del campo sourceAddress. Si el **presentationIndicator** y el indicador de presentación del IE número de parte llamante están presentes y en conflicto, se utilizará el indicador de presentación del IE número de parte llamante.

screeningIndicator – Indica si el punto extremo o red (controlador de acceso) proporciona el campo sourceAddress, y si este campo fue verificado por un controlador de acceso. Si el **screeningIndicator** y el indicador de verificación del IE número de parte llamante están presentes y en conflicto, se utilizará el indicador de verificación del IE número de parte llamante.

NOTA – Si está presente **información de llamada suplementaria de destino (destExtraCallInfo)**, se puede suministrar un CRV para cada llamada, en **destExtraCRV**. Estos CRV se utilizarán para identificar cualquier respuesta a cada llamada lanzada. Estos procedimientos seguirán en estudio. Si el campo destExtraCRV no está presente, una pasarela agregará toda la información de llamada en una única respuesta, con el resultado de que si una llamada fracasa en el lado RCC, la llamada completa es tratada como un fallo.

7.3.11 Acuse de establecimiento (Setup Acknowledge)

Este mensaje puede ser enviado por una entidad H.323. Sin embargo, puede ser remitido desde la red a través de una pasarela. Su procesamiento al ser recibido es opcional, pero una entidad que indique canOverlapSend en el mensaje Establecimiento deberá soportar acuse de establecimiento.

El contenido y la semántica de un mensaje ACUSE DE ESTABLECIMIENTO recibido de la red se definen en el cuadro 3-16/Q.931.

7.3.12 Situación (Status)

El mensaje SITUACIÓN se utilizará para responder a un mensaje de señalización de llamada desconocido o a un mensaje CONSULTA DE SITUACIÓN (STATUS INQUIRY).

Seguir el cuadro 3-17/Q.931 con la única modificación de que el CRV tiene 2 octetos de longitud.

7.3.13 Consulta de situación (Status Inquiry)

El mensaje CONSULTA DE SITUACIÓN puede utilizarse para solicitar la situación de la llamada descrita en 8.4.2/H.323.

Seguir el cuadro 3-18/Q.931 con la única modificación de que el IE de referencia de llamada tiene 3 octetos de longitud.

7.4 Detalles de un mensaje Q.932

Los mensajes definidos a continuación se derivan de las Recomendaciones Q.932 y de las Recomendaciones de la serie H.450.x. Para más detalles véanse las citadas Recomendaciones.

7.4.1 Facilidad (Facility)

El mensaje Facilidad se utilizará para proporcionar información sobre adónde direccionar una llamada (FacilityReason = routeCallToMC), o para que un punto extremo indique que la llamada entrante debe pasar por un controlador de acceso (FacilityReason = routeCallToGatekeeper).

Para señalizar redireccionamiento de la llamada específico de los procedimientos H.323, se utiliza el elemento de información usuario a usuario del mensaje Facilidad. Este caso particular se indicará codificando un IE Facilidad de longitud cero. En este caso, el elemento de información facilidad constará exactamente de 2 octetos. Una entidad H.323 tratará adecuadamente el IE Facilidad vacío (específico de H.323) y será capaz de hacer caso omiso de los IE Facilidad que no comprenda.

El mensaje Facilidad puede utilizarse para pedir o acusar recibo de un servicio suplementario de conformidad con las Recomendaciones de la serie H.450. Por ese motivo, una o más APDU del servicio suplementario H.450 serán transportadas dentro del elemento de información usuario a usuario del mensaje Facilidad. Las APDU del servicio suplementario H.450 se codificarán según la cláusula 8/H.450.1. El elemento de información facilidad se contendrá con longitud cero. Obsérvese que un mensaje Facilidad que lleve únicamente las APDU del servicio suplementario H.450 no podría utilizar el UUIE Facilidad, pero sí en cambio la opción **cuerpo de mensaje h323 (h323-message-body)** "vacío".

Si está presente un IE Facilidad que lleva la semántica de la Recomendación Q.932 y está codificado tal como se define en las Recomendaciones Q.932 y Q.95x, constará por lo menos de 8 octetos tal como se indica en el cuadro 7-2/Q.932. La utilización de los IE Facilidad de ese tipo queda en estudio.

El mensaje facilidad puede ser utilizado por un punto extremo o un controlador de acceso para pedir al recibiente que establezca un canal H.245 entre las dos entidades (FacilityReason = starth245).

Seguir 7.1.1/Q.932 y 10.8 de ISO/CEI 11582, con las modificaciones del cuadro 13.

Cuadro 13/H.225.0 – Facilidad

Elemento de información	Situación H.225.0 (M/F/O)	Longitud H.225.0
Discriminador de protocolo	M	1
Referencia de llamada	M	3
Tipo de mensaje	M	1
Facilidad ampliada	O (nota)	8-*
Facilidad	O (nota)	2 u 8-*
Indicador de notificación	O	2-*
Visualización	O	2-82
Número de la parte llamante	F	No aplicable
Número de la parte llamada	F	No aplicable
Usuario a usuario	M	2-131
NOTA – Si se utiliza el mensaje Facilidad para llevar la señalización del servicio suplementario Q.95x, se necesita el elemento de información facilidad o bien el elemento de información facilidad ampliada. Si se utiliza el mensaje Facilidad para el control del servicio suplementario de conformidad con las Recomendaciones de la serie H.450.x, o si se utiliza el mensaje Facilidad para el reencaminamiento hacia las funciones MC/GK, se requiere el elemento de información facilidad de longitud cero.		

Codificación del elemento de información tipo de mensaje

El elemento de información tipo de mensaje del mensaje Facilidad se codificará "0110 0010".

El elemento de información usuario a usuario contiene el UUIE Facilidad definido en la sintaxis de mensaje H.225.0. El UUIE Facilidad incluye lo siguiente:

protocolIdentifier (**identificador de protocolo**) – Fijado según la versión de la Recomendación H.225.0 soportada.

alternativeAddress (dirección alternativa) – Es una dirección de transporte específica a la cual la parte llamante debe dirigir la llamada; si está presente no se necesita **dirección alias alternativa**.

alternativeAliasAddress (dirección alias alternativa) – Contiene los alias que se pueden utilizar para redireccionar la llamada; si se proporciona un alias no se necesita **dirección alternativa**.

conferenceID (ID de conferencia) – Identificador de conferencia único; no es necesario si se utiliza el campo **conferencias**.

reason (motivo) – Más información sobre el mensaje facilidad.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

destExtraCallInfo (**información de llamada suplementaria de destino**) – Necesario para efectuar posibles llamadas de canal adicional, es decir, para una llamada 2*64 kbit/s en el lado WAN. Sólo contendrá las direcciones E.l64, y no contendrá el número del canal inicial.

remoteExtensionAddress (**dirección de extensión lejano**) – Contiene la dirección de alias de un punto extremo llamado en los casos en que es necesaria esta información para atravesar múltiples pasarelas.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

conferences (conferencias) – Una o más conferencias a las que es posible incorporarse.

h245Address (dirección h245) – Dirección de transporte específica en la que el punto extremo o el controlador de acceso que envía esta facilidad desearía que el recibiente estableciera la señalización H.245.

multipleCalls – Si es VERDADERO indica que el emisor del mensaje puede señalizar múltiples llamadas a través de una sola conexión de señalización de llamadas.

maintainConnection – Si es VERDADERO indica que el emisor del mensaje puede soportar una conexión de señalización cuando no hay llamadas actualmente señalizadas a través de la conexión.

7.4.2 Notificación

Este mensaje puede ser enviado por una entidad H.323. El procesamiento en recepción es opcional.

Los contenidos y la semántica de un mensaje NOTIFICACIÓN se definen en 3.1.7/Q.931.

7.4.3 Otros mensajes

Los mensajes de control de llamada que pueden llevar los elementos de información opcionales Facilidad, Facilidad Ampliada o Indicador de Notificación se especifican en 8.3.

7.5 Valores de temporizadores Q.931

Se soportarán dos temporizadores Q.931:

- El "temporizador de establecimiento" T303 (véanse los cuadros 9-1/Q.931 y 9-2/Q.931), que define cuánto tiempo esperará el punto extremo llamante un mensaje AVISO, LLAMADA EN CURSO, CONEXIÓN, CONEXIÓN COMPLETA u otro mensaje de la entidad llamada después de que ha enviado un mensaje ESTABLECIMIENTO. Este valor de temporización será de por lo menos 4 segundos. Cabe señalar que pueden aparecer algunas aplicaciones en redes que tienen de por sí retardos más amplios (por ejemplo, compárese Internet con una red de empresa local o intranet).
- El "temporizador de establecimiento" T301 (véanse los cuadros 9-1/Q.931 y 9-2/Q.931), que define después de cuánto tiempo el punto extremo llamado responda. Este temporizador arranca cuando se recibe el mensaje AVISO y termina normalmente en CONEXIÓN o cuando el llamante termina el intento de llamada y envía LIBERACIÓN COMPLETA. Este valor de temporización será 180 segundos (3 minutos) o superior.

Adviértase que los valores del lado red de paquetes de estos temporizadores son los mismos que se utilizaron en la RCC.

Pueden soportarse otros temporizadores como parte de las Recomendaciones de la serie H.450 sobre servicios suplementarios opcionales.

7.6 Elementos comunes de mensajes H.225.0

Esta subcláusula describe estructuras ASN.1 que se utilizan en más de un mensaje RAS (registro, admisión y situación). Algunas pueden utilizarse en la parte usuario a usuario de los mensajes Q.931.

requestSeqNum (número secuencial de petición) en los mensajes se utiliza para seguir la pista a las múltiples peticiones pendientes. Junto con los mensajes de respuesta asociados (éxito o fracaso) devolverá el **requestSeqNum**. Los mensajes retransmitidos tendrán el mismo **requestSeqNum**. **RequestSeqNum** se incrementa en 1 módulo 65536.

El **protocolIdentifier (identificador de protocolo)** se incluye como parte del mensaje de descubrimiento, registro y establecimiento/conexión para permitir a las partes que intervienen determinar la antigüedad de las implementaciones que intervienen.

nonStandardParameter (parámetro no normalizado): Este parámetro es opcional en las secuencias de descubrimiento, registro y establecimiento/conexión para permitir a las partes que

intervienen determinar la situación no normalizada de los puntos extremos que intervienen. Un controlador de acceso o una pasarela no está obligado a pasar nonStandardData que no soporta ni entiende, ya que éstos podrían interferir con las operaciones.

La estructura **dirección de transporte (TransportAddress)** se destina a capturar los diversos formatos de transporte e incluye cualquier esquema específico de transporte, además de la referencia posiblemente local a un identificador de TSAP.

Las direcciones IPv4 e IPv6 se codificarán con el octeto más significativo de la dirección que es el primer octeto en la CADENA DE OCTETOS respectiva, por ejemplo, la clase dirección B IPv4 130.1.2.97 tendrá el '130' codificado en el primer octeto de la CADENA DE OCTETOS, seguido de '1' y así sucesivamente.

La dirección IPv6 a148:2:3:4:a:b:c:d tendrá 'a1' codificado en el primer octeto '48' en el segundo octeto, '00' en el tercero, '02' en el cuarto y así sucesivamente.

Una estructura **TransportAddress** del tipo **ipSourceRoute** en la cual la SECUENCIA **route** no tiene entrada se interpretará como representando la misma dirección que la del tipo **ipAddress** que contiene los mismos valores para **ip** y **port**.

Las direcciones IPX, **node, netnum** y **port** se codificarán con el octeto más significativo de cada campo como el primer octeto en la respectiva CADENA DE OCTETOS.

Adviértase que esta estructura no utiliza el lenguaje dirección de transporte = "dirección de red de paquetes más identificador TSAP" de la Recomendación H.323. En su lugar, se utilizan los términos comunes en cada dominio de transporte.

La estructura **tipo de punto extremo (EndpointType)** transmite información sobre el elemento H.323 en el extremo del enlace de señalización. El elemento H.323 podría completar uno o más elementos de mensaje **controlador de acceso**, **pasarela**, **mcu** o **terminal**. Si el elemento H.323 tiene un MC, la variable booleana **mc** podría ser entonces verdadera. La presencia del componente **set** indica que la entidad es un dispositivo del tipo punto extremo simple (SET, *simple endpoint type*) como se define en la Recomendación H.323 anexo F entre otras. Las posiciones de bits en el componente set indican que el tipo de dispositivo SET; su significado se define en el anexo F/H.323 y en otras Recomendaciones que especifican los tipos de dispositivos SET.

La estructura **información de pasarela (GatewayInfo)** contiene un elemento **protocolo**, que permite a la pasarela indicar los protocolos que soporta.

velocidades de datos soportados (dataRatesSupported) indica las velocidades de datos que para cada protocolo soporte el dispositivo. prefijos soportados (supportedPrefixes) indica los prefijos asociados con un protocolo soportado y también, en algunos casos, con las velocidades de datos.

La estructura **velocidad de datos (DataRate)** proporciona información sobre la velocidad del protocolo de pasarela. **velocidad de canal (channelRate)** es la velocidad de canal básica en cientos de bits. **multiplicador de canal (channelmultiplier)** indica el número de canales en channelRate. Por ejemplo, si una pasarela soporta una llamada 3B, channelMultiplier = 3 y channelRate = 640 para un canal de 64 kbit/s.

La estructura **identificador de vendedor (VendorIdentifier)** permite a un vendedor identificar un producto. El elemento **vendedor (vendor)** permite la identificación en términos de indicativo de país, extensión y código del fabricante. **identificador de producto (productId)** e **identificador de versión (versionId)** son cadenas de textos que pueden dar información sobre el producto.

La estructura **Dirección de alias (AliasAddress)** está destinada a capturar los diversos formatos de dirección externos que hacen referencia a una determinada ubicación de transporte en la red de paquetes. Cuando se registra una dirección E.164 con un controlador de acceso, un punto extremo utilizará solamente las cifras 0-9 en el campo **e164**.

Se considera que los mecanismos que utiliza un punto extremo para determinar el tipo de dirección dependen de la implementación. En el cuadro 14 se transcribe la representación de los diversos tipos de números en los mensajes. Obsérvese que si un punto extremo no conoce el tipo o el alcance de un número E.164, lo representará como un número desconocido privado cuando esté codificado en el mensaje Q.931 y como una e164 AliasAddress cuando esté codificado en los mensajes RAS.

Cuadro 14/H.225.0 – Correspondencia de representaciones de tipos de números

Tipo de número	Representación Q.931	Representación RAS H.225
Desconocido (modo de interfuncionamiento por defecto y de la versión 1)	Plan de numeración privado – desconocido	e164 AliasAddress
Privado, desconocido	Plan de numeración privado – desconocido	e164 AliasAddress
Privado, abonado/abreviado, etc.	Plan de numeración privado, tipo de número según su alcance	privateNumber of PartyNumber AliasAddress
Número público, todos los tipos	RDSI/plan de numeración telefónico	publicNumber of PartyNumber AliasAddress

La estructura **Endpoint (punto extremo)** se utiliza para indicar la información de reserva, redundante o alternativa sobre un punto extremo:

– **nonStandardData (datos no normalizados)**: Lleva información no definida en esta Recomendación **(**por ejemplo, datos patentados**)**.

– **aliasAddress (dirección de alias)**: Es una lista de direcciones de alias por las que otros puntos extremos pueden identificar este punto extremo.

– **callSignalAddress (dirección de señalización de llamada)**: Es la dirección de transporte de señalización de llamada para este punto extremo.

– **rasAddress (dirección RAS)**: Es la dirección de transporte de registro y situación para este punto extremo.

– **endpointType (tipo de punto extremo)**: Especifica el tipo del punto extremo.

– **tokens (testigos)**: Testigos asociados en este punto extremo (esto es, el punto extremo descrito en la estructura Endpoint**)**.

– **cryptoTokens (testigos criptados)**: Testigos criptados con este punto extremo (esto es, el punto extremo descrito en la estructura Endpoint).

– **priority (prioridad)**: Se utiliza cuando se presenta una SECUENCIA de puntos extremos. Se prefieren los puntos extremos con números de prioridad inferior a los puntos extremos con números de prioridad superior. Los puntos extremos sin números de prioridad son equivalentes a los que tienen una prioridad de 0 (la prioridad más alta).

– **remoteExtensionAddress (dirección de extensión lejano)**: Contiene la dirección de alias de un punto extremo en los ceros en que es necesaria esta información para atravesar multiples pasarelas.

– **destExtraCallInfo (información de llamada suplementaria de destino)**: Contiene direcciones externas para múltiples llamadas.

La estructura **controlador de acceso alternativo (AlternateGK)** se utiliza para indicar una lista de controladores de acceso alternativos o de reserva:

– **rasAddress (dirección ras)**: Dirección de transporte utilizada para la señalización RAS.

– **gatekeeperIdentifier (identificador de controlador de acceso)**: Se incluye opcionalmente para identificar el controlador de acceso de reserva o alternativo. Si se suministra, deberá incluirse en futuros mensajes RAS enviados al controlador de acceso de reserva.

– **needToRegister (necesidad de registro)**: Fijado a VERDADERO para indicar que el punto extremo se debe registrar con el controlador de acceso alternativo antes de enviar otras peticiones RAS.

– **priority (prioridad)**: Indica la prioridad del controlador de acceso de reserva o alternativo. Un número bajo implica una prioridad alta.

La estructura **información de controlador de acceso alternativo (AltGKInfo)** se utiliza para dar información sobre controladores de acceso alternativos:

– **alternateGatekeeper (controlador de acceso alternativo)**: Secuencia de alternateGateKeeper prioritarios para gatekeeperIdentifier y rasAddress para que el cliente haga un reintento de la petición.

– **altGKisPermanent (el controlador de acceso alternativo es permanente)**: VERDADERO si todas las señales RAS futuras se redireccionan a una dirección del alternateGatekeeper, FALSO si sólo se redirecciona el mensaje que causó el rechazo.

Un controlador de acceso puede enviar a un punto extremo una lista de controladores de acceso alternativos en diversos mensajes. Cuando se comunica con su controlador de acceso, un punto extremo que aplica el mecanismo de controlador de acceso alternativo reemplazará cualquier lista de controladores de acceso alternativos recibida previamente por la lista de controladores de acceso alternativos recibida más recientemente. Es posible que un controlador de acceso alternativo envíe una lista de controladores de acceso alternativos. Si un punto de acceso envía una petición a un controlador de acceso alternativo que potencialmente transforme su controlador de acceso permanente, aceptará la nueva lista de controladores de acceso alternativos. De otro modo, si el controlador de acceso alternativo no transformara potencialmente su controlador de acceso permanente, se ignorará toda lista recibida de controladores de acceso alternativo. Un controlador de acceso puede tener la posibilidad de convertirse en un controlador de acceso permanente de punto extremo si el controlador de acceso actual se torna no responsable o si la bandera "AltGKisPermanent" se pone a VERDADERO en la estructura " altGKInfo".

La estructura **opciones de la serie Q (Qseries Options)** suministra información al controlador de acceso o a otros puntos extremos relativa al soporte por un terminal de protocolos opcionales de la serie Q. Se utiliza en los mensajes ARQ, SETUP y RRQ.

GloballyUniqueID (identificador único a nivel mundial) y ConferenceIdentifier (identificador de conferencia) son considerados identificadores únicos a nivel mundial, cuya utilización se describe en la Recomendación H.323. Un GloballyUniqueID se codifica primero con el octeto cero. Un GloballyUniqueID está formado según el cuadro 15.

Cuadro 15/H.225.0 – Formación GloballyUniqueID

Campo	Tipo de datos	Número de octetos	Nota
time_low	Entero de 32 bits sin signo	0-3	El campo bajo de la indicación de tiempo
time_mid	Entero de 16 bits sin signo	4-5	El campo medio de la indicación de tiempo
time_hi_and_version	Entero de 16 bits sin signo	6-7	El campo alto de la indicación de tiempo multiplexado con el número de versión
clock_seq_hi_and_reserved	Entero de 8 bits sin signo	8	El campo alto de la secuencia de reloj multiplexado con la variante

Cuadro 15/H.225.0 – Formación GloballyUniqueID *(fin)*

Campo	Tipo de datos	Número de octetos	Nota
clock_seq_low	Entero de 8 bits sin signo	9	El campo bajo de la secuencia de reloj
node	Entero de 48 bits sin signo	10-15	El identificador de nodo único a nivel espacial

El GloballyUniqueID está formado por un registro de 16 octetos y no debe contener relleno entre campos. El tamaño total es de 128 bits.

Para que sea menos confusa la asignación de bits dentro de los octetos, el registro del GloballyUniqueID se define únicamente en términos de campos que son números enteros de octetos. El número de versión es multiplexado con la indicación de tiempo (*time_high*), y el campo de la variante es multiplexado con la secuencia de reloj (*clock_seq_high*).

La indicación de tiempo es un valor de 60 bits representado por el Tiempo Universal Coordinado (UTC, *Coordinated Universal Time*) como un cómputo de intervalos de 100 nanosegundos a partir de las 00:00:00.00 del 15 de octubre de 1582 (fecha de la reforma gregoriana del calendario cristiano).

El número de versión es multiplexado en los 4 bits más significativos del campo *time_hi_and_version*, y se fija a 1 (número binario 0001).

El campo variable determina la disposición del GloballyUniqueID. La estructura de un GloballyUniqueID de DCE se fija en las distintas versiones. Es posible que otras variantes de GloballyUniqueID no puedan interfuncionar con un GloballyUniqueID de DCE. El interfuncionamiento de los GloballyUniqueID se define como la aplicabilidad de operaciones tales como conversión de cadena, comparación y ordenamiento del léxico en los distintos sistemas. El campo *variante* está formado por un número variable de los bits más significativos (MSB) del campo *clock_seq_hi_and_reserved* (véase el cuadro 16).

Cuadro 16/H.225.0 – Contenido del campo de variante DCE

msb1	msb2	msb3	Descripción
0	–	–	Reservado, compatibilidad hacia atrás NCS
1	0	–	Variante DCE
1	1	0	Reservado, Microsoft Corporation GUID
1	1	1	Reservado para futura definición

La secuencia de reloj es necesaria para detectar las posibles pérdidas de monotonicidad del reloj. La secuencia de reloj se codifica en los 6 bits menos significativos del campo *clock_seq_hi_and_reserved* y en el campo *clock_seq_low*.

El campo *node* está formado por la dirección IEEE, generalmente la dirección central. Para sistemas con múltiples nodos IEEE 802, puede utilizarse cualquier dirección de nodo disponible. El octeto direccionado más bajo (octeto número 10) contiene el bit mundial/local y el bit unidifusión/multidifusión, y es el primer octeto de la dirección transmitida por una red de paquetes 802.3.

El valor de la secuencia de reloj se cambiará cuando:

- El generador del GloballyUniqueID detecte que el valor local del UTC ha retrocedido; esto puede deberse al funcionamiento normal del servicio de tiempo del DCE.
- El generador del GloballyUniqueID haya perdido su estado del último valor de UTC utilizado, lo que indica que el tiempo ha retrocedido; esto suele ocurrir en el rearranque.

Aunque un nodo esté operativo, el generador del GloballyUniqueID conserva siempre el último UTC utilizado para crear un GloballyUniqueID. Cada vez que se crea un nuevo GloballyUniqueID, el *UTC* actual se compara con el valor conservado y si el valor actual es menor (caso del reloj no monotónico) o si se ha perdido el valor conservado, la *secuencia de reloj* se incrementa en módulo 16 384, evitando así la duplicación del GloballyUniqueID.

La *secuencia de reloj* deberá inicializarse en un número aleatorio para reducir al mínimo la correlación entre sistemas.

Un GloballyUniqueID se genera aplicando el siguiente algoritmo:

1) Determinar los valores de la indicación de tiempo basada en UTC y la secuencia de reloj que se ha de utilizar en el GloballyUniqueID.
2) Fijar el campo *time_low* igual a los 32 bits menos significativos (bits numerados de 0 a 31 inclusive) de la indicación de tiempo en el mismo orden de importancia.
3) Fijar el campo *time_mid* igual a los bits numerados de 32 a 47 inclusive de la indicación de tiempo en el mismo orden de importancia.
4) Fijar los 12 bits menos significativos (bits numerados de 0 a 11 inclusive) del campo *time_hi_and_version* igual a los bits numerados de 48 a 59 inclusive de la indicación de tiempo en el mismo orden de importancia.
5) Fijar los 4 bits más significativos (bits numerados de 12 a 15 inclusive) del campo *time_hi_and_version* en el número de versión de 4 bits correspondientes a la versión del GloballyUniqueID que se crea, tal como se indicó en el cuadro 15.
6) Fijar el campo *clock_seq_low* en los 8 bits menos significativos (bits numerados de 0 a 7 inclusive) de la *secuencia de reloj* en el mismo orden de importancia.
7) Fijar los 6 bits menos significativos (bits numerados de 0 a 5 inclusive) del campo *clock_seq_hi_and_reserved* en los 6 bits más significativos (bits numerados de 8 a 13 inclusive) de la *secuencia de reloj* en el mismo orden de importancia.
8) Fijar los 2 bits más significativos (bits numerados de 6 a 7) del campo *clock_seq_hi_and_reserved* a 0 y 1, respectivamente.
9) Fijar el campo *node* en la dirección IEEE de 48 bits en el mismo orden de importancia que la dirección.

Si un sistema desea generar un GloballyUniqueID pero no tiene una tarjeta de red conforme a IEEE 802 u otra fuente de direcciones IEEE 802, se utilizará un método alternativo para generar un valor de sustitución de la dirección. La solución ideal es obtener un número aleatorio de calidad criptográfica de 47 bits, y utilizarlo como los 47 bits más significativos del ID de nodo, con el bit menos significativo del primer octeto del ID de nodo fijado a 1. Éste es el bit unidifusión/multidifusión, que no se fijará nunca en las direcciones IEEE 802 obtenidas a partir de las tarjetas de red; por lo tanto, nunca puede surgir un conflicto entre GloballyUniqueID generados por máquinas que dispongan o no de tarjetas de red.

Aunque algunos sistemas no tengan una primitiva con la que generar números aleatorios de calidad criptográfica, en muchos de ellos se dispone por lo general de un número relativamente importante de fuentes de aleatoridad a partir de las cuales puede generarse uno de esos números. Esas fuentes son específicas del sistema pero suelen incluir el porcentaje de memoria utilizada, la capacidad de memoria principal en bytes, la capacidad de memoria principal libre en bytes, el tamaño del fichero

de desplazamiento por páginas en bytes, los bytes libres del fichero de desplazamiento por páginas, el tamaño total del espacio de la dirección virtual de usuario en bytes, los bytes totales disponibles del espacio de la dirección de usuario, el tamaño de la unidad del disco de carga en bytes, el espacio libre del disco en la unidad de carga en bytes, el tiempo actual, el tiempo transcurrido desde la carga inicial del sistema, las dimensiones de cada fichero en los diversos directorios del sistema, etc.

Para ser utilizada en un texto legible por el hombre, la representación en cadena de un GloballyUniqueID se especifica como una secuencia de campos, algunos de los cuales se separan con guiones.

Cada campo es tratado como un entero y su valor se imprime como una cadena de dígitos hexadecimales rellena de ceros, cuyo primer dígito es el dígito más significativo. Los valores hexadecimales de a a f inclusive se representan con caracteres en minúscula en la salida, en la entrada, son independientes del tamaño de los caracteres que los represente. La secuencia es la misma que el tipo construido de GloballyUniqueID.

La definición formal de la representación en cadena del GloballyUniqueID viene dada por la siguiente forma Backus Naur (BNF) ampliada:

```
UUID                   = <time_low> <hyphen> <time_mid> <hyphen>
                         <time_high_and_version> <hyphen>
                         <clock_seq_and_reserved>
                         <clock_seq_low> <hyphen> <node>
time_low               = <hexOctet> <hexOctet> <hexOctet> <hexOctet>
time_mid               = <hexOctet> <hexOctet>
time_high_and_version  = <hexOctet> <hexOctet>
clock_seq_and_reserved = <hexOctet>
clock_seq_low          = <hexOctet>
node                   = <hexOctet><hexOctet><hexOctet>
                         <hexOctet><hexOctet><hexOctet>
hexOctet               = <hexDigit> <hexDigit>p
hexDigit               = <digit | <a> | <b | <c | <d | <e | <f
digit                  = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |
                         "8" | "9"
hyphen                 = "-"
a                      = "a" | "A"
b                      = "b" | "B"
c                      = "c" | "C"
d                      = "d" | "D"
e                      = "e" | "E"
f                      = "f" | "F"
```

A continuación se indica un ejemplo de la representación en cadena de un GloballyUniqueID:

f81d4fae-7dec-11d0-a765-00a0c91e6bf6

TimeToLive (tiempo de vida) es el número de segundos durante los cuales se considera válido un registro.

7.7 Soporte necesario de los mensajes RAS

El cuadro 17 muestra los mensajes RAS que son soportados por diferentes tipos de puntos extremos.

Cuadro 17/H.225.0 – Situación de los mensajes RA

Mensajes RAS	Punto extremo (Tx)	Punto extremo (Rx)	Controlador de acceso (Tx)	Controlador de acceso (Rx)
GRQ	O			M
GCF		O	M	
GRJ		O	M	
RRQ	M			M
RCF		M	M	
RRJ		M	M	
URQ	O	M	O	M
UCF	M	O	M	O
URJ	O	O	M	O
ARQ	M			M
ACF		M	M	
ARJ		M	M	
BRQ	M	M	O	M
BCF	M (nota 1)	M	M	O
BRJ	M	M	M	O
IRQ		M	M	
IRR	M			M
IACK		O	CM	
INAK		O	CM	
DRQ	M	M	O	M
DCF	M	M	M	M
DRJ	M (nota 2)	M	M	M
LRQ	O		O	M
LCF		O	M	O
LRJ		O	M	O
NSM	O	O	O	O
XRS	M	M	M	M
RIP	CM	M	CM	M
RAI	O			M
RAC		O	M	

M: Obligatorio (*mandatory*), O: Opcional, F: Prohibido (*forbidden*), CM: Condicionalmente obligatorio (*conditionally mandatory*), blanco: No aplicable.

NOTA 1 – Obsérvese que si un controlador de acceso envía un BRQ solicitando una velocidad inferior, el punto extremo responderá con BCF si la velocidad más baja es soportada, en los demás casos con BRJ. Si un controlador de acceso envía un BRQ solicitando una velocidad superior, el punto extremo puede responder con BCF o BRJ.

NOTA 2 – El terminal no enviará DRJ mientras está en una llamada en respuesta a DRQ procedente de un controlador de acceso.

7.8 Mensajes de descubrimiento de terminal y de pasarela

El mensaje GRQ pide que cualquier controlador de acceso que lo reciba responda con un GCF que le conceda permiso para registrar. El GRJ es un rechazo de esta petición que indica que el punto extremo solicitante debe buscar otro controlador de acceso.

7.8.1 GatekeeperRequest (GRQ) (petición de controlador de acceso)

Se señala que se envía a un GRQ por punto extremo lógico; así, una MCU o una pasarela podría mandar muchos.

El mensaje GRQ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del llamante. Debe ser devuelto por el llamado en cualesquiera mensajes asociados con este mensaje concreto.

protocolIdentifier (identificador de protocolo) – Identifica la antigüedad del envío del punto extremo según la Recomendación H.225.0.

nonStandardData (datos no normalizados) – Transporta información no definida en esta Recomendación (por ejemplo, datos patentados).

rasAddress (dirección ras) – Es la dirección de transporte que este punto extremo utiliza para mensajes de registro y de situación.

endpointType (tipo de punto extremo) – Especifica el tipo o tipos del punto extremo que está registrando (el bit MC no será fijado por él mismo).

gatekeeperIdentifier (identificador de controlador de acceso) – Cadena para identificar el controlador de acceso desde el que el terminal desearía recibir permiso para registrarse. Un **gatekeeperIdentifier** faltante o de cadena nula indica que el terminal está interesado en cualquier controlador de acceso disponible.

callServices (servicios de llamada) – Proporciona información sobre el soporte de protocolos opcionales de la serie Q para el controlador de acceso y el terminal llamado.

endpointAlias (alias de punto extremo) – Lista de direcciones de alias por la cual otros terminales pueden identificar este terminal.

alternateEndpoints (puntos extremos alternativos) – Secuencia de alternativas de puntos extremos prioritarios para rasAddress, endpointType o endpointAlias.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

authenticationCapability (capacidad de autenticación) – Indica los mecanismos de autenticación soportados por el punto extremo.

algorithmOIDs – Indica el conjunto completo de algoritmos de criptación soportados por el punto extremo.

integrity (integridad) – Indica al recibiente el mecanismo de integridad que se debe aplicar en los mensajes RAS.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación de mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta a todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.8.2 GatekeeperConfirm (GCF) (confirmación de controlador de acceso)

El mensaje GCF incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Será el mismo valor que fue pasado en el GRQ.

protocolIdentifier (identificador de protocolo) – Identifica la antigüedad del controlador de acceso aceptador según la Recomendación H.225.0.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

gatekeeperIdentifier (identificador de controlador de acceso) – Cadena para identificar el controlador de acceso que está enviando el GCF.

rasAddress (dirección ras) – Dirección de transporte que el controlador de acceso utiliza para los mensajes de registro y situación.

alternateGatekeeper (controlador de acceso alternativo) – Secuencia de alternativas prioritarias para gatekeeperIdentifier y rasAddress. El cliente deberá utilizar estas alternativas en el futuro si el controlador de acceso no responde a una petición.

authenticationMode (modo de autenticación) – Indica el mecanismo de autenticación que se ha de utilizar. El controlador de acceso debe elegir el **authenticationMode** de la **authenticationCapability** proporcionada por el punto extremo en GRQ.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

algorithmOID – Indica el algoritmo de criptación requerido por el controlador de acceso.

integrity (integridad) – Indica al recibiente el mecanismo de integridad que se debe aplicar en los mensajes RAS.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta a todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.8.3 GatekeeperReject (GRJ) (rechazo de controlador de acceso)

El mensaje GRJ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Será el mismo valor que fue pasado en el GRQ.

protocolIdentifier (identificador de protocolo) –Identifica la antigüedad del controlador de acceso rechazante.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

gatekeeperIdentifier (identificador de controlador de acceso) – Cadena para identificar el controlador de acceso que está enviando el GRJ.

rejectReason (motivo del rechazo) – Codifica por qué el GRQ fue rechazado por este controlador de acceso.

altGKInfo (información de controlador de acceso alternativo) – Información opcional sobre controladores de acceso alternativos. Si se suministra esta información, un punto extremo debe retransmitir la petición a uno de los controladores de acceso alternativos enumerados. Si un controlador de acceso alternativo rechaza la petición, el punto extremo aceptará el rechazo. Si un controlador de acceso alternativo no responde, el punto extremo puede enviar la petición a otro controlador de acceso alternativo en la lista.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación de mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta a todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.9 Mensajes de registro de terminal y de pasarela

El RRQ es una petición de registro de un terminal a un controlador de acceso. Si éste responde con un RCF, el terminal utilizará el controlador de acceso respondedor para futuras llamadas. Si el controlador de acceso responde con un RRJ, el terminal debe buscar otro controlador de acceso en el que registrarse.

7.9.1 RegistrationRequest (RRQ) (petición de registro)

El mensaje RRQ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del llamante. Debe ser devuelto por el llamado en cualesquiera mensajes asociados con este mensaje concreto.

protocolIdentifier (identificador de protocolo) – Identifica la antigüedad del punto extremo emisor según la Recomendación H.225.0

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

discoveryComplete (descubrimiento completo) – Se pone a VERDADERO si el punto extremo solicitante ha precedido este mensaje con el procedimiento de descubrimiento de controlador de acceso; se pone a FALSO si se trata del registro solamente. Obsérvese que el registro puede envejecer y el punto extremo obtendrá un fallo en un RRQ o ARQ con un código de motivos de **discoveryRequired** o **notRegistered** respectivamente. Esto indica que el punto extremo debe efectuar el procedimiento de descubrimiento (dinámico o estático) antes de emitir RRQ con **discoveryComplete** puesto a VERDADERO.

callSignalAddress (dirección de señalización de llamada) – Es la dirección de transporte de señalización de llamada para este punto extremo. Si se soportan múltiples transportes, deben registrarse todos a la vez.

rasAddress (dirección ras) – Es la dirección de transporte de registro y de situación para este punto extremo.

terminalType (tipo de terminal) – Especifica el tipo (o tipos) del punto extremo que se está(n) registrando; adviértase que el bit MC no será fijado por él mismo; se fijará también el bit de terminal, de MCU, de pasarela o de controlador de acceso. Si se proporciona información sobre el vendedor, ésta será idéntica a la proporcionada en **endpointVendor**.

terminalAlias (alias de terminal) – Este valor opcional es una lista de direcciones de alias, mediante las cuales otros terminales pueden identificar este terminal. Si el **terminalAlias** es nulo, o no está presente una dirección E.164, una dirección E.164 puede ser asignada por el controlador de acceso, e incluirse en el RCF. Si se dispone de un ID de correo electrónico para el punto extremo deberá registrarse. Adviértase que múltiples direcciones de alias pueden designar las mismas direcciones de transporte. Todos los alias de punto extremo se incluirán en cada RRQ.

gatekeeperIdentifier (identificador de controlador de acceso) – Cadena para identificar el controlador de acceso en el que el terminal desea registrarse.

endpointVendor (vendedor de punto extremo) – Información sobre el vendedor de punto extremo.

alternateEndpoints (puntos extremos alternativos) – Secuencia de alternativas de puntos extremos prioritarios para callSignalAddress, rasAddress, terminalType o terminalAlias.

timeToLive (tiempo de vida) – Duración de la validez del registro, en segundos, transcurrida la cual el controlador de acceso puede considerar que el registro ha caducado.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

keepAlive (mantener vivo) – Si está fijado en VERDADERO indica que el punto extremo ha enviado este RRQ como un mensaje de "mantener vivo". Un punto extremo puede enviar un RRQ ligero que conste únicamente de rasAddress, keepAlive, endpointIdentifier, gatekeeperIdentifier, tokens y timeToLive. Cuando un controlador de acceso reciba una RRQ con un campo keepAlive fijado en VERDADERO ignorará los campos que no sean endpointIdentifier, gatekeeperIdentifier, tokens y timeToLive. El campo rasAddress en un mensaje RRQ será únicamente utilizado por un controlador de acceso como el destino para un RRJ cuando el punto extremo no se registra.

endpointIdentifier (identificador de punto extremo) – Identificador de punto extremo proporcionado por el controlador de acceso durante el RCF original.

willSupplyUUIEs (proporcionará los UUIE) – Si está fijado en VERDADERO, indica que el punto extremo suministrará información sobre el mensaje Q.931 en mensajes IRR, si lo solicita el controlador de acceso.

maintainConnection – Si es VERDADERO indica que el emisor del mensaje puede soportar una conexión de señalización cuando ninguna llamada no se selecciona actualmente a través de la conexión.

supportsAnnexECallSignalling – Si es VERDADERO, indica que el emisor de este mensaje puede señalizar la llamada de un canal de transporte no fiable como se describe en la Recomendación H.323 anexo E.

7.9.2 RegistrationConfirm (RCF) (confirmación de registro)

El mensaje RCF incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Debe ser el mismo valor que fue pasado en el RRQ.

protocolIdentifier (identificador de protocolo) – Identifica la antigüedad del controlador de acceso aceptador.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

callSignalAddress (dirección de señalización de llamada) – Es una formación de direcciones de transporte para mensajes de señalización de llamada H.225.0; una para cada transporte al que responderá el controlador de acceso. Esta dirección incluye el identificador de TSAP.

terminalAlias (alias de terminal) – Este valor opcional es una lista de direcciones de alias, mediante las cuales otros terminales pueden identificar este terminal.

gatekeeperIdentifier (identificador de controlador de acceso) – Cadena para identificar el controlador de acceso que ha aceptado el registro de terminales.

endpointIdentifier (Identificador de punto extremo) – Cadena de identidad de terminal asignada por un controlador de acceso; se devolverá en eco en mensajes RAS subsiguientes.

alternateGatekeeper (controlador de acceso alternativo) – Secuencia de controladores de acceso alternativos prioritarios para gatekeeperIdentifier y rasAddress. El cliente deberá utilizar en el futuro estas alternativas si el controlador de acceso no responde a una petición.

timeToLive (tiempo de vida) – Duración de la validez del registro, en segundos, transcurrida la cual el controlador de acceso puede considerar que el registro ha caducado.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

willRespondToIRR (responderá a IRR) – Verdadero si el controlador de acceso envía un mensaje IACK o INAK en respuesta a un mensaje IRR no solicitado con su campo **needsResponse** fijado en VERDADERO.

preGrantedARQ (ARQ concedido previamente) – Indica eventos cuya admisión ha concedido previamente el controlador de acceso. Esto permite tiempos de establecimiento de llamada más breves en entornos en los que la admisión está garantizada por medios distintos del intercambio ARQ/ACF. Obsérvese que incluso si estos campos están fijados en VERDADERO, un punto extremo puede todavía enviar un ARQ al controlador de acceso por razones tales como la traducción de dirección, o porque el punto extremo no soporta este modo de señalización modificado. Si la secuencia de **preGrantedARQ** no está presente, la señalización ARQ se utilizará en todos los casos. Los campos son:

– **makeCall (efectuar llamada)** – Si la bandera **makeCall** es VERDADERO, el controlador de acceso ha concedido permiso previamente al punto extremo para iniciar llamadas sin enviar primero un ARQ. Si la bandera **makeCall** es FALSO, el punto extremo deberá enviar siempre un ARQ a fin de obtener permiso para efectuar una llamada.

– **useGKCallSignalAddressToMakeCall (utilizar dirección de señalización de llamada GK para efectuar llamada**) – Si las banderas **makeCall** y **useGKCallSignalAddressTo MakeCall** están fijadas ambas en VERDADERO, y el punto extremo no envía entonces un ARQ al controlador de acceso para efectuar una llamada, dicho punto deberá enviar la

señalización de todas las llamadas H.225.0 al canal de señalización de llamada del controlador de acceso.

– **answerCall (responder a llamada)** – Si la bandera **answerCall** es VERDADERO, el controlador de acceso ha concedido permiso previamente al punto extremo para responder a llamadas sin enviar primero un ARQ. Si la bandera **answerCall** es FALSO, el punto extremo deberá enviar siempre un ARQ a fin de obtener permiso para responder a una llamada.

– **useGKCallSignalAddressToAnswer (utilizar dirección de señalización de llamada GK para responder)** – Si las banderas **answerCall** y **useGKCallSignalAddressToAnswer** están fijadas ambas en VERDADERO y un punto extremo no envía entonces un ARQ al controlador de acceso para responder a una llamada, dicho punto velará por que la señalización de todas las llamadas H.225.0 provenga del controlador de acceso. Si se ha ordenado a un punto extremo que utilice el controlador de acceso cuando responda a una llamada, pero no sabe si una llamada entrante proviene del controlador de acceso (lo que quizás implique observar la dirección de transporte), el punto extremo emitirá un ARQ independientemente del estado en que esté la bandera **useGKCallSignalAddressToAnswer**.

– **irrFrequencyInCall (frecuencia del IRR en la llamada)** – Este campo indica la frecuencia, en segundos, de los mensajes IRR enviados al controlador de acceso cuando el punto extremo está en una o más llamadas. Si no está presente, el controlador de acceso no desea mensajes IRR no solicitados. Cuando el punto extremo envía esos mensajes IRR, el valor de referencia de llamada será único para el terminal, como hubiera sido generado en una petición de admisión. Sin embargo, éste no es un valor de referencia de llamada "normal" y no se puede reutilizar para nueva comunicación (DRQ, IRQ o BRQ). El identificador de llamada será el mismo que el utilizado en los mensajes del canal de señalización de llamada para la llamada pertinente.

– **totalBandwidthRestriction (restricción de anchura de banda total)** – Este campo limita la utilización total de la anchura de banda para el punto extremo cuando se encuentra en llamada. Si no está presente, no hay una restricción de anchura de banda constante.

– **useAnnexECallSignalling (utilización de señalización de llamada según anexo E)** – Si está en VERDADERO, este parámetro indica que el punto extremo que recibe este mensaje utilizará exclusivamente señalización de llamada en un canal de transporte no fiable como se describe en el anexo E/H.323 cuando se colocan llamadas. Si es FALSO, no utilizará el anexo E/H.323 para la señalización de llamada. Si no está presente (o recibido por un controlador de acceso de versión 2 o menor), el punto extremo puede tratar ambos métodos para compatibilidad hacia atrás como se describe en el anexo E/H.323.

maintainConnection (mantener conexión) – Si es VERDADERO, indica que el controlador de acceso (en el caso de encaminamiento de control de acceso) puede soportar una conexión de señalización cuando ninguna llamada está señalizada actualmente sobre la conexión.

7.9.3 RegistrationReject (RRJ) (rechazo de registro)

El mensaje RRJ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Debe ser el mismo valor que fue pasado en el RRQ.

protocolIdentifier (identificador de protocolo) – Identifica la antigüedad del controlador de acceso rechazante.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

rejectReason (motivo del rechazo) – Motivo del rechazo del registro.

gatekeeperIdentifier (identificador de controlador de acceso) – Cadena para identificar el controlador de acceso que ha rechazado el registro del terminal.

altGKInfo (información de controlador de acceso alternativo) – Información opcional sobre controladores de acceso alternativos. Si se suministra esta información, un punto extremo debe retransmitir la petición a uno de los controladores de acceso alternativos enumerados. Si un controlador de acceso alternativo rechaza la petición, el punto extremo aceptará el rechazo. Si un controlador de acceso alternativo no responde, el punto extremo puede enviar la petición a otro controlador alternativo en la lista.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.10 Mensajes de desregistro de terminal/controlador de acceso

7.10.1 UnregistrationRequest (URQ) (petición de desregistro)

El URQ solicita que se interrumpa la asociación entre un terminal y un controlador de acceso. Adviértase que ese registro es bidireccional; es decir, un controlador de acceso puede pedir a un terminal que se considere a sí mismo desregistrado, y un terminal puede informar a un controlador de acceso que está revocando un registro anterior.

El mensaje URQ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del emisor. Debe ser devuelto por el receptor en cualquier respuesta asociada con este mensaje concreto.

callSignalAddress (dirección de señalización de llamada) – Ésta es una o más de las direcciones de señalización de llamada de transporte para este punto extremo que han de ser desregistradas.

endpointAlias (alias de punto extremo) – Este valor opcional es una lista de direcciones de alias, mediante las cuales otros terminales pueden identificar este terminal. Si este campo opcional no está presente, todos los alias son desregistrados en un solo mensaje. La dirección E.164, si está asignada, es necesaria. Sólo se desregistran los valores enumerados aquí; esto permite, por ejemplo, desregistrar un H323_ID mientras se abandona la dirección E.164 registrada.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

endpointIdentifier (identificador de punto extremo) – Confirmación de identidad, no es enviado por el controlador de acceso.

alternateEndpoints (puntos extremos alternativos) – Secuencia de alternativas de puntos extremos prioritarios para callSignalAddress o endpointAlias.

gatekeeperIdentifier (identificador de controlador de acceso) – Un gatekeeperIdentifier que el cliente recibió en la lista de alternateGatekeeper en el mensaje RCF proveniente del controlador de acceso cuando se registró o en un mensaje URJ anterior. Utilizado como reserva si el controlador de acceso original no respondió a la petición o la rechazó.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

reason (motivo) – Se utiliza cuando el controlador de acceso envía el mensaje URQ para indicar por qué considera desregistrado el punto extremo.

7.10.2 UnregistrationConfirm (UCF) (confirmación de desregistro)

El mensaje UCF incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Debe ser el mismo valor que fue pasado en el URQ.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.10.3 UnregistrationReject (URJ) (rechazo de desregistro)

El mensaje URJ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Debe ser el mismo valor que fue pasado en el URQ.

rejectReason (motivo del rechazo) – Motivo del rechazo del registro.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

altGKInfo (información de controlador de acceso alternativo) – Información opcional sobre controladores de acceso alternativos. Si se suministra esta información, un punto extremo debe retransmitir la petición a uno de los controladores de acceso alternativos enumerados. Si un controlador de acceso alternativo rechaza la petición, el punto extremo aceptará el rechazo. Si un controlador de acceso alternativo no responde, el punto extremo puede enviar la petición a otro controlador alternativo en la lista.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de

integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.11 Mensajes de admisión de terminal a controlador de acceso

El mensaje ARQ solicita que a un punto extremo le sea permitido el acceso a la red de paquetes por el controlador de acceso, que concede la petición con un ACF o la deniega con un ARJ.

7.11.1 Petición de admisión (ARQ, *admissionRequest*)

El mensaje ARQ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del emisor. Debe ser devuelto por el llamado en cualesquiera mensajes asociados con este mensaje concreto.

callType (tipo de llamada) – Utilizando este valor, el controlador de acceso puede intentar determinar la utilización de anchura de banda "real". El valor por defecto es **pointToPoint** para todas las llamadas. Debe reconocerse que el tipo de llamada puede cambiar dinámicamente durante la misma, y que el tipo de llamada final puede no ser conocido cuando se envía el ARQ.

callModel (modelo de llamada) – Si es **direct**, el punto extremo es solicitando el modelo de llamada directo de terminal a terminal. Si es **gatekeeperRouted**, el punto extremo está solicitando el modelo mediado por el controlador de acceso. No es necesario que el controlador de acceso acceda a esta petición.

endpointIdentifier (identificador de punto extremo) – Es un identificador de punto extremo que fue asignado al terminal por RCF.

destinationInfo (información de destino) – Secuencia de direcciones de alias para el destino, tales como direcciones E.164 o H323_ID. Cuando se envía el ARQ para responder a una llamada, destinationInfo indica el destino de la llamada (el punto extremo que responde). Si en un controlador de acceso se registra al menos un alias y en el ARQ no se registran dos alias a distintas personas, el controlador de acceso reconocerá el ARQ como referido a la identidad registrada. En el caso de alias en conflicto se rechazará la petición de admisión con la causa AliasesInconsistent. Si el controlador de acceso no proporciona esta validación, considerará que la primera dirección registrada es el destino.

destCallSignalAddress (dirección de señalización de llamada de destino) – Dirección de transporte utilizada en el destino para la señalización de llamada.

destExtraCallInfo (información de llamada extra de destino) – Contiene direcciones exteriores para múltiples llamadas.

srcInfo (información scr) – Secuencia de direcciones de alias para el punto extremo de origen, tales como direcciones E.164 o H323_ID. Cuando se envía el ARQ para responder a una llamada, srcInfo indica el originador de la llamada.

srcCallSignalAddress (dirección de señalización de llamada scr) – Dirección de transporte utilizada en el origen para la señalización de llamada.

bandWidth (anchura de banda) – El número de 100 bits solicitado para la llamada bidireccional. Por ejemplo, una llamada de 128 kbit/s se señalizaría como una petición de 256 kbit/s. El valor se refiere sólo a la velocidad binaria de audio y de vídeo, incluidos encabezamientos y tara.

callReferenceValue (valor de referencia de llamada) – El CRV de la Q.931 para esta llamada; sólo tiene validez local. Es utilizado por un controlador de acceso para asociar la ARQ con una determinada llamada.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

callServices (servicios de llamada) – Proporciona información sobre el soporte de protocolos opcionales de la serie Q para el controlador de acceso y el terminal llamado.

conferenceID (ID de conferencia) – Identificador de conferencia exclusivo.

activeMC – Si es VERDADERO, la parte llamante tiene un MC activo; en los demás casos, es FALSO.

answerCall – Se utiliza para indicar a un controlador de acceso que una llamada está entrando.

canMapAlias (puede copiar alias) – Si está fijado en VERDADERO indica que, si el ACF resultante contiene los campos **destinationInfo**, **destExtraCallInfo** y/o **remoteExtension**, el punto extremo puede copiar esta información en los campos **destinationAddress**, **destExtraCallInfo** y **remoteExtensionAddress** del mensaje ESTABLECIMIENTO, respectivamente. Si el GK reemplaza la información de direccionamiento del ARQ y **canMapAlias** es FALSO, el controlador de acceso rechazará el ARQ.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

srcAlternatives (alternativas src) – Secuencia de alternativas de punto extremo de origen prioritarias para srcInfo, srcCallSignalAddress, o rasAddress.

destAlternatives (alternativas de destino) – Secuencia de alternativas de punto extremo de destino prioritarias para destinationInfo o destCallSignalAddress.

gatekeeperIdentifier (identificador de controlador de acceso) – Un gatekeeperIdentifier que el cliente recibió en la lista de alternateGatekeeper en el mensaje RCF proveniente del controlador de acceso cuando se registró o en un mensaje ARJ anterior. Utilizado como reserva si el controlador de acceso original no respondió a la petición o la rechazó.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

transportQOS (calidad de servicio de transporte) – Un punto extremo puede utilizarlo para indicar su capacidad para reservar recursos de transporte.

WillSupplyUUIEs – Si está fijado en VERDADERO, indica que el punto extremo suministrará información sobre el mensaje Q.931 en mensajes IRR, si lo solicita el controlador de acceso.

La estructura de TransportQOS incluye lo siguiente:

endpointControlled (punto extremo controlado) – El punto extremo aplicará su propio mecanismo de reserva.

gatekeeperControlled (controlador de acceso controlado) – El controlador de acceso efectuará la reserva de recursos en nombre del punto extremo.

noControl (sin control) – No es necesaria ninguna reserva de recursos.

NOTA – No se requieren los dos elementos **destinationInfo** y **destCallSignalAddress**, pero al menos uno estará presente, a menos que el punto extremo esté respondiendo a una llamada. No hay ninguna regla absoluta sobre cuál se prefiere, lo que puede ser específico de la ubicación, a menos que el punto extremo esté respondiendo a una llamada, pero debe proporcionarse la dirección E.164 si está disponible. Se advierte que los mejores resultados se obtendrán considerando la naturaleza de los protocolos de transporte en uso.

7.11.2 AdmissionConfirm (ACF) (confirmación de admisión)

El mensaje ACF incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Será el mismo valor que fue pasado en el ARQ.

bandWidth (anchura de banda) – La máxima anchura de banda permitida para la llamada; puede ser menor que la solicitada.

callModel (modelo de llamada) – Dice al terminal si la señalización de llamada enviada en destCallSignalAddress va a un controlador de acceso o a un terminal. Un valor de **gatekeeperRouted** indica que la señalización de llamada se está pasando a través del controlador de acceso, mientras que **direct** indica que está en uso el modo llamada de punto extremo a punto extremo.

destCallSignalAddress (dirección de respuesta) – La dirección de transporte a la que se envía señalización de llamada Q.931, pero puede ser una dirección de punto extremo o de controlador de acceso según el modelo de llamada en uso.

irrFrequency (frecuencia irr) – La frecuencia, en segundos, con que el terminal enviará IRR al controlador de acceso mientras está en una llamada, incluido cuando está en retención. Si no está presente, el punto extremo no envía IRR mientras está activo en una llamada, y se cree que el controlador de acceso interrogará secuencialmente el punto extremo.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

destinationInfo (información de destino) – Dirección del canal inicial utilizada cuando se efectúa una llamada a través de una pasarela.

destExtraCallInfo (información de llamada extra de destino) – Necesario para efectuar posibles llamadas de canal adicional, es decir, para una llamada 2*64 kbit/s en el lado WAN. Sólo contendrá las direcciones E.164, y no contendrá el número del canal inicial.

destinationType (tipo de destino) – Especifica el tipo del punto extremo de destino.

remoteExtensionAddress (dirección de extensión distante) – Contiene la dirección de alias de un punto extremo llamado en los casos en que es necesaria esta información para atravesar múltiples pasarelas.

alternateEndpoints (puntos extremos alternativos) – Secuencia de alternativas de puntos extremos prioritarios para destCallSignalAddress o destinationInfo.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

TransportQOS (calidad de servicio de transporte) – El controlador de acceso puede indicar al punto extremo qué entidad se ocupa de la reserva de recursos. Si el controlador de acceso recibió TransportQOS en ARQ, incluirá entonces TransportQOS (posiblemente modificado según la implementación del controlador de acceso) en ACF.

willRespondToIRR (responderá a IRR) – VERDADERO si el controlador de acceso envía un mensaje IACK o INAK en respuesta a un mensaje IRR no solicitado cuando el campo **needsResponse** de IRR está fijado en VERDADERO.

uuiesRequested (uuie solicitadas) – El controlador de acceso puede solicitar al punto extremo que notifique al controlador de acceso los mensajes de señalización de llamada H.225.0 que el punto extremo envía o recibe si el punto extremo indicó esta capacidad en el ARQ fijando **willSupplyUUIEs** en VERDADERO. **uuiesRequested** indica el conjunto de mensajes de señalización de llamada H.225.0 que el punto extremo notificará al controlador de acceso.

language (idioma) – Indica el o los idiomas en que el usuario desea recibir anuncios y avisos. El campo contiene uno o más rótulos de idioma que satisfacen RFC 1766.

useAnnexECallSignalling (utilización de señalización de llamada según anexo E) – Si es VERDADERO, este parámetro indica que el punto extremo que recibe este mensaje utilizará exclusivamente señalización de llamada en un canal de transporte no fiable como se describe en el anexo E/H.323 para señalización de llamada hacia la dirección de señalización de llamada indicada anteriormente. Si es FALSO no utilizará el anexo E/H.323 para señalización de llamada. Si no está presente (o recibido por un controlador de acceso de versión 2 o inferior), el punto extremo puede tratar ambos métodos para compatibilidad hacia atrás como se describe en el anexo E/H.323.

NOTA – Si la confirmación de admisión se relaciona con una llamada entrante, este campo será ignorado.

7.11.3 AdmissionReject (ARJ) (rechazo de admisión)

El mensaje ARJ incluye lo siguiente:

requestSeqNum (número esencial de petición) – Será el mismo valor que fue pasado en el ARQ.

RejectReason (causa del rechazo) – Indica el motivo por el que se denegó la petición de admisión. Se señala que el campo **rejectReason** de routeCallToSCN es una elección adecuada sólo cuando el ARJ está dirigido a una pasarela de ingreso (el ARQ fue enviado por una pasarela y el **answerCall** booleano en el ARQ es FALSO). Si **rejectReason** es routeCallToSCN, el campo **rejectReason** para esta elección también incluye un número de teléfono, o lista de números telefónicos, al cual la pasarela puede redirigir la llamada en el RCC, si la pasarela soporta tal procedimiento.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

altGKInfo (información de controlador de acceso alternativo) – Información opcional sobre controladores de acceso alternativos. Si se suministra esta información, un punto extremo debe retransmitir la petición a uno de los controladores de acceso alternativos enumerados. Si un controlador de acceso alternativo rechaza la petición, el punto extremo aceptará el rechazo. Si un controlador de acceso alternativo no responde, el punto extremo puede enviar la petición a otro controlador de acceso alternativo en la lista.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

callSignalAddress (dirección de señalización de llamada) – Es la dirección de señalización de llamada del controlador de acceso devuelta cuando el motivo del rechazo es routeCallToGatekeeper.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de

integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.12 Peticiones de terminal a controlador de acceso de cambios de anchura de banda

El mensaje BRQ pide que a un punto extremo le sea concedido un cambio en la asignación de anchura de banda LAN por el controlador de acceso, que concede la petición con un BCF o la deniega con un BRJ.

El controlador de acceso puede solicitar que un punto extremo eleve o reduzca la anchura de banda en uso con un BRQ. Si la petición es de elevar la velocidad, el punto extremo puede responder con un BRJ o BCF. Si lo que se pide es una velocidad inferior, el punto extremo responderá con un BCF si la velocidad inferior es soportada, sino con BRJ.

7.12.1 BandwidthRequest (BRQ) (petición de ancho de banda)

El mensaje BRQ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del llamante. Debe ser devuelto por el llamado en cualesquiera mensajes asociados con este mensaje concreto.

endpointIdentifier (identificador de punto extremo) – Es un identificador de punto extremo que fue asignado al terminal por RCF.

conferenceID (ID de conferencia) – ID de la llamada a la que tiene que cambiarse la anchura de banda.

callReferenceValue (valor de referencia de llamada) – El CRV de la Q.931 para esta llamada; sólo tiene validez local. Es utilizado por un controlador de acceso para asociar el BRQ con una determinada llamada.

callType (tipo de llamada) – Utilizando este valor, el controlador de acceso puede intentar determinar la utilización de anchura de banda "real".

bandWidth (anchura de banda) – El NUEVO número de incrementos de 100 bits solicitado para la llamada. Es un valor absoluto que incluye sólo trenes de bits de audio y de vídeo sin contar encabezamientos ni tara.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

gatekeeperIdentifier (identificador de controlador de acceso) – Un gatekeeperIdentifier que el cliente recibió en la lista de alternateGatekeeper en el mensaje RCF proveniente del controlador de acceso cuando se registró o en un mensaje BRJ anterior. Utilizado como reserva si el controlador de acceso original no respondió a la petición o la rechazó.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de

integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

answeredCall (llamada respondida) – Fijado en VERDADERO para indicar que esta parte fue el destino original (esta parte respondió la llamada).

7.12.2 BandwidthConfirm (BCF) (confirmación de ancho de banda)

El mensaje BCF incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Debe ser el mismo valor que fue pasado en el BRQ.

bandWidth (anchura de banda) – El máximo permitido en ese momento en incrementos de 100 bits.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.12.3 BandwidthReject (BRJ) (rechazo de ancho de banda)

El mensaje BRJ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Debe ser el mismo valor que fue pasado en el BRQ.

rejectReason (motivo del rechazo) – Motivo por el que el cambio fue rechazado por el controlador de acceso.

allowedBandWidth (anchura de banda permitida) – El máximo permitido en ese momento en incrementos de 100 bits, incluida la asignación vigente.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

altGKInfo (información de controlador de acceso alternativo) – Información opcional sobre controladores de acceso alternativos. Si se suministra esta información, un punto extremo puede retransmitir la petición a uno de los controladores de acceso alternativos enumerados. Si un controlador de acceso alternativo rechaza la petición, el punto extremo aceptará el rechazo. Si un controlador de acceso alternativo no responde, el punto extremo puede enviar la petición a otro controlador de acceso alternativo que figura en la lista.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de

integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.13 Location Request messages (LRQ) (mensajes de petición de localización)

El LRQ solicita que un controlador de acceso proporcione traducción de dirección. El controlador de acceso responde con un LCF que contiene la dirección de transporte del destino, o rechaza la petición con LRJ.

7.13.1 LocationRequest (LRQ) (petición de localización)

El mensaje LRQ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del llamante. Debe ser devuelto por el llamado en cualesquiera mensajes asociados con este mensaje concreto.

endpointIdentifier (identificador de punto extremo) – Es un identificador de punto extremo que fue asignado al terminal por RCF.

destinationInfo (información de destino) – Secuencia de direcciones de alias para el terminal de destino, tales como direcciones E.164 o H323_ID. Si en un controlador de acceso se registra al menos un alias y en el mensaje LRQ no se registran dos alias a distintas personas, el controlador de acceso reconocerá la petición de ubicación en referencia a la identidad registrada. En el caso de alias en conflicto la petición de localización se rechazará con la causa AliasesInconsistent. Si el controlador de acceso no proporciona esta validación, considerará que la primera dirección registrada es el destino.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

replyAddress (dirección de respuesta) – Dirección de transporte para enviar LCF/LRQ.

sourceInfo (información de origen) – Indica el emisor del LRQ. El controlador de acceso puede utilizar esta información para decidir cómo responder al LRQ.

canMapAlias (puede copiar alias) – Si está fijado en VERDADERO indica que si el LCF resultante contiene los campos **destinationInfo**, **destExtraCallInfo** y/o **remoteExtension**, el punto extremo puede copiar esta información en los campos **destinationAddress**, **destExtraCallInfo** y **remoteExtensionAddress** del mensaje ESTABLECIMIENTO, respectivamente. Si el GK reemplaza la información de direccionamiento proveniente del LRQ y **canMapAlias** es FALSO, el controlador de acceso rechazará el LRQ.

gatekeeperIdentifier (identificador de controlador de acceso) – Un gatekeeperIdentifier que el cliente recibió en la lista de alternateGatekeeper en el mensaje RCF proveniente del controlador de acceso cuando se registró o en un mensaje LRJ anterior. Utilizado como reserva si el controlador de acceso original no respondió a la petición o la rechazó.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.13.2 LocationConfirm (LCF) (confirmación de localización)

El mensaje LCF incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Será el mismo valor que fue pasado en el LRQ.

callSignalAddress (dirección de señalización de llamada) – La dirección de transporte a la que se envía señalización de llamada Q.931; utiliza el puerto conocido o dinámico fiable, pero puede ser una dirección de punto extremo o de controlador de acceso según el modelo de llamada en uso.

rasAddress (dirección ras) – Dirección de registro, de admisiones y de situación para el punto extremo localizado.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

destinationInfo (información de destino) – Secuencia de direcciones de alias para el destino, tales como direcciones E.164 o H323_ID.

destExtraCallInfo (información de llamada extra de destino) – Contiene direcciones exteriores para múltiples llamadas.

destinationType (tipo de destino) – Especifica el tipo del punto extremo de destino.

remoteExtensionAddress (dirección de extensión distante) – Contiene la dirección de alias de un punto extremo llamado en los casos en que es necesaria esta información para atravesar múltiples pasarelas.

alternateEndpoints (puntos extremos alternativos) – Secuencia de alternativas de puntos extremos prioritarios para callSignalAddress, rasAddress, o destinationInfo.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

suportsAnnexECallSignalling – Si es VERDADERO indica que el punto extremo a que hace referencia el campo callSignallingAddress puede recibir señalización de llamada en un canal de transporte no fiable como se describe en el anexo E/H.323. Si es FALSO indica que el punto extremo no puede recibir señalización de llamada conforme al anexo E/H.323, de modo tal que no se utilizará este tipo de señalización. Si el campo no está presente, el punto extremo llamante o el controlador de acceso pueden probar ambos métodos para compatibilidad hacia atrás como se describe en el anexo E/H.323.

7.13.3 LocationReject (LRJ) (rechazo de localización)

El mensaje LRJ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Será el mismo valor que fue pasado en el LRQ.

rejectReason (motivo del rechazo) – Indica el motivo por el que se denegó la petición de localización. Si **rejectReason** es routeCallToSCN, el motivo del rechazo incluirá también un número de teléfono o lista de números telefónicos a los que la pasarela pueda redirigir la llamada en el RCC, si la pasarela soporta ese procedimiento.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

altGKInfo (información de controlador de acceso alternativo) – Información opcional sobre controladores de acceso alternativos. Si se suministra esta información, un punto extremo debe retransmitir el pedido a uno de los controladores de acceso alternativos enumerados. Si un controlador de acceso alternativo rechaza el pedido, el punto extremo aceptará el rechazo. Si un controlador de acceso alternativo no responde, el punto extremo puede enviar el pedido a otro controlador de acceso alternativo que figura en la lista.

tokens (testigos) – Se trata de algunos datos que pueden ser requeridos para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.14 Mensajes de desligamiento

7.14.1 DisengageRequest (DRQ) (petición de desligamiento)

Si se envía de un terminal a un controlador de acceso, el DRQ informa al controlador de acceso que un punto extremo está siendo abandonado. Si se envía de un controlador de acceso a un punto extremo, el DRQ obliga a una llamada a ser abandonada; dicha petición no será rehusada. El DRQ no se envía directamente entre puntos extremos.

Adviértase que DRQ no es el mismo que **ReleaseComplete**, dado que su finalidad es informar al controlador de acceso de la terminación de una llamada; el controlador de acceso puede no recibir la liberación completa si no está terminando el canal de señalización de llamada.

El mensaje DRQ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del emisor. Debe ser devuelto por el receptor en cualesquiera mensajes asociados con este mensaje concreto.

endpointIdentifier (identificador de punto extremo) – Es un identificador de punto extremo que fue asignado al terminal por RCF.

conferenceID (ID de conferencia) – ID de la llamada de la que ha de liberarse la anchura de banda.

callReferenceValue (valor de referencia de llamada) – El CRV de la Q.931 para esta llamada; sólo tiene validez local. Es utilizado por un controlador de acceso para asociar el mensaje con una determinada llamada.

disengageReason (motivo del desligamiento) – Motivo por el que fue solicitado el cambio por el controlador de acceso o el terminal.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

gatekeeperIdentifier (identificador de controlador de acceso) – Un gatekeeperIdentifier que el cliente recibió en la lista de alternateGatekeeper en el mensaje RCF proveniente del controlador de acceso cuando se registró o en un mensaje DRJ anterior. Utilizado como reserva si el controlador de acceso original no respondió a la petición o la rechazó.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

answeredCall (llamada respondida) – Fijado en VERDADERO para indicar que esta parte fue el destino original (esta parte respondió la llamada).

7.14.2 DisengageConfirm (DCF) (confirmación de desligamiento)

El mensaje DCF incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Debe ser el mismo valor que fue pasado en el DRQ.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.14.3 DisengageReject (DRJ) (rechazo de desligamiento)

DRJ es enviado por el controlador de acceso si el punto extremo es desregistrado.

El mensaje DRJ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Debe ser el mismo valor que fue pasado en el DRQ.

rejectReason (motivo del rechazo) – Motivo por el que se rechazó la petición.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

altGKInfo (información de controlador de acceso alternativo) – Información opcional sobre controladores de acceso alternativos. Si se suministra esta información, un punto extremo debe retransmitir el pedido a uno de los controladores de acceso alternativos enumerados. Si un controlador de acceso alternativo rechaza el pedido, el punto extremo aceptará el rechazo. Si un controlador de acceso alternativo no responde, el punto extremo puede enviar el pedido a otro controlador de acceso alternativo que figura en la lista.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.15 Mensajes de petición de situación

El IRQ es enviado desde un controlador de acceso a un terminal solicitando información de situación en forma de un IRR. El IRR puede también ser enviado por el terminal en un intervalo especificado en el mensaje ACF sin el recibo de un IRQ procedente del controlador de acceso. Este mensaje no debe confundirse con el mensaje SITUACIÓN (STATUS) Q.931.

Cuando un IRR no solicitado es enviado por un punto extremo a un controlador de acceso de la versión 2 o versión más alta, puede indicar en el campo **needResponse** que desea que el controlador de acceso acuse recibo del IRR. En este caso, rellena el campo **requestSeqNum** con un número distinto de 1. El controlador de acceso devuelve un mensaje IACK (acuse de recibo positivo) o bien un mensaje INAK (acuse de recibo negativo) y debe devolver el mismo número en el campo **requestSeqNum**.

7.15.1 InfoRequest (IRQ) (petición de información)

El mensaje IRQ incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del llamante. Debe ser devuelto por el llamado en todos los mensajes asociados con este mensaje concreto.

callReferenceValue (valor de referencia de llamada) – CRV de la llamada sobre la que trata la interrogación. Si es cero, este mensaje se interpreta como una petición de un IRR para cada llamada en la que el terminal está activo. Si el terminal no está activo en ninguna llamada, se enviará IRR en respuesta a un valor de referencia de llamada 0 con los campos apropiados. Si callReferenceValue es 0, el punto extremo ignorará callIdentifier. En este caso el controlador de acceso fijará callIdentifier en 0.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

replyAddress (dirección de respuesta) – Una dirección de transporte para enviar IRR, quizás no al controlador de acceso.

callIdentifier (identificador de llamada) – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue,

este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

uuiesRequested (uuie solicitadas) – El controlador de acceso puede solicitar al punto extremo que notifique al controlador de acceso los mensajes de señalización de llamada H.225.0 que el punto extremo envía o recibe si el punto extremo indicó esta capacidad en el ARQ fijando **willSupplyUUIEs** en VERDADERO. **uuiesRequested** indica el conjunto de mensajes de señalización de llamada H.225.0 que el punto extremo notificará al controlador de acceso.

7.15.2 InfoRequestResponse (IRR) (respuesta a petición de información)

El mensaje IRR incluye lo siguiente:

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

requestSeqNum (número secuencial de petición) – Contendrá el número secuencial procedente del IRR o uno para un informe no solicitado a un controlador de acceso de la versión 1. Contiene un número monotónicamente creciente (que será devuelto por el controlador de acceso en su respuesta) si **needResponse** es VERDADERO.

endpointType (tipo de punto extremo) – Proporciona información acerca del punto extremo.

endpointIdentifier (identificador de punto extremo) – Valor asignado por el controlador de acceso en el RCF.

rasAddress (dirección ras) – Dirección para registro, admisiones, etc.

callSignalAddress (dirección de señalización de llamada) – Dirección de señalización de llamada H.225.0.

endpointAlias (alias de punto extremo) – Alias para el punto extremo.

perCallInfo (información por llamada) – Información sobre cada llamada:

- **nonStandardData (datos no normalizados)** – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).
- **callReferenceValue (valor de referencia de llamada)** – CRV Q.931 de esa llamada sobre la que trata la respuesta.
- **conferenceID (ID de conferencia)** – Identificador de conferencia único.
- **originator (originador)** – Si es VERDADERO, el punto extremo interrogado fue el originador de la llamada; si es FALSO el punto extremo fue el destino de la llamada.
- **Audio** – Información sobre los canales audio.
- **Video** – Información sobre los canales vídeo.
- **data (datos)** – Información sobre los canales de datos.
- **h245** – Dirección de transporte del canal de control H.245.
- **callSignaling (señalización de llamada)** – Dirección de transporte del canal de señalización de llamada H.225.0.
- **callType (tipo de llamada)** – Proporciona información sobre la topología de las llamadas.
- **bandwidth (anchura de banda)** – Utilización actual en incrementos de 100 bit/s; incluye sólo audio y vídeo, excluidos encabezamientos y tara.
- **callModel (modelo de llamada)** – Indica que el punto extremo sabe cuál es el modelo de llamada que se utiliza.
- **callIdentifier (identificador de llamada)** – Identificador de llamada único a nivel mundial fijado por el punto extremo de origen que puede utilizarse para asociar la señalización RAS con la señalización Q.931 modificada que se utiliza en esta Recomendación.

– **tokens (testigos)** – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

– **cryptoTokens** – **Testigos** criptados.

– **substituteConfIDs (ID de conferencia de sustitución)** – Listado de todos los ConferenceIDs recibidos en los mensajes SubstituteCID H.245 pertenecientes a **perCallInfo conferenceID** de RAS original.

– **pdu (unidad de datos de protocolo)**:

  • **h323pdu** – Copia de una PDU H.225.0 y Q.931 solicitada por el controlador de acceso en **uuiesRequested** en ACF o bien en IRQ.

  • **sent (enviado)** – Fijado en VERDADERO para indicar que el punto extremo envió la **h323pdu**; fijado en FALSO para indicar que el punto extremo recibió la **h323pdu**.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

needResponse (necesidad de respuesta) – Si está fijado en VERDADERO y el controlador de acceso indicó en RCF o ACF que responderá a los IRR no solicitados (fijando **willRespondToIRR** en VERDADERO), el controlador de acceso responderá con IACK o INAK. Si el controlador de acceso no indicó en RCF ni en ACF que responderá a los IRR no solicitados (fijando **willRespondToIRR** en FALSO), ignorará la variable BOOLEANA **needResponse**.

7.15.3 InfoRequestAck (IACK) (acuse de recibo positivo de petición de información)

El mensaje IACK incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Este campo contendrá el requestSeqNum que estaba en IRR.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.15.4 InfoRequestNak (INAK) (acuse de recibo negativo de petición de información)

El mensaje INAK incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Este campo contendrá el requestSeqNum que estaba en IRR.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

nakReason (motivo de acuse de recibo negativo) – Motivo por el cual el acuse de recibo de IRR fue negativo.

altGKInfo (información de controlador de acceso alternativo) – Información opcional sobre controladores de acceso alternativos. Si se suministra esta información, un punto extremo debe retransmitir el pedido a uno de los controladores de acceso alternativos enumerados. Si un controlador de acceso alternativo rechaza el pedido, el punto extremo aceptará el rechazo. Si un controlador de acceso alternativo no responde, el punto extremo puede enviar el pedido a otro controlador de acceso alternativo que figura en la lista.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.16 Mensaje no normalizado

La estructura de un **NonStandardMessage** es la siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del emisor.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.17 Mensaje no entendido

Este mensaje se envía siempre que un punto extremo H.323 recibe un mensaje RAS que no entiende.

RequestSeqNum – Será el **requestSeqNum** del mensaje no entendido, si puede ser decodificado, si no será cero.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.18 Mensajes de disponibilidad de recursos de pasarela

La indicación disponibilidad de recurso (RAI, *resource availability indication*) es una notificación enviada por una pasarela a un controlador de acceso indicando su capacidad de llamada en esos momentos para cada protocolo de la serie H y la velocidad de datos para ese protocolo. El controlador de acceso responde con una confirmación de disponibilidad de recurso (RAC, *resource availability confirmation*) tras recibir una RAI para acusar recibo de su recepción.

7.18.1 ResourcesAvailableIndicate (RAI) (indicación de disponibilidad de recursos)

El mensaje RAI incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es un número monotónicamente creciente exclusivo del emisor. Debe ser devuelto por el receptor en cualquier respuesta asociada con este mensaje concreto.

protocolIdentifier (identificador de protocolo) – Identifica la antigüedad del punto extremo emisor.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

endpointIdentifier (identificador de punto extremo) – Un controlador de acceso asignó cadena de identidad de punto extremo.

protocols (protocolos) – Indica las velocidades de datos actuales para cada protocolo que puede ser soportado dado el estado actual del dispositivo.

almostOutOfResources (casi sin recursos) – Cuando está fijado en VERDADERO, el dispositivo alcanza su plena capacidad o se acerca a ella. Cualquier acción basada en este campo queda a juicio del fabricante. Si el dispositivo no alcanza su plena capacidad ni se acerca a ella, este campo se fijará en falso.

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.18.2 ResourcesAvailableConfirm (RAC) (confirmación de disponibilidad de recursos)

El mensaje RAC incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Debe ser el mismo valor que fue pasado en la RAI.

protocolIdentifier (identificador de protocolo) – Identifica la antigüedad del controlador de acceso aceptador.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

7.19 Temporizadores RAS y petición en curso (RIP, *request in progress*)

El cuadro 18 muestra los valores de temporización por defecto para responder a los mensajes RAS y los cómputos de reintentos subsiguientes recomendados si no se recibe una respuesta. (Estos valores pueden cambiar a medida que se vaya disponiendo de más experiencia y datos en relación con la implementación.)

Cuadro 18/H.225.0 – Valores de temporización por defecto recomendados

Mensajes RAS	Valor de temporización (segundos)	Cómputo de reintentos
GRQ	5	2
RRQ (nota 1)	3	2
URQ	3	1
ARQ	5	2
BRQ	3	2
IRQ	3	1
IRR (nota 2)	5	2
DRQ	3	2
LRQ	5	2
RAI	3	2
NOTA 1 – El valor de temporización se ha de volver a calcular en base al tiempo de vida (que puede ser indicado por el controlador de acceso en el mensaje RCF) y el número deseado de reintentos. NOTA 2 – En los casos en que se espere que el controlador de acceso responda a un IRR no solicitado con IACK o INAK, puede activarse la temporización si no se recibe ninguna respuesta al IRR.		

Si una entidad recibe una petición de una entidad de versión 2 (o posterior) para la que no puede generarse una respuesta dentro de un periodo de temporización de reintento típico, puede enviar un mensaje RIP especificando el periodo (en el campo **demora**) tras el que deberá haberse generado una respuesta. En cuanto esté disponible una respuesta, la entidad respondedora deberá enviarla sin esperar a que concluya el retardo de la RIP. Si una entidad solicitante no ha recibido una respuesta en el momento en que concluye el retardo de la RIP, volverá a enviar la petición. La entidad respondedora puede enviar una respuesta duplicada o bien otro mensaje RIP. En la figura 2 se muestra un ejemplo de intercambio de mensajes con el que se describen varios aspectos de estrategia de los reintentos.

Los vendedores han de saber que cualquier reintento repercutirá en el tiempo de establecimiento de la llamada, que debería reducirse al mínimo. Por ello, conviene que los tiempos de reintento sean breves. Para que las entidades distantes puedan anticipar tiempos de reintento típicos, a fin de decidir cuándo se envía un mensaje RIP, las entidades deberán evitar periodos de reintento inferiores a 100 ms. Para los tiempos de ida y vuelta se recomienda el cálculo exponencial y ajustes. Las entidades pueden utilizar el tiempo de ida y vuelta medido del proceso de registro RRQ/RCF para modificar una estimación inicialmente conservadora (de unos pocos segundos) a estos efectos. Las entidades pueden también utilizar el proceso de registro para intercambiar números de versiones de modo que se asegure la no utilización del mecanismo de reintentos basado en la RIP cuando participen entidades de versión 1 en la señalización.

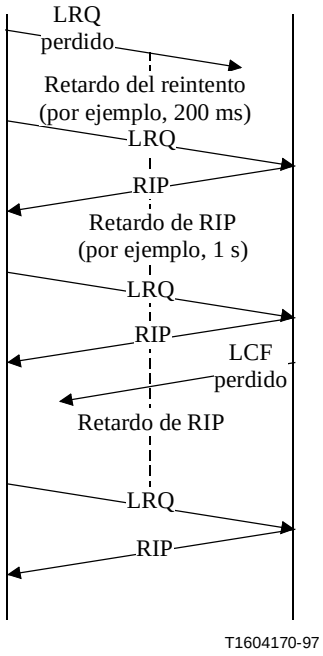


Figura 2/H.225.0 – Ejemplo de utilización del mensaje RIP

El mensaje RIP incluye lo siguiente:

requestSeqNum (número secuencial de petición) – Es el requestSeqNum de la petición que se está procesando.

nonStandardData (datos no normalizados) – Lleva información no definida en esta Recomendación (por ejemplo, datos patentados).

tokens (testigos) – Se trata de algunos datos que pueden ser necesarios para permitir la operación. Los datos se insertarán en el mensaje, si están disponibles.

cryptoTokens – **Testigos** criptados.

integrityCheckValue (valor de verificación de integridad) – Proporciona la integridad/autenticación del mensaje mejorada de los mensajes RAS. El valor de verificación de integridad sobre una base criptográfica es calculado por el emisor aplicando un algoritmo de integridad negociado y la clave secreta en todo el mensaje. Antes de calcular el integrityCheckValue, este campo deberá ser ignorado y estar vacío. Después del cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje.

delay (retardo) – Especifica el tiempo en milisegundos que esperará un punto extremo antes de tratar de efectuar un reintento. El punto extremo respondedor puede responder antes que concluya este periodo.

8 Mecanismos para mantener la calidad de servicio (QOS)

8.1 Planteamiento general e hipótesis

La calidad de servicio (QOS) de transporte de una red de paquetes incluye características tales como:

- tasa de errores de bit;
- tasa de pérdida de paquetes;
- retardo.

Cualquier señalización relacionada con la QOS de transporte (por ejemplo, una petición de reserva a un encaminador) es efectuada por el terminal cuanto antes o por el controlador de acceso en su nombre. El terminal puede desear formular algunas reservas, ya que el controlador de acceso lógicamente no puede estar cerca del terminal, ni formular peticiones relacionadas con la QOS en nombre del terminal. El modo en que el terminal o el controlador de acceso hace reservas de QOS o de anchura de banda cae fuera del alcance de esta Recomendación.

Los informes de emisor y de receptor de RTPC serán el medio por el que se evaluará la QOS.

Hay dos tipos de retraso relacionado con la congestión que podrían medirse:

- Aumentos de corta duración del retardo que producirían una reducción perceptible, pero no molesta, de la velocidad de trama.
- Un aumento general del retardo debido a la congestión de la red de paquetes en el tiempo de manera que sea de utilidad un mecanismo basado en la realimentación.

Esencialmente, las ráfagas de corta duración son aproximadas mediante ocultación de errores, y una congestión de más larga duración es aproximada mediante reducción de la carga multimedios. Se adopta la hipótesis de que todos los terminales multimedios red de paquetes son terminales H.323, y todos intentarán reducir la utilización de la red de paquetes a medida que la congestión aumenta, más que "robarse" anchura de banda entre sí.

Los errores de bit en una red de paquetes son corregidos por lo general en una capa inferior, o dan lugar a pérdida de paquetes, por lo que no se consideran en esta cláusula.

La pérdida de paquetes exige que el receptor pueda compensar los paquetes perdidos de una manera que oculte los errores en la máxima medida posible. Para datos y control, se utiliza retransmisión en la capa de transporte. Para audio y vídeo, la retransmisión queda en estudio.

Un determinado nivel de QOS de transporte produce un nivel de la QOS de audio/vídeo percibida por el usuario que es función en parte de la efectividad de los métodos utilizados para superar problemas de QOS de transporte.

8.2 Utilización del RTCP al medir la calidad de servicio (QOS)

8.2.1 Informes de emisor

El informe de emisor cumple tres fines principales:

1) Permite la sincronización de múltiples trenes RTP, tales como audio y vídeo.
2) Permite al receptor conocer la velocidad de datos y la velocidad de paquetes esperadas.
3) Permite al receptor medir la distancia en tiempo al emisor.

De estos tres fines, 1) es el más pertinente para esta Recomendación. Los fabricantes pueden utilizar los informes del remitente de otros modos a su discreción.

El campo pertinente para la sincronización de trenes es la indicación de tiempo RTP y la indicación de tiempo NTP en el informe de emisor del RTCP. La indicación de hora NTP (si está disponible) indica el tiempo "de reloj", que corresponde a la indicación de tiempo que tiene las mismas unidades y desplazamiento aleatorio de la indicación de tiempo de captura RTP en los paquetes de medios.

8.2.2 Informes de receptor

Se utilizan cuatro partes de los informes de receptor en esta Recomendación para medir la QOS:

1) Fracción perdida.

2) Los paquetes acumulativos perdidos.

3) El número secuencial más alto extendido recibido.

4) Fluctuación entre llegadas.

Los apartados 2) y 3) se utilizan para calcular el número de paquetes perdidos desde el informe de receptor anterior. Esta medida puede tomarse como una medida a corto plazo de la congestión de la red de paquetes. Véase en A.6.3.4 un ejemplo de cálculo. Si esta velocidad de pérdida sobrepasa un valor fijado por el fabricante, el terminal H.225.0 debería reducir las velocidades de medios en el lado red de paquetes de acuerdo con los procedimientos expuestos más adelante en 8.4. Si el apartado 1) sobrepasa un valor fijado por el fabricante, puede también ser conveniente ejercer una acción correctiva.

Si el intervalo entre informes de receptor sobrepasa un valor fijado por las especificaciones del fabricante, los terminales H.323 deben utilizar el apartado 1) como un indicador de congestión grave que exige reducción de velocidad de medios en el lado red de paquetes.

El apartado 4) debe utilizarse como una indicación de congestión inminente. Si la fluctuación entre llegadas aumenta en tres informes de receptor consecutivos, el terminal emisor H.323 debe ejercer acción correctiva.

8.3 Procedimientos de fluctuación de audio/vídeo

La Recomendación H.245 proporciona instrucciones y procedimientos para indicaciones de retardo de ida y vuelta utilizando **petición de retardo de ida y vuelta (RoundTripDelayRequest)** y **respuesta de retardo de ida y vuelta (RoundTripDelayResponse)**. En una llamada multipunto, el MC responde a una petición del punto extremo. RTCP contiene un método de calcular retardos de ida y vuelta basándose en los mensajes de informe de emisor y de informe de receptor. Adviértase que la magnitud que se mide en cada caso no es la misma, por lo que no hay contradicción en utilizar ambos métodos para medir la fluctuación.

Véase en 6.2.5/H.323 un análisis de la posible forma de utilizar la señalización de nivel H.245 para reducir opcionalmente los retardos relacionados con la fluctuación.

8.4 Procedimientos de sesgo de audio/vídeo

Véase en 6.2.6/H.323 un análisis sobre la forma de utilizar la señalización de nivel H.245 para limitar el sesgo entre diferentes canales lógicos.

8.5 Procedimientos para mantener la calidad de servicio (QOS)

Existen algunos métodos para que la pasarela/el terminal H.323 responda a un aumento en la pérdida de paquetes o en la fluctuación entre llegadas en el receptor de extremo distante. Estos métodos pueden agruparse en los que son apropiados para una rápida respuesta a un problema de corta duración, tal como un paquete perdido o retardado, o los que son apropiados para una respuesta a un problema de mayor duración tal como el crecimiento en la congestión en la red de paquetes. Adviértase que estos métodos no pretenden mantener la actual calidad de servicio, sino más bien

proveer una degradación ordenada del servicio. Se observaron las siguientes prioridades, de manera que, si aparecen, los medios se degradarán en el orden siguiente: Vídeo, Datos, Audio, Control.

Respuestas a corto plazo:

- Reducción de la velocidad de trama durante un breve periodo de tiempo, lo que puede dar lugar a que la pasarela H.323 envíe tramas de relleno H.261 adicionales en el sentido red de paquetes → WAN para compensar el subflujo de paquetes.
- Reducción de la velocidad de paquetes por conmutación al modo opcional, en el que el audio/vídeo se mezclan en un paquete (queda en estudio).
- La velocidad de paquetes puede también reducirse mediante el uso de fragmentación MB del tren de vídeo.

Respuestas a más largo plazo:

- Reducción de la velocidad binaria de medios (por ejemplo, conmutación de 384 kbit/s a 256 kbit/s). Esto puede exigir una simple instrucción al codificador en un terminal, o el uso de una función reductora de velocidad en la pasarela H.323. Estos cambios se señalizan mediante instrucciones **control de flujo (FlowControl)** H.245, o mediante señalización de canal lógico, según convenga.
- Desactivación de medios de menor importancia (por ejemplo, desactivación de vídeo para permitir un mayor volumen de tráfico T.120).
- Devolución de una señal de ocupado (ocupado adaptativa) al receptor como una indicación de congestión de la red de paquetes. Ésta puede combinarse con la desactivación de un medio, o de incluso todos los medios que no sean el puerto de transporte de control. Ocupado adaptativo es señalizado mediante un valor de causa Q.931 en **Liberación completa (Release Complete)**.

Debe señalarse que responder a fluctuación entre llegadas en un trayecto multiencaminadores en el que llegan un gran porcentaje de paquetes deteriorados resulta difícil. Puede resultar imposible distinguir la fuente de fluctuación de otras fuentes, o basar la estrategia de base de recuperación tras errores en la fluctuación medida. Sin embargo, la pérdida de paquetes es cuantificable e inambigua.

8.6 Control de eco

El control del eco acústico es competencia del terminal H.323. En general, dado el retardo que interviene en la compresión de vídeo/audio, se supone que todos los terminales H.320, H.323 y H.324 tienen alguna forma de control de eco (compensación o conmutación).

Sin embargo, cuando el terminal H.323 se utiliza para llamar a un teléfono de la RTGC, suele darse el caso de que el teléfono RTGC no dispone de control de eco. Así, el usuario del terminal H.323 puede oír retorno de eco acústico procedente del lado RTGC. Este retorno de eco acústico puede minimizarse mediante el uso de un teléfono de altavoz con control de eco, o el uso de un microteléfono o auriculares. Los fabricantes pueden añadir pérdida al trayecto de audio cuando un terminal H.323 está conectado a un teléfono POTS de la RTGC.

El control del eco híbrido (de dos a cuatro hilos) corresponde a la pasarela H.323.

ANEXO A

RTP/RTCP

El lector debe advertir que todas las referencias de este anexo corresponden a una bibliografía, y no son normativas, con excepción de [A-10] a ISO/CEI 10646-1, que también aparece en la cláusula de referencias de esta Recomendación. En algunos casos aparecerá una referencia al apéndice I; dichas referencias sólo tienen carácter informativo. Todos los detalles necesarios para implementar la Recomendación H.323 y esta Recomendación están contenidos en este anexo y en otros anexos y Recomendaciones o Normas Internacionales relacionados publicados por el UIT-T o la ISO.

El lector debe advertir que este anexo no es la especificación completa y primaria de RTP/RTCP; por favor véase el apéndice I para esta referencia informativa. Este anexo ha sido formulado solamente para utilización con la Recomendación H.323 y esta Recomendación.

Los lectores deben también advertir que la terminología utilizada en este anexo difiere algo de la utilizada en la Recomendación H.323 y esta Recomendación, de acuerdo con el cuadro A.1.

Cuadro A.1/H.225.0 – Correspondencia terminológica

Término H.323 y H.225.0	Término del anexo A (RTP/RTCP)
tren de medios	datos
dirección de transporte	dirección de transporte
dirección de red de paquetes	dirección de red
identificador de TSAP	puerto
anexo A	especificación o documento
Deberá	Debe
Debería	Debería

Debe además señalarse que los "traductores" y "mezcladores" no forman parte del sistema H.323. Los puntos extremos H.323, tales como pasarelas y MCU, tienen algunas de las características de los traductores y mezcladores, por lo que este texto se ha conservado como una guía para el implementador. Sin embargo, la incorporación de traductores y mezcladores no forma parte de la H.323, y estas subcláusulas se considerarán informativas.

Por último, se recuerda a los implementadores que implementen el RTP solamente como se describe en esta Recomendación, incluidos los anexos A, B y C, que contienen detalles y aclaraciones de interés para la H.323/H.225.0. En todos los casos, el texto de esta Recomendación tendrá precedencia sobre el texto de los anexos A, B o C.

A.1 Introducción

Esta especificación especifica el protocolo de transporte en tiempo real (RTP, *real-time transport protocol*) que proporciona servicios de entrega extremo a extremo de datos con características en tiempo real, tales audio y vídeo interactivos. Estos servicios incluyen identificación de tipo de cabida útil, numeración secuencial, indicación de tiempo y supervisión de entrega. Las aplicaciones suelen hacer pasar el RTP encima del UDP para hacer uso de sus servicios de multiplexación y de suma de control; ambos protocolos contribuyen con partes de la funcionalidad del protocolo de transporte. Sin embargo, el RTP puede utilizarse con otros protocolos de red o de transporte subyacentes adecuados (véase A.10, RTP sobre los protocolos de red y de transporte). RTP soporta la

transferencia de datos a múltiples destinos utilizando distribución multidifusión si es proporcionada por la red subyacente.

Adviértase que el propio RTP no proporciona ningún mecanismo para asegurar la entrega en su momento oportuno o proporcionar otras garantías de calidad de servicio, sino que confía en servicios de capa inferior para hacerlo. No garantiza la entrega ni impide la entrega en otro orden, ni supone que la red subyacente es fiable y entrega los paquetes en secuencia. Los números secuenciales incluidos en el RTP permiten al receptor reconstruir la secuencia de paquetes del remitente, pero los números de secuencia podrían también ser utilizados para determinar la ubicación adecuada de un paquete, por ejemplo, en codificación de vídeo, sin decodificar necesariamente los paquetes en secuencia.

Aunque el RTP está primordialmente diseñado para satisfacer las necesidades de conferencias multimedios de múltiples participantes, no se limita a esa aplicación determinada. El almacenamiento de datos continuos, la simulación distribuida interactiva, el distintivo identificador activo, y las aplicaciones de control y de medición pueden también encontrar aplicable el RTP.

Esta Recomendación define el RTP, compuesto de dos partes estrechamente vinculadas:

- El protocolo de transporte en tiempo real (RTP), para transportar datos que tienen propiedades de tiempo real.
- El protocolo de control del RTP (RTCP), para supervisar la calidad de servicio y transmitir información sobre los participantes en una sesión en curso. Este último aspecto del RTCP puede ser suficiente para sesiones "menos estrictamente controladas", es decir, cuando no hay ningún control ni establecimiento de participación explícito, pero no está necesariamente destinado a soportar todos los requisitos de comunicación de control de una aplicación. Esta funcionalidad puede ser totalmente o parcialmente asumida por un protocolo de control de sesión separado, que cae fuera del alcance de esta Recomendación.

El RTP representa un nuevo estilo de protocolo que sigue los principios de entramación de niveles de aplicación y de procesamiento de capas integrado propuesto por Clark y Tennenhouse [A-1]. Es decir, se pretende que el RTP sea maleable para proporcionar la información requerida por una determinada aplicación y a menudo estará integrada en el procesamiento de la aplicación en lugar de implementarse como una capa separada. El RTP es un marco de protocolo que deliberadamente está incompleto. Esta Recomendación especifica las funciones que se espera sean comunes a lo largo de todas las aplicaciones para las que el RTP sería apropiado. A diferencia de los protocolos convencionales en los que podrían acomodarse funciones adicionales haciendo el protocolo más general o añadiendo un mecanismo de opción que exigiría análisis sintáctico, se pretende que el RTP se adapte mediante modificaciones y/o adiciones a los encabezamientos que se necesiten. Se incluyen ejemplos en A.5.3, Modificaciones específicas del perfil en el encabezamiento RTP.

Por tanto, además de esta Recomendación, una especificación completa del RTP para una determinada aplicación exigiría uno o más documentos acompañantes (véanse los anexos B y C):

- Un documento de especificación de perfil, que define un conjunto de códigos de tipo cabida útil y su correspondencia con formatos de cabida útil (por ejemplo, codificaciones de medios). Un perfil puede también definir extensiones o modificaciones del RTP que sean específicas de una determinada clase de aplicaciones. Una aplicación suele operar bajo un solo perfil. En el anexo B puede verse un perfil para datos de audio y de vídeo.
- Documentos de especificación de formato de cabida útil, que definen cómo una determinada cabida útil, tal como una codificación de audio o vídeo ha de ser transportada en el RTP. Véase el anexo C.

Varias aplicaciones RTP, tanto experimentales como comerciales, ya han sido implementadas a partir de proyectos de especificaciones. Entre estas aplicaciones se hallan el audio y el vídeo, junto con herramientas de diagnóstico tales como monitores de tráfico. Los usuarios de estas herramientas se cuentan por millares. Sin embargo, la actual Internet no puede aún soportar la demanda potencial

completa de servicios en tiempo real. Los servicios de gran anchura de banda que utilizan RTP, como es el vídeo, podrían degradar seriamente la calidad de servicio de otros servicios. Por tanto, los implementadores deberían adoptar precauciones adecuadas para limitar la utilización accidental de anchura de banda. La documentación de la aplicación debe describir las limitaciones y el posible impacto operacional de los servicios de gran anchura de banda en tiempo real en los servicios de Internet y en otros servicios de red.

A.2 Ejemplos de utilización del RTP

En las subcláusulas que siguen se describen algunos aspectos del uso del RTP. Los ejemplos se eligieron para ilustrar el funcionamiento básico de aplicaciones que utilizan el RTP, no para limitar las posibilidades de utilizar el RTP. En esos ejemplos, RTP se transporta encima de IP y UDP, y sigue los convenios establecidos por el perfil para audio y vídeo especificados en el anexo B.

A.2.1 Audioconferencia multidifusión simple

Un grupo de trabajo del IETF se reúne para examinar el último proyecto de protocolo, utilizando los servicios multidifusión IP de Internet para comunicaciones de voz. Mediante algún mecanismo de asignación, la presidencia del grupo de trabajo obtiene una dirección de grupo multidifusión y un par de puertos. Un puerto se utiliza para datos de audio y el otro para paquetes de control (RTCP). Esta información de dirección y de puertos es distribuida a los participantes previstos. Si se desea privacidad, los paquetes de datos y de control pueden criptarse como se especifica en la Recomendación H.323. La aplicación de conferencia audio utilizada por cada participante en la conferencia envía datos de audio en pequeños trozos, por ejemplo, de 20 ms de duración. Cada trozo de datos de audio está precedido por un encabezamiento RTP; el encabezamiento y los datos RTP están a su vez contenidos en un paquete UDP. El encabezamiento RTP indica qué tipo de codificación de audio (tal como MIC, MICDA o LPC) está contenido en cada paquete de manera que los emisores puedan cambiar la codificación durante una conferencia, por ejemplo, para acoger a un nuevo participante que está conectado a través de un enlace de pequeña anchura de banda o reaccionar a indicaciones de congestión de red.

Internet, como otras redes de paquetes, en ocasiones pierde y reordena paquetes y los retarda en cantidades de tiempo variables. Para hacer frente a estas degradaciones, el encabezamiento de RTP contiene información de temporización y un número secuencial que permite a los receptores reconstruir la temporización producida por la fuente, por lo que en este ejemplo, los trozos de audio se reproducen uno tras otro por el altavoz cada 20 ms. Esta reconstrucción de temporización se efectúa separadamente para cada fuente de paquetes RTP en la conferencia. El número secuencial puede también ser utilizado por el receptor para estimular cuántos paquetes se han perdido.

Como los miembros del grupo de trabajo se incorporan a la conferencia y la abandonan durante la misma, resulta útil saber quiénes están participando en todo momento y si están recibiendo bien los datos de audio. A tal fin, cada ejemplar de la aplicación de audio en la conferencia multidifunde periódicamente un informe de recepción, más el nombre de su usuario en el puerto (de control) RTCP. El informe de recepción indica lo bien que se está recibiendo al orador y puede utilizarse para controlar codificaciones adaptativas. Además del nombre de usuario, puede también incluirse otra información identificadora sujeta a los límites de la anchura de banda de control. Un puesto envía el paquete RTCP BYE (véase A.6.5, BYE: paquete RTCP de despedida) cuando abandona la conferencia.

A.2.2 Audioconferencia y videoconferencia

Si se utilizan ambos medios de audio y vídeo en una conferencia, se transmiten como paquetes RTCP de sesiones RTP separadas para cada medio que utilice dos pares de puertos UDP diferentes y/o direcciones de multidifusión. No hay ningún acoplamiento directo al nivel RTP entre las sesiones de audio y de vídeo, salvo que un usuario que participe en ambas sesiones debe utilizar el mismo

nombre distinguido (canónico) en los paquetes RTCP en ambas, de manera que puedan asociarse las sesiones.

Un incentivo para esta separación es permitir que algunos participantes en la conferencia reciban sólo un medio si así lo deciden. Se dan más explicaciones en A.5.2, Multiplexación de sesiones RTP. Pese a la separación, la reproducción sincronizada del audio y el vídeo de una fuente pueden obtenerse utilizando información de temporización transportada en los paquetes RTCP para ambas sesiones.

A.2.3 Mezcladores y traductores

Hasta ahora hemos supuesto que todos los puestos desean recibir datos de medios en el mismo formato. Sin embargo, esto no siempre puede resultar apropiado. Considérese el caso en que los participantes de una zona están conectados mediante un enlace de baja velocidad a la mayoría de los participantes de la conferencia, quienes disfrutan de acceso de red a alta velocidad. En lugar de obligar a todos a utilizar una anchura de banda menor, codificación de audio de calidad reducida, puede disponerse un relé de nivel RTP denominado mezclador cerca del área de baja anchura de banda. Este mezclador resincroniza los paquetes de audio entrantes para reconstruir el espaciamiento de 20 ms constante generado por el emisor, mezcla estos trenes de audio reconstruidos en un solo tren, traduce la codificación a una anchura de banda inferior y remite el tren de paquetes de anchura de banda inferior a través del enlace de baja velocidad. Estos paquetes podrían ser unidifundidos a un único destinatario o multidifundidos en una dirección diferente a múltiples destinatarios. El encabezamiento RTP incluye un medio para que los mezcladores identifiquen las fuentes que han contribuido a un paquete mixto para que pueda proporcionarse indicación correcta del hablante en todos los receptores.

Algunos de los participantes previstos en la conferencia de audio pueden conectarse con enlaces de gran anchura de banda, pero no podrían ser directamente alcanzables mediante multidifusión IP. Por ejemplo, podrían hallarse detrás de un cortafuego a nivel de aplicación que no deje pasar paquetes IP. Para estos puestos, el mezclado podría no ser necesario, en cuyo caso puede utilizarse otro tipo de relé a nivel RTP denominado traductor. Se instalan dos traductores, uno a cada lado del cortafuego, con el del lado exterior encauzando todos los paquetes multidifusión recibidos a través de una conexión segura al traductor interior al cortafuego. El traductor interior al cortafuego los envía de nuevo como paquetes multidifusión a un grupo multidifusión restringido a la red interna del puesto.

Pueden diseñarse mezcladores y traductores para una variedad de fines. Un ejemplo es un mezclador de vídeo que escala las imágenes de distintas personas en trenes de vídeo separados y las compone en un tren de vídeo para simular una escena de grupo. Otros ejemplos de traducción incluyen la conexión de un grupo de invitados hablando solo IP/UDP a un grupo de invitados que entienden sólo ST-II, o la traducción de codificación paquete a paquete de trenes de vídeo a partir de fuentes individuales sin resincronización ni mezclado. Los detalles del funcionamiento de los mezcladores y traductores se indican en A.7, Traductores y mezcladores RTP.

A.3 Definiciones

A.3.1 cabida útil de protocolo de transporte en tiempo real: Los datos transportados por el RTP en un paquete, por ejemplo, muestras de audio o datos de vídeo comprimidos. El formato y la interpretación de la cabida útil caen fuera del alcance de esta Recomendación.

A.3.2 paquete de protocolo de transporte en tiempo real: Paquete de datos compuesto por el encabezamiento RTP fijo, una posible lista vacía de fuentes contribuyentes (véase a continuación), y los datos de cabida útil. Algunos protocolos subyacentes pueden exigir un encapsulado del paquete RTP a definir. Un paquete del protocolo subyacente suele contener un solo paquete RTP, pero varios paquetes RTP pueden estar contenidos si lo permite el método de encapsulado (véase A.10, RTP sobre los protocolos de red y de transporte).

A.3.3 paquete de protocolo de control de transporte en tiempo real: Paquete de control compuesto de una parte encabezamiento fija similar a la de los paquetes de datos RTP, seguida por elementos estructurados que varían dependiendo del tipo de paquete RTCP. Los formatos se definen en A.6, Protocolo de control RTP-RTCP. Suelen enviarse múltiples paquetes RTCP juntos como un paquete RTCP compuesto en un único paquete del protocolo subyacente; esto lo permite el campo de longitud del encabezamiento fijo de cada paquete RTCP.

A.3.4 puerto: La "abstracción que los protocolos de transporte utilizan para distinguir entre múltiples destinos dentro de un determinado computador principal. Los protocolos TCP/IP identifican puertos utilizando enteros positivos pequeños" [A-2]. Los selectores de transporte (TSEL, *transport selector*) utilizados por la capa de transporte de OSI son equivalentes a puertos. RTP depende del protocolo de capa inferior para proporcionar algún mecanismo tal como puertos para multiplexar los paquetes RTP y RTCP de una sesión.

A.3.5 dirección de transporte: Combinación de una dirección de red y un puerto que identifica un punto extremo de nivel de transporte, por ejemplo, una dirección IP y un puerto UDP. Los paquetes se transmiten de una dirección de transporte de origen a una dirección de transporte de destino.

A.3.6 sesión de protocolo de transporte en tiempo real: La asociación entre un conjunto de participantes que se comunican con RTP. Para cada participante, la sesión es definida por un determinado par de direcciones de transporte de destino (una dirección de red más un par de puertos para RTP y RTCP). El par de direcciones de transporte de destino puede ser común para todos los participantes, como ocurre en la multidistribución IP, o puede ser diferente para cada uno, como en el caso de direcciones de red unidistribución y puertos. En cada sesión multimedios cada medio es transportado en una sesión RTP separada con sus propios paquetes RTCP. Las sesiones RTP múltiples se distinguen por diferentes pares de números de puertos y/o diferentes direcciones multidifusión.

A.3.7 fuente de sincronización (SSRC, *synchronization source*): La fuente de un tren de paquetes RTP identificados por un identificador de SSRC numérico de 32 bits transportado en el encabezamiento RTP de manera que no sea independiente de la dirección de red. Todos los paquetes procedentes de una fuente de sincronización forman parte del mismo espacio de temporización y de número secuencial, por lo que un receptor agrupa paquetes por fuente de sincronización para su reproducción. Ejemplos de fuentes de sincronización son el emisor de un tren de paquetes derivado de una fuente de señal, tal como un micrófono o una cámara, o un mezclador RTP (véase más adelante). Una fuente de sincronización puede cambiar su formato de datos, por ejemplo, codificación de audio, en el tiempo. El identificador de SSRC es un valor aleatoriamente elegido destinado a ser globalmente exclusivo dentro de una determinada sesión RTP (véase A.8, Asignación y utilización de identificadores de SSRC). Un participante no necesita utilizar el mismo identificador de SSRC para todas las sesiones RTP de una sesión multimedios; la vinculación de los identificadores de SSRC se proporciona mediante RTCP (véase A.6.4.1, CNAME: Elemento SDES identificador de punto extremo canónico). Si un participante genera múltiples trenes en una sesión RTP, por ejemplo, desde cámaras de vídeo separadas, cada uno debe ser identificado por un SSRC diferente.

A.3.8 fuente contribuyente (CSRC, *contributing source*): Fuente de un tren de paquetes RTP que ha contribuido al tren combinado producido por un mezclador RTP (véase más adelante). El mezclador inserta una lista de los identificadores de SSRC de las fuentes que han contribuido a la generación de un determinado paquete en el encabezamiento RTP de ese paquete. Esta lista se denomina la lista de CSRC. Un ejemplo de aplicación es la audioconferencia, en la que un mezclador indica a todos los hablantes cuyo discurso se combinó para producir el paquete saliente, permitiendo al receptor indicar al hablante en ejercicio, aun cuando todos los paquetes contienen el mismo identificador SSRC (el del mezclador).

A.3.9 sistema de extremo: Aplicación que genera el contenido a enviar en paquetes RTP y/o consume el contenido de paquetes RTP recibidos. Un sistema de extremo puede actuar como una o más fuentes de sincronización en una determinada sesión RTP, pero sólo suele haber una.

A.3.10 mezclador: Sistema intermedio que recibe paquetes RTP de una o más fuentes, posiblemente cambia el formato de datos, combina los paquetes de alguna manera y remite entonces un nuevo paquete RTP. Dado que la temporización entre múltiples fuentes de entrada no estará generalmente sincronizada, el mezclador hará ajustes de temporización entre los trenes y generará su propia temporización para el tren combinado. Así, todos los paquetes originarios de un mezclador se identificarán como paquetes que tienen el mezclador como su fuente de sincronización.

A.3.11 traductor: Sistema intermedio que remite paquetes RTP con su identificador de fuente de sincronización intacto. Ejemplos de traductores son los dispositivos que convierten codificaciones sin mezclado, replicadores de multidifusión a unidifusión y filtros de nivel aplicación en cortafuegos.

A.3.12 monitor: Aplicación que recibe paquetes RTCP enviados por participantes en una sesión RTP, en particular los informes de recepción, y estima la calidad de servicio vigente para supervisión de distribución, diagnosis de averías y estadísticas a largo plazo. La función monitor es posible que se incorpore en la aplicación (o aplicaciones) que participa(n) en la sesión, pero puede también ser una aplicación separada que de otro modo no participe y no envíe ni reciba los paquetes de datos RTP. Éstos se llaman monitores de tercera parte.

A.3.13 medios que no son del protocolo de transporte en tiempo real: Protocolos y mecanismos que pueden ser necesarios además del RTP para proporcionar un servicio utilizable. En particular, para las conferencias multimedios, una aplicación de control de conferencia puede distribuir direcciones multidifusión y claves para criptación, negociar el algoritmo de criptación a utilizar, y definir correspondencias dinámicas entre los valores de tipo de cabida útil RTP y los formatos de cabida útil que representan para formatos que no tienen un valor de tipo de cabida útil predefinido. En aplicaciones simples, puede también utilizarse el correo electrónico o una base de datos de conferencia. La especificación de dichos protocolos y mecanismos cae fuera del alcance de esta Recomendación.

A.4 Orden, alineación y formato horario de los bytes

Todos los campos de enteros son transportados en el orden de bytes de la red, es decir, primero el byte (octeto) más significativo. Este orden de los bytes suele conocerse como el del gran final. El orden de transmisión se describe en detalle en [A-3]. A menos que se indique otra cosa, las constantes numéricas están en base decimal (base 10).

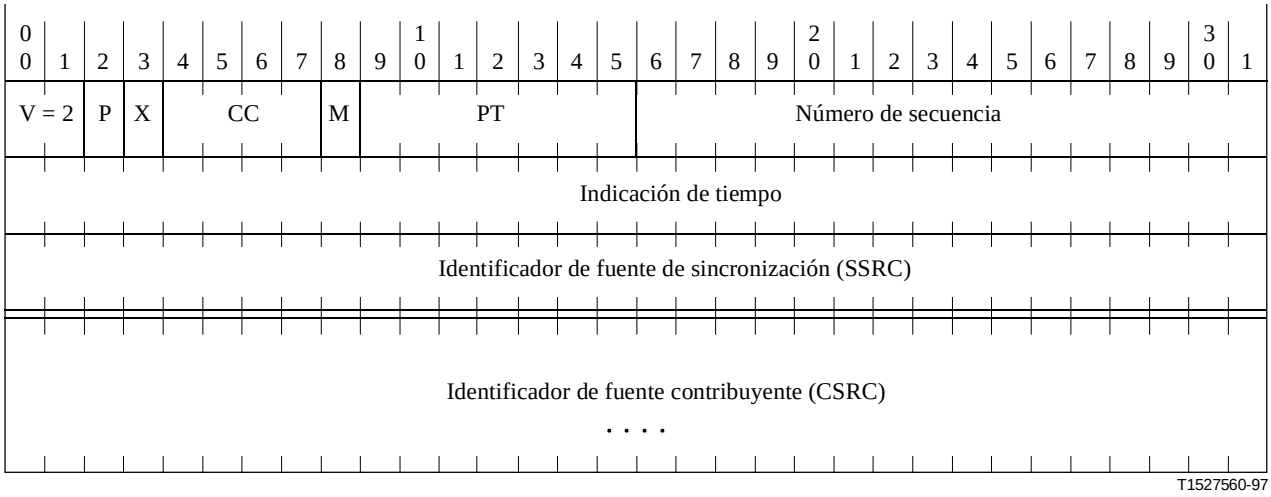
Todos los datos de encabezamiento están alineados en su longitud alineal, es decir, los campos de 16 bits están alineados en desplazamientos pares, los campos de 32 bits están alineados en desplazamientos divisibles por cuatro, etc. Los octetos designados como relleno tienen el valor cero.

El tiempo de reloj (tiempo absoluto) se representa utilizando el formato del protocolo de tiempos de la red (NTP, *network time protocol*), que se indica en segundos con relación a las 0h UTC del 1 de enero de 1900 [A-4]. La indicación de tiempo NTP de resolución completa es un número de punto fijo sin signo de 64 bits, con la parte entera en los primeros 32 bits y la parte fraccionaria en los últimos 32 bits. En algunos campos en los que es apropiada una representación más compacta, sólo se utilizan los 32 bits centrales; es decir, los 16 bits inferiores de la parte entera y los 16 bits superiores de la parte fraccionaria. Los 16 bits superiores de la parte entera deben determinarse independientemente.

A.5 Protocolo de transferencia de datos RTP

A.5.1 Campos de encabezamiento fijo RTP

El encabezamiento RTP tiene el siguiente formato:



Los doce primeros octetos están presentes en cada paquete RTP, mientras que la lista de identificadores de CSRC sólo está presente cuando ha sido insertada por un mezclador. Los campos tienen el siguiente significado:

versión (V): 2 bits. Este campo identifica la versión del RTP. La versión definida por esta especificación es dos (2). (El valor 1 es utilizado por el primer proyecto de versión de RTP y el valor 0 es utilizado por el protocolo inicialmente implementado en la herramienta de audio "vat".)

relleno (P, *padding*): 1 bit. Si el bit de relleno está fijado, el paquete contiene uno o más octetos de relleno adicionales al final que no forman parte de la cabida útil. El último octeto del relleno contiene una cuenta de cuántos octetos de relleno deben ser ignorados. El relleno puede ser necesitado por algunos algoritmos de criptación con tamaños de bloque fijos o para transportar varios paquetes RTP en una unidad de datos de protocolo de capa inferior.

extensión (X, *extension*): 1 bit. Si el bit de extensión está fijado, el encabezamiento fijo va seguido exactamente por una extensión de encabezamiento, por un formato definido en A.5.3, Modificaciones específicas del perfil en el encabezamiento RTP.

cuenta de CSRC (CC, *CSRC count*): 4 bits. La cuenta de CSRC contiene el número de identificadores de CSRC que siguen al encabezamiento fijo.

marcador (M, *marker*): 1 bit. La interpretación del marcador está definida por un perfil. Está destinado a permitir que eventos significativos tales como las fronteras de tramas estén marcados en el tren de paquetes. Un perfil puede definir bits marcadores adicionales o especificar que no hay ningún bit marcador cambiando el número de bits en el campo de tipo de cabida útil (véase A.5.3, Modificaciones específicas del perfil en el encabezamiento RTP).

tipo de cabida útil (PT, *payload type*): 7 bits. Este campo identifica el formato de la cabida útil RTP y determina su interpretación por la aplicación. Un perfil especifica una correspondencia estática por defecto de los códigos de tipo de cabida útil a formatos de cabida útil. Pueden definirse dinámicamente códigos adicionales de tipo de cabida útil por medios no RTP (véase A.3, Definiciones). En el anexo B se especifica un conjunto inicial de correspondencias por defecto para audio y vídeo. Un emisor RTP emite un único tipo de cabida útil RTP en cualquier momento dado; este campo no está destinado a multiplexar trenes de medios separados (véase A.5.2, Multiplexación de sesiones RTP).

número secuencial: 16 bits. El número secuencial aumenta en uno por cada paquete de datos RTP enviado, y puede ser utilizado por el receptor para detectar pérdida de paquetes y restablecer la secuencia de paquetes. El valor inicial del número secuencial es aleatorio (impredecible) para hacer los ataques al texto claro conocido más difíciles mediante criptación, aun si la propia fuente no cripta, ya que los paquetes pueden pasar por un traductor que sí lo hace. Las técnicas para elegir números impredecibles se tratan en [A-5].

indicación de tiempo (hora): 32 bits. La indicación de tiempo refleja el instante de muestreo del primer octeto del paquete de datos RTP. El instante de muestreo debe derivarse de un reloj que incrementa monotónicamente y linealmente en el tiempo para permitir la sincronización y los cálculos de fluctuación (véase A.6.3.1, SR: Paquete RTCP de informe de emisor). La resolución del reloj debe ser suficiente para la exactitud de sincronización deseada y para medir la fluctuación de llegada de paquetes (una indicación por trama de vídeo no suele ser suficiente). La frecuencia de reloj depende del formato de los datos transportados como cabida útil, y se especifica estáticamente en la especificación de formato de perfil o de cabida útil que define el formato, o puede especificarse dinámicamente para formatos de cabida útil definidos a través de medios no RTP. Si los paquetes RTP son generados periódicamente, ha de utilizarse el instante de muestreo nominal determinado a partir del reloj de muestreo, y no una lectura del reloj del sistema. Por ejemplo, para audio a tarifa fija, el reloj de indicación de tiempo probablemente aumentaría en uno para cada periodo de muestreo. Si una aplicación de audio lee bloques que cubren 160 periodos de muestreo desde el dispositivo de entrada, la indicación de tiempo aumentaría en 160 para cada uno de dichos bloques, independientemente de si el bloque es transmitido en un paquete o abandonado como un bloque de silencio.

El valor inicial de la indicación de tiempo es aleatorio, como en el número secuencial. Varios paquetes RTP consecutivos pueden tener iguales indicaciones de tiempo si son (lógicamente) generados a la vez, por ejemplo, pertenecen a la misma trama de vídeo. Los paquetes RTP consecutivos pueden contener indicaciones de tiempo que no sean monotónicas si los datos no se transmiten en el orden en que se muestrearon, como ocurre en el caso de tramas de vídeo interpoladas por MPEG. (Los números secuenciales de los paquetes transmitidos seguirán siendo monotónicos.)

SSRC: 32 bits. El campo SSRC identifica la fuente de sincronización. Este identificador se elige aleatoriamente, con el propósito de que no haya dos fuentes de sincronización dentro de la misma sesión RTP que tengan el mismo identificador de SSRC. En A.8.2 se presenta un ejemplo de algoritmo para generar un identificador aleatorio. Aunque la probabilidad de que múltiples fuentes elijan el mismo identificador es baja, todas las implementaciones RTP deben estar preparadas para detectar y resolver colisiones. En A.8, Asignación y utilización de identificadores SSRC, se describe la probabilidad de colisión junto con un mecanismo para resolver colisiones y detectar bucles de envío de nivel RTP basándose en la del identificador de SSRC. Si una fuente cambia su dirección de transporte de origen, debe también elegir un nuevo identificador de SSRC para evitar que se interprete como una fuente bucleada.

lista de CSRC: 0 a 15 elementos, de 32 bits cada uno. La lista de CSRC identifica las fuentes contribuyentes para la cabida útil contenida en este paquete. El número de identificadores es indicado por el campo CC. Si hay más de 15 fuentes contribuyentes, sólo 15 pueden ser identificadas. Los identificadores de CSRC son insertados por mezcladores, utilizando los identificadores de SSRC de las fuentes contribuyentes. Por ejemplo, para los paquetes de audio se enumeran los identificadores de SSRC de todas las fuentes que se mezclaron juntas para crear un paquete, permitiendo una indicación correcta del hablante en el receptor.

A.5.2 Multiplexación de sesiones RTP

Para un procesamiento eficaz del protocolo, debe reducirse al mínimo el número de puntos de multiplexación, como se describe en el principio de diseño de procesamiento de capa integrado [A-1]. En el RTP, la multiplexación es proporcionada por la red de transporte de destino (dirección de red y número de puerto) que define una sesión RTP. Por ejemplo, en una teleconferencia compuesta de medios de audio y de vídeo codificados por separado, cada medio debe transportarse en una sesión RTP separada con su propia dirección de transporte de destino. No se pretende que el audio y el vídeo sean transportados en una única sesión RTP y demultiplexados en base al tipo de cabida útil o los campos de SSRC. El entrelazado de paquetes con diferentes tipos de cabida útil, pero utilizando la misma SSRC, presentaría varios problemas:

1) Si se conmuta un tipo de cabida útil durante una sesión, no habría medios generales para identificar a cuál de los valores antiguos sustituyó el nuevo.

2) Un SSRC se define para identificar un solo espacio de temporización y de número de secuencia. El entrelazado de múltiples tipos de cabida útil exigiría diferentes espacios de temporización si las velocidades de reloj de los medios difieren, y requeriría suficientes espacios de número de secuencia para decir qué tipo de cabida útil sufrió pérdida de paquetes.

3) Los informes de emisor y de receptor RTCP (véase A.6.3, Informes de emisor y de receptor) sólo pueden describir un espacio de temporización y de número secuencial por SSRC y no transportan un campo de tipo de cabida útil.

4) Un mezclador RTP no podría combinar trenes entrelazados de medios incompatibles en un solo tren.

5) El transporte de múltiples medios en una sesión RTP excluye: el uso de diferentes trayectos de red o asignaciones de recursos de red si así conviene; la recepción de un subconjunto de los medios si así se desea, por ejemplo, sólo audio si con el vídeo se superase la anchura de banda disponible; e implementaciones de receptor que utilizan procesos separados para los diferentes medios, mientras que la utilización de sesiones RTP separadas permite implementaciones de un solo proceso o de múltiples procesos.

Utilizar una SSRC diferente para cada medio, pero enviarlos en la misma sesión RTP, evitaría los tres primeros problemas, pero no los dos últimos.

A.5.3 Modificaciones específicas del perfil en el encabezamiento RTP

El encabezamiento de paquetes de datos RTP existente se cree que está completo para el conjunto de funciones requeridas en común a través de todas las clases de aplicación que el RTP podría soportar. Sin embargo, de acuerdo con el principio de diseño del ALF, el encabezamiento puede ajustarse mediante modificaciones o adiciones definidas en una especificación de perfil, pero permitiendo que funcionen las herramientas de supervisión y registro independientes del perfil:

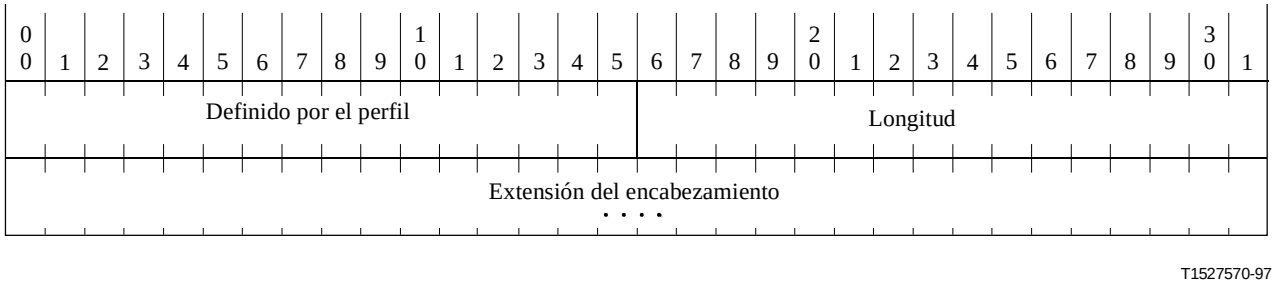
- Los campos de bits marcadores y de tipo de cabida útil transportan información específica del perfil, pero están asignados en el encabezamiento fijo, ya que muchas aplicaciones se cree que los necesitarán y podrían en otro caso añadir otra palabra de 32 bits simplemente para contenerlos. El octeto que contiene estos bits puede ser redefinido por un perfil para ajustarse a los diferentes requisitos, por ejemplo, con más o menos bits marcadores. Si hay algunos bits marcadores, uno debería situarse en el bit más significativo del octeto ya que monitores independientes del perfil podrían ser capaces de observar una correlación entre los patrones de pérdida de paquetes y el bit marcador.
- La información adicional que se requiere para un determinado formato de cabida útil, tal como una codificación de vídeo, debe transportarse en la sección de cabida útil del paquete. Ésta podría estar en un encabezamiento que esté siempre presente al comienzo de la sección de cabida útil o podría ser indicado por un valor reservado en el patrón de datos.
- Si una determinada clase de aplicaciones necesita funcionalidad adicional independiente del formato de cabida útil, el perfil bajo el cual operan estas aplicaciones debe definir campos fijos adicionales que sigan inmediatamente después del campo de SSRC del encabezamiento fijo existente. Estas aplicaciones podrán acceder rápida y directamente a los campos adicionales, mientras que los monitores o registradores independientes del perfil pueden todavía procesar los paquetes RTP interpretando sólo los doce primeros octetos.

Si resulta que se necesita funcionalidad adicional en común a lo largo de todos los perfiles, debe entonces definirse una nueva versión de RTP para hacer un cambio permanente al encabezamiento fijo.

A.5.3.1 Extensión del encabezamiento RTP

Se provee un mecanismo de extensión para permitir que las diferentes implementaciones experimenten con nuevas funciones independientes del formato de cabida útil que requieren que se transporte información adicional en el encabezamiento de paquete de datos RTP. Este mecanismo está diseñado de manera que la extensión del encabezamiento pueda ser ignorada por otras implementaciones interoperantes que no hayan sido extendidas.

Obsérvese que esta extensión del encabezamiento sólo se destina a un uso limitado. La mayor parte de los usos potenciales de este mecanismo sería mejor hacerlos de otro modo, utilizando los mecanismos descritos en la subcláusula anterior. Por ejemplo, una extensión específica del perfil del encabezamiento fijo es menos cara de procesar, ya que no es condicional ni está en una ubicación variable. La información adicional necesaria para un determinado formato de cabida útil no debe utilizar esta extensión del encabezamiento, sino que debe transportarse en la sección de cabida útil del paquete.



Si el bit X del encabezamiento RTP es uno, se agrega al encabezamiento RTP una extensión del encabezamiento variable, siguiendo a la lista de CSRC si está presente. La extensión del encabezamiento contiene un campo de longitud de 16 bits que cuenta el número de palabras de 32 bits de la extensión, excluido el encabezamiento de extensión de 4 octetos (por tanto, cero es una longitud válida). Sólo puede agregarse una extensión al encabezamiento de datos RTP. Para permitir que múltiples implementaciones interoperantes experimenten cada una independientemente con diferentes extensiones de encabezamiento, o para permitir que una determinada implementación experimente con más de un tipo de extensión del encabezamiento, los 16 primeros bits de la extensión del encabezamiento, se dejan abiertos para distinguir identificadores o parámetros. El formato de estos 16 bits ha de ser definido por la especificación de perfil bajo la cual están operando las implementaciones. Esta especificación RTP no define extensiones del encabezamiento.

A.6 Protocolo de control RTP (RTCP)

El protocolo de control RTP (RTCP, *RTP control protocol*) se basa en la transmisión periódica de los paquetes de control a todos los participantes en la sesión, utilizando el mismo mecanismo de distribución que los paquetes de datos. El protocolo subyacente debe proporcionar multiplexación de los paquetes de datos y de control, por ejemplo, utilizando números de puertos separados con UDP. El RTCP efectúa cuatro funciones:

1) La función primordial es proporcionar realimentación sobre la calidad de la distribución de datos. Ésta es una parte integrante del papel del RTP como protocolo de transporte, y está relacionada con las funciones de control de flujo y de congestión de otros protocolos de transporte. La realimentación puede ser directamente de utilidad para el control de las codificaciones adaptativas [A-6] y [A-7], pero experimentos con la multidifusión de IP han revelado que es también crítico obtener realimentación procedente de los receptores para diagnosticar averías en la distribución. El envío de informes de realimentación de recepción a todos los participantes permite a quien esté observando problemas evaluar si estos problemas son locales o globales. Con un mecanismo de distribución como es la multidifusión IP, es también posible que una entidad tal como un proveedor de servicios de

red que no intervenga por otra parte en la sesión reciba la información de realimentación actúe como un monitor de tercera parte para diagnosticar problemas de red. La función de realimentación es efectuada por los informes de emisor y de receptor RTCP, descritos más adelante en A.6.3, Informes de emisor y de receptor.

2) RTCP transporta un identificador de nivel de transporte persistente para una fuente RTP denominada el nombre canónico o CNAME (véase A.6.4.1, CNAME: Elemento SDES identificador de punto extremo canónico). Como el identificador de SSRC puede cambiar si se descubre una contradicción o si se reinicia un programa, los receptores requieren que el CNAME siga la pista de cada participante. Los receptores también requieren que el CNAME asocie múltiples trenes de datos de un participante dado en un conjunto de sesiones RTP relacionadas, por ejemplo para sincronizar audio y vídeo.

3) Las dos primeras funciones requieren que todos los participantes envíen paquetes RTCP, por lo que la velocidad debe ser controlada a fin de que el RTP se escale hasta un gran número de participantes. Haciendo que cada participante envíe sus paquetes de control a todos los demás, cada uno puede observar independientemente el número de participantes. Este número se utiliza para calcular la velocidad a la que se envían los paquetes, como se indica en A.6.2, Intervalo de transmisión RTCP.

4) Una cuarta función opcional es transportar información de control de sesión mínima, por ejemplo, identificación de participantes a visualizar en la interfaz de usuario. Lo más probable es que esto sea útil en sesiones "con control menos riguroso", en las que los participantes entran y salen sin control de la participación ni negociación de parámetros. RTCP sirve como un canal conveniente para llegar a todos los participantes, pero no se espera que soporte necesariamente todos los requisitos de comunicación de control de una aplicación. Puede necesitarse un protocolo de control de sesión de nivel superior, lo que cae fuera del alcance de esta Recomendación.

Las funciones 1 a 3 son obligatorias cuando se utiliza RTP en el entorno multidifusión IP, y se recomiendan para todos los entornos. Se aconseja que los diseñadores de aplicaciones RTP eviten mecanismos que sólo puedan funcionar en modo unidifusión y que no los escalen a números mayores.

A.6.1 Formato de paquetes RTCP

Esta especificación define varios tipos de paquetes RTCP para transportar una variedad de información de control:

SR: Informe de emisor, con los datos estadísticos de transmisión y recepción de los participantes que son emisores activos.

RR: Informe de receptor, con los datos estadísticos de recepción de los participantes que no son emisores activos.

SDES: Elementos de descripción de origen, incluido CNAME.

BYE: Indica fin de la participación.

APP: Funciones específicas de la aplicación.

Cada paquete RTCP comienza por una parte fija similar a la de los paquetes de datos de RTP, seguida por elementos estructurados que pueden ser de longitud variable según el tipo de paquete, pero que siempre terminan en una frontera de 32 bits. El requisito de alineación y un campo de longitud en la parte fija se incluyen para hacer los paquetes RTCP "apilables". Múltiples paquetes RTCP pueden concatenarse sin separadores intermedios para formar un paquete RTCP compuesto que es enviado en un solo paquete del protocolo de capa inferior, por ejemplo, UDP. No hay cuenta explícita de paquetes RTCP individuales en el paquete compuesto, ya que los protocolos de capa inferior se cree que proporcionarán la longitud total para determinar el fin del paquete compuesto.

Cada paquete RTCP individual del paquete compuesto puede procesarse independientemente sin requisitos sobre el orden de combinación de los paquetes. Sin embargo, a fin de efectuar las funciones del protocolo, se imponen las siguientes constricciones:

- Deben enviarse datos estadísticos de recepción (en SR o RR) tan a menudo como lo permitan las constricciones de anchura de banda para maximizar la resolución de los datos estadísticos, por lo que cada paquete RTCP compuesto periódicamente transmitido debe incluir un paquete de informe.
- Los receptores nuevos necesitan recibir el CNAME de una fuente lo antes posible para identificar la fuente y comenzar a asociar medios para fines tales como sincronización de los labios, por lo que cada paquete RTCP compuesto debe incluir el SDES CNAME.
- El número de tipos de paquetes que puede aparecer primero en el paquete compuesto debe limitarse a aumentar el número de bits constantes en la primera palabra y la probabilidad de validar con éxito paquetes RTCP frente a paquetes de datos RTP mal direccionados u otros paquetes no relacionados.

Así, todos los paquetes RTCP deben enviarse en un paquete compuesto de al menos dos paquetes individuales, con el siguiente formato recomendado:

Prefijo de criptación: Exclusivamente si el paquete compuesto ha de ser criptado, es prefijado por una cantidad de 32 bits aleatorios para cada paquete compuesto transmitido.

SR o RR: El primer paquete RTCP del paquete compuesto debe siempre ser un paquete de informe para facilitar la validación del encabezamiento descrita en A.2. Esto es cierto aun si no se han enviado ni recibido datos, en cuyo caso se envía un RR vacío, y aun si el único otro paquete RTCP del paquete compuesto es un BYE.

RR adicionales: Si el número de fuentes sobre las que se comunican datos estadísticos de recepción es superior a 31, el número que encajará en un paquete SR o RR, y luego paquetes RR adicionales deben seguir al paquete de informe inicial.

SDES: Debe incluirse un paquete SDES que contenga un elemento CNAME en cada paquete RTCP compuesto. Pueden opcionalmente incluirse otros elementos de descripción de fuente si lo requiere una determinada aplicación, a reserva de constricciones de anchura de banda (véase A.6.2.2, Asignación de anchura de banda de descripción de fuente).

BYE o APP: Otros tipos de paquetes RTCP, incluidos los que están aún por definir, pueden seguir en cualquier orden, salvo si BYE debe ser el último paquete enviado con una determinada SSRC/CSRC. Los tipos de paquetes pueden aparecer más de una vez.

Es aconsejable que los traductores y mezcladores combinen paquetes RTCP individuales a partir de las múltiples fuentes que están remitiendo en un paquete compuesto siempre que sea realizable, a fin de amortizar la tara de paquete (véase A.7, Traductores y mezcladores RTP). En la figura A.1 se muestra un ejemplo de paquete compuesto que podría ser producido por un mezclador. Si la longitud total de un paquete compuesto supera la unidad de transmisión máxima (MTU, *maximun transmission unit*) del trayecto de red, puede segmentarse en múltiples paquetes compuestos más cortos a transmitir en múltiples paquetes separados del protocolo subyacente. Adviértase que cada uno de los paquetes compuestos debe comenzar por un paquete SR o RR.

Una implementación puede ignorar paquetes RTCP entrantes con tipos que le sean desconocidos. Pueden registrarse tipos de paquetes RTCP adicionales en la autoridad de números asignados de Internet (IANA, *Internet assigned numbers authority*).

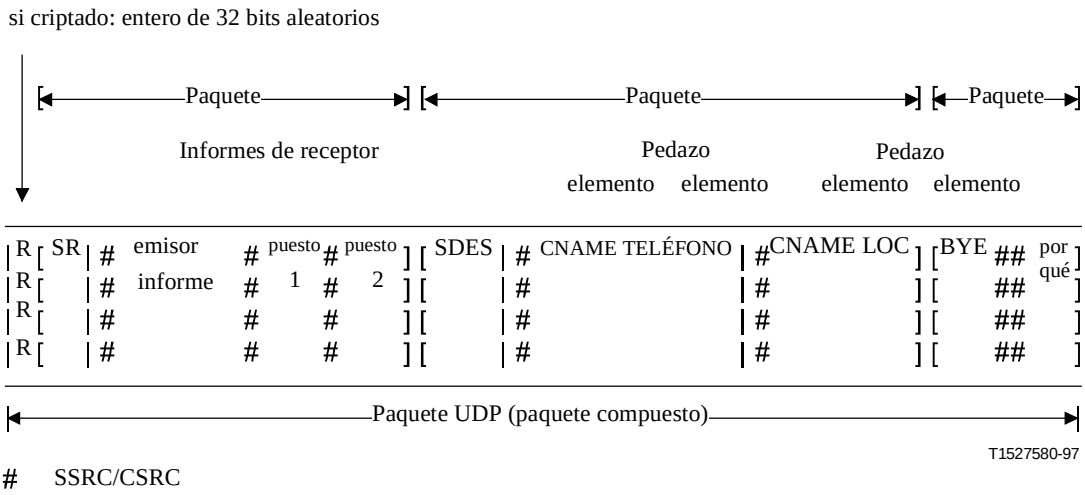


Figura A.1/H.225.0 – Ejemplo de un paquete compuesto RTCP

A.6.2 Intervalo de transmisión RTCP

RTP está diseñado para permitir que una aplicación se escale automáticamente en tamaños de sesión que varían de pocos a miles de participantes. Por ejemplo, en una audioconferencia el tráfico de datos es de por sí autolimitador, ya que sólo una o dos personas hablarán a un tiempo, por lo que en la distribución multidifusión la velocidad binaria en un determinado enlace sigue siendo relativamente constante independientemente del número de participantes. Sin embargo, el control de tráfico no es autolimitador. Si los informes de recepción de cada participante se enviaran a velocidad constante, el tráfico de control crecería linealmente con el número de participantes. Por tanto, la velocidad debe escalarse en sentido descendente.

Para cada sesión se supone que los datos de tráfico están sujetos a un límite agregado denominado la "anchura de banda de sesión" que ha de dividirse entre los participantes. Esta anchura de banda podría estar reservada y el límite ser introducido por la red, o podría simplemente ser una parte razonable. La anchura de banda de sesión puede elegirse sobre la base de algún costo o del conocimiento *a priori* de la anchura de banda de red disponible para la sesión. Es algo independiente de la codificación de los medios, pero la elección de la codificación puede ser limitada por la anchura de banda de sesión. El parámetro anchura de banda de sesión se espera que sea suministrado por una aplicación de gestión de sesión cuando invoca una aplicación de medios, pero las aplicaciones de medios pueden también fijar un valor por defecto basado en la anchura de banda de datos de un solo usuario para la codificación seleccionada para la sesión. La aplicación puede también introducir límites de anchura de banda basados en reglas de alcance multidifusión u otros criterios.

Entre los cálculos de anchura de banda para tráfico de control y de datos se hallan los protocolos de transporte y de red de capa inferior (por ejemplo, UDP e IP), ya que eso es lo que el sistema de reserva de recursos necesitaría conocer. La aplicación puede también esperarse que conozca cuál de estos protocolos están en uso. Los encabezamientos de nivel de enlace no se incluyen en el cálculo, ya que el paquete será encapsulado con diferentes encabezamientos de nivel enlace a medida que viaja.

El tráfico de control debe limitarse a una pequeña y conocida fracción de la anchura de banda de sesión: pequeña para que no se degrade la función primaria del protocolo de transporte de transportar datos; conocida para que el tráfico de control pueda incluirse en la especificación de anchura de banda dada a un protocolo de reserva de recursos, y para que cada participante pueda calcular independientemente su parte. Se sugiere que la fracción de la anchura de banda de sesión asignada a RTCP pueda fijarse al 5%. Aunque el valor de ésta y otras constantes en el cálculo del intervalo no

es crítica, todos los participantes en la sesión deben utilizar los mismos valores, por lo que se calculará el mismo intervalo. Por tanto, estas constantes deben fijarse para un perfil determinado.

El algoritmo descrito en A.7 se diseñó para que alcanzase las metas antes expuestas. Calcula el intervalo entre paquetes RTCP compuestos de emisión para dividir la anchura de banda de tráfico de control permitida entre los participantes, lo cual permite a una aplicación proporcionar respuesta rápida para pequeñas sesiones en las que, por ejemplo, es importante la identificación de todos los participantes, pero también adaptarse automáticamente a sesiones grandes. El algoritmo incorpora las siguientes características:

- Se asigna colectivamente a los emisores al menos 1/4 de la anchura de banda de tráfico de control, a fin de que en las sesiones con un gran número de receptores pero un pequeño número de emisores, los participantes de incorporación reciente reciban más rápidamente el CNAME para los puestos de emisión.
- Es necesario que el intervalo calculado entre paquetes RTCP sea mayor que 5 segundos como mínimo para evitar que haya ráfagas de paquetes RTCP que superen la anchura de banda permitida cuando el número de participantes es pequeño y el tráfico no se ha alisado de acuerdo con la ley de los grandes números.
- El intervalo entre paquetes RTCP se varía aleatoriamente en la gama de [0,5, 1,5] veces el intervalo calculado para evitar la sincronización no deliberada de todos los participantes [A-8]. El primer paquete RTCP enviado después de incorporarse a una sesión es también demorado por una variación aleatoria de la mitad del intervalo RTCP mínimo en caso de que la aplicación se inicie en múltiples puestos simultáneamente, por ejemplo, como si fuera iniciada por un anuncio de sesión.
- Se calcula una estimación dinámica del tamaño de paquete RTCP compuesto medio, incluidos todos los recibidos y enviados, para adaptarse automáticamente a cambios en la cantidad de información de control transportada.

Este algoritmo puede utilizarse en sesiones en las que todos los participantes son autorizados a emitir. En ese caso, el parámetro anchura de banda de sesión es el producto de la anchura de banda de cada emisor individual por el número de participantes, y la anchura de banda RTCP es el 5% de esa cantidad.

A.6.2.1 Mantenimiento del número de miembros de sesión

El cálculo del intervalo de paquetes RTCP depende de la estimación del número de puestos que participan en la sesión. Los nuevos puestos se añaden a la cuenta cuando son oídos, y se crea una nueva entrada en un cuadro indexado por el identificador de SSRC o CSRC (véase A.8.2, Resolución de colisiones y detección de bucles) para seguirles la pista. Las nuevas entradas no pueden ser consideradas válidas hasta que se han recibido múltiples paquetes que transportan la nueva SSRC (véase A.6.1). Las entradas pueden suprimirse del cuadro cuando se recibe un paquete RTCP BYE con el correspondiente identificador de SSRC.

Un participante puede marcar otro puesto como inactivo o suprimirlo si no es aún válido, si no se ha recibido ningún paquete RTP o RTCP durante un pequeño número de intervalos de informe RTCP (se sugiere que sean 5). Esto permite una cierta solidez contra la pérdida de paquetes. Todos los puestos deben calcular aproximadamente el mismo valor para el intervalo de informe RTCP a fin de que esta temporización funcione adecuadamente.

Una vez que se ha validado un puesto, si éste posteriormente se marca inactivo debe no obstante conservarse el estado de ese puesto, el cual debe seguir contándose en el número total de puestos que comparten anchura de banda RTCP durante un periodo suficientemente largo para comprender particiones de red típicas. Se evita así un tráfico excesivo, cuando la partición se subsana, debido a un intervalo de informe RTCP que es demasiado pequeño. Se sugiere una temporización de 30 minutos. Adviértase que este tiempo sigue siendo todavía 5 veces mayor que el mayor valor al que se espera que el intervalo de informe RTCP se escale convenientemente, de unos 2 a 5 minutos.

A.6.2.2 Asignación de anchura de banda de descripción de fuente

Esta especificación define varios elementos de descripción de fuente (SDES, *source description*) además del elemento CNAME obligatorio, tales como NAME (nombre personal) y EMAIL (dirección de correo electrónico). También proporciona un medio de definir nuevos tipos de paquetes RTCP específicos de la aplicación. Las aplicaciones deben actuar con precaución al atribuir anchura de banda de control a esta información adicional, ya que reducirá la velocidad a la que se envían informes de recepción y CNAME, con lo que se degrada la prestación del protocolo. Se recomienda que se utilice no más del 20% de la anchura de banda RTCP asignada a un solo participante para transportar la información adicional. Además, no se pretende que todos los elementos SDES deban incluirse en cada aplicación. Los que se incluyan deben asignarse como fracción de la anchura de banda de acuerdo con su utilidad. Más que estimar estas fracciones dinámicamente, se recomienda que los porcentajes se traduzcan estáticamente a cuentas de intervalo de informe sobre la base de la longitud típica de un elemento.

Por ejemplo, una aplicación puede diseñarse para que envíe sólo CNAME, NAME y EMAIL, y no otros elementos. A NAME podría dársele mucha mayor prioridad que a EMAIL, debido a que el NAME se visualizaría continuamente en la interfaz de usuario de aplicación, mientras que EMAIL se visualizaría sólo cuando se solicitase. En cada intervalo RTCP, se enviaría un paquete RR y un paquete SDES con el elemento CNAME. En una pequeña sesión en cuya operación se aplique el intervalo mínimo, eso sería cada 5 segundos en promedio. Cada tres intervalos (15 segundos), se incluiría un elemento extra en el paquete SDES. Siete de las ocho veces éste sería el elemento NAME, y cada octava vez (2 minutos) sería el elemento EMAIL.

Cuando funcionan múltiples aplicaciones concertadamente utilizando vinculación de aplicaciones recíprocas mediante un CNAME común para cada participante, por ejemplo, en una conferencia multimedios compuesta por una sesión RTP para cada medio, la información SDES adicional podría enviarse únicamente en una sesión RTP. Las otras sesiones transportarían sólo el elemento CNAME.

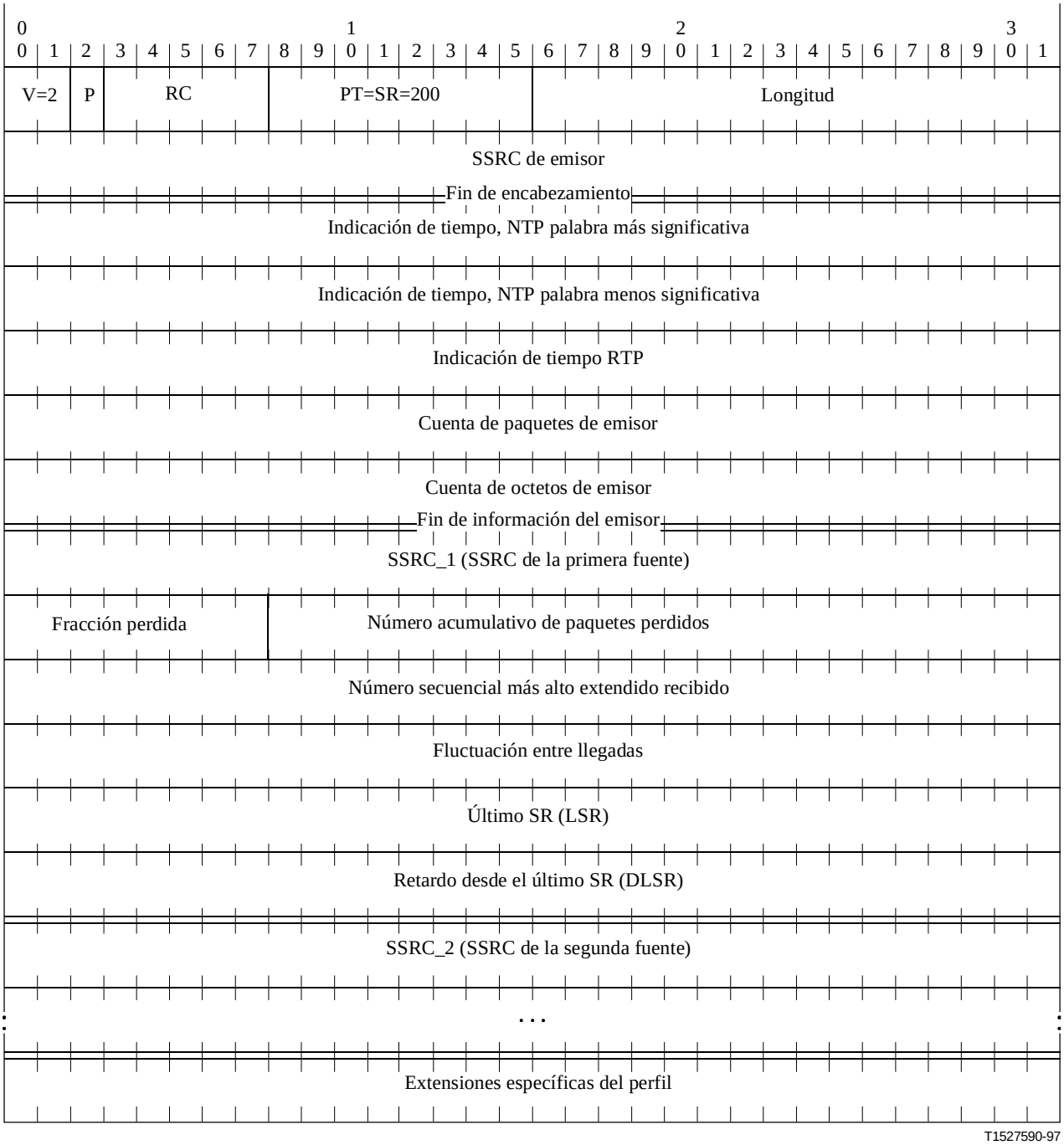
A.6.3 Informes de emisor y de receptor

Los receptores RTP proporcionan realimentación de calidad recepción utilizando paquetes de informe RTCP que pueden adoptar una de dos formas dependiendo de si el receptor es también un emisor. La única diferencia entre las formas de informe de emisor (SR, *sender report*) y de informe de receptor (RR, *receiver report*), además del código de tipo de paquete, es que el informe de emisor incluye una sección de información del emisor de 20 bytes, para su utilización por emisores activos. El SR es emitido si un puesto ha enviado cualesquiera paquetes de datos durante el intervalo desde que emitió el último informe o el anterior; en otro caso, se emite el RR.

Las formas SR y RR incluyen ambos cero o más bloques de informe de recepción, uno para cada una de las fuentes de sincronización de las cuales este receptor ha recibido paquetes de datos RTP desde el último informe. Los informes no se emiten para fuentes contribuyentes enumeradas en la lista de CSRC. Cada bloque de informe de recepción proporciona estadísticas sobre los datos recibidos de cada fuente considerada indicados en ese bloque. Dado que en un paquete SR o RR cabrán 31 bloques de informe de recepción, los paquetes RR adicionales pueden apilarse después del paquete SR o RR inicial si es necesario para que contengan los informes de recepción de todas las fuentes oídas durante el intervalo desde el último informe.

En las subcláusulas que siguen se definen los formatos de los dos informes, cómo pueden extenderse de una manera específica del perfil si una aplicación requiere información de realimentación adicional, y cómo pueden utilizarse los informes. En A.7, Traductores y mezcladores RTP, se dan detalles de información de recepción suministrada por traductores y mezcladores.

A.6.3.1 SR: Paquete RTCP de informe de emisor



El paquete de informe de emisor consta de tres secciones, posiblemente seguidas por una cuarta sección de extensión específica del perfil, si se define. La primera sección, el encabezamiento, tiene 8 octetos de largo. Los campos tienen el siguiente significado:

versión (V): 2 bits. Identifica la versión de RTP, que es la misma en los paquetes RTCP que en los paquetes de datos RTP. La versión definida por esta especificación es dos (2).

relleno (P, *padding*): 1 bit. Si el bit de relleno se fija, este paquete RTCP contiene algunos octetos del relleno adicionales al final que no forman parte de la información de control. El último octeto del relleno es una cuenta de cuántos octetos de relleno deben ser ignorados. El relleno pueden necesitarlo algunos algoritmos de criptación con tamaños de bloque fijos. En un paquete RTCP compuesto, el relleno debe sólo necesitarse en el último paquete individual, debido a que el paquete compuesto está criptado en su conjunto.

cuenta de informe de recepción (RC, *report count*): 5 bits. El número de bloques de informe de recepción contenido en este paquete. El valor cero es válido.

tipo de paquete (PT, *packet type*): 8 bits. Contiene la constante 200 para identificar éste como un paquete RTCP SR.

longitud: 16 bits. La longitud de este paquete RTCP en palabras de 32 bits menos uno, incluido el encabezamiento y cualquier relleno. (El desplazamiento de uno hace cero una longitud válida y evita un posible bucle infinito al explorar un paquete RTCP compuesto, mientras que contar palabras de 32 bits evita una comprobación de validez para un múltiplo de 4.)

SSRC: 32 bits. El identificador de fuente de sincronización del originador de este paquete SR.

La segunda sección, la información de emisor, tiene 20 octetos de largo y está presente en cada paquete de informe de emisor. Hace un sumario de las transmisiones de datos desde este emisor. Los campos tienen el siguiente significado:

indicación de tiempo NTP: 64 bits. Indica el tiempo de reloj cuando se envió este informe de manera que pueda utilizarse en combinación con indicaciones de tiempo devueltas en informes de recepción procedentes de otros receptores para medir la propagación de ida y vuelta a esos receptores. Los receptores deben esperar que la exactitud de medición de la indicación de tiempo pueda limitarse a bastante menos que la resolución de la indicación de tiempo NTP. La incertidumbre de medición de la indicación de tiempo no se indica, ya que no puede conocerse. Un emisor que puede estar al corriente del tiempo transcurrido, pero que no tiene ninguna noción del tiempo de reloj, puede en su lugar utilizar el tiempo transcurrido desde la incorporación a la sesión. Éste se supone que es inferior a 68 años, por lo que el bit superior será cero. Es admisible utilizar el reloj de muestreo para estimar el tiempo de reloj transcurrido. Un emisor que no tiene ninguna noción del tiempo de reloj o del tiempo transcurrido puede fijar la indicación de tiempo NTP a cero.

indicación de tiempo RTP: 32 bits. Corresponde al mismo tiempo que la indicación de tiempo NTP (véase más arriba), pero en las mismas unidades y con el mismo desplazamiento aleatorio que las indicaciones de tiempo RTP en los paquetes de datos. Esta correspondencia puede utilizarse para la sincronización intramedios e intermedios de fuentes cuyas indicaciones de tiempo NTP están sincronizadas, y puede ser utilizada por receptores independientes de los medios para estimar la frecuencia de reloj RTP nominal. Adviértase que en la mayoría de los casos esta indicación de tiempo no será igual a la indicación de tiempo RTP en cualquier paquete de datos adyacente. Se calcula más bien a partir de la indicación de tiempo NTP correspondiente utilizando la relación entre el contador de indicaciones de tiempo RTP y el tiempo real tal como es mantenida comprobando periódicamente el tiempo de reloj en un instante de muestreo.

cuenta de paquetes del emisor: 32 bits. El número total de paquetes de datos RTP transmitidos por el emisor desde el comienzo de la transmisión hasta el momento en que se generó este paquete SR. La cuenta se reinicia si el emisor cambia su identificador de SSRC.

cuenta de octetos del emisor: 32 bits. El número total de octetos de cabida útil (es decir, sin incluir encabezamiento ni relleno) transmitidos en paquetes de datos RTP por el emisor desde el comienzo de la transmisión hasta el momento en que se generó este paquete SR. La cuenta se reinicia si el emisor cambia su identificador de SSRC. Este campo puede utilizarse para estimar la velocidad de datos de cabida útil media.

La tercera sección contiene cero o más bloques de informe de recepción, según el número de otras fuentes oídas por este emisor desde el último informe. Cada bloque de informe de recepción transporta datos estadísticos sobre la recepción de paquetes RTP procedentes de una sola fuente de sincronización. Los receptores no transportan otros datos estadísticos cuando una fuente cambia su identificador SSRC debido a una colisión. Estos datos estadísticos son:

SSRC_n (identificador de fuente): 32 bits. El identificador de SSRC de la fuente a la que pertenece la información de este bloque de informe de recepción.

fracción perdida: 8 bits. La fracción de paquetes de datos RTP procedentes del SSRC_n de fuente perdido desde que se envió el paquete SR o RR anterior, expresada como número de punto fijo con el punto binario en el borde izquierdo del campo. (Esto es equivalente a tomar la parte entera después de multiplicar la fracción de pérdida por 256 puntos.) Esta fracción se define como el número de paquetes perdidos dividido por el número de paquetes esperados, que se define en el párrafo siguiente. En A.6.3 se muestra una implementación. Si la pérdida es negativa debido a duplicados, la fracción perdida se pone a cero. Adviértase que un receptor no puede decir si se perdieron paquetes después del último recibido, y que no habrá ningún bloque de informe de recepción emitido para una fuente si se han perdido todos los paquetes procedentes de esa fuente durante el último intervalo de información.

número acumulativo de paquetes perdidos: 24 bits. El número total de paquetes de datos RTP procedentes del SSRC_n de fuente que se han perdido desde el comienzo de la recepción. Este número se define como el número de paquetes esperado menos el número de paquetes realmente recibidos, donde el número de paquetes recibidos incluye posibles paquetes tardíos o duplicados. Así, los paquetes que llegan tarde no se cuentan como perdidos, y la pérdida puede ser negativa si hay duplicados. El número de paquetes esperado se define como el último número secuencial con extendido recibido. Puede calcularse como se indica en A.6.3.

número secuencial más alto extendido recibido: 32 bits. Los 16 bits bajos contienen el número de secuencia más alto recibido en un paquete de datos RTP procedente del SSRC_n de fuente, y los 16 bits más significativos extienden ese número secuencial con la cuenta correspondiente de ciclos de números secuenciales, que puede mantenerse según el algoritmo de A.13. Adviértase que diferentes receptores dentro de la misma sesión generarán extensiones diferentes al número secuencial y sus tiempos de comienzo difieren significativamente.

fluctuación entre llegadas: 32 bits. Una estimación de la varianza estadística del tiempo entre llegadas de paquetes de datos RTP, medido en unidades de indicación de tiempo 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 paquetes en el receptor en comparación con la del emisor para un par de paquetes. Como se muestra en la ecuación que sigue, es equivalente a la diferencia en el "tiempo de tránsito relativo" para los dos paquetes; el tiempo de tránsito relativo es la diferencia entre una indicación de tiempo RTP de paquete y el reloj de receptor en el momento de la llegada, medida en las mismas unidades.

Si es la indicación de tiempo RTP del paquete i, y Ri es el tiempo de llegada en unidades de indicación de tiempo RTP para el paquete i, entonces para dos paquetes i y j, D puede expresarse como:

$$D(i+j) = (Rj - Ri) - (Sj - Si) = (Rj - Sj) - (Rj - Si)$$

La fluctuación entre llegadas se calcula continuamente a medida que cada paquete de datos i es recibido del SSRC_n de fuente, utilizando esta diferencia D para ese paquete y el paquete anterior i – 1 en orden de llegada (no necesariamente en secuencia), según la fórmula:

$$J = J + \frac{\left|D(i-1,i)\right| - J}{16}$$

Siempre que se emite un informe de recepción, el valor corriente de J es muestreado.

El cálculo de la fluctuación se recomienda aquí que permita a los monitores independientes del perfil hacer interpretaciones válidas de informes procedentes de diferentes implementaciones. Este algoritmo es el estimador óptimo de primer orden y el parámetro de ganancia 1/16 produce una buena relación de reducción de ruido, pero manteniendo una razonable velocidad de convergencia [A-9]. En A.8 se muestra un ejemplo de implementación.

última indicación de tiempo SR (LSR, *last SR timestamp*): 32 bits. Los 32 bits centrales de los 64 de la indicación de tiempo NTP (que se explica en A.4, Orden, alineación y formato horario de los bytes) recibidos como parte del más reciente paquete de informe de emisor RTCP (SR) procedente del SSRC_n de fuente. Si aún no se ha recibido ningún SR, el campo se pone a cero.

retardo desde el último SR (DLSR, *delay since last SR*): 32 bits. El retardo, expresado en unidades de 1/65536 segundos, comprendido entre la recepción del último paquete SR procedente de la fuente SSRC_n y el envío de este bloque de informe de recepción. Si no se ha recibido aún ningún paquete SR del SSRC_n, el campo DLSR se fija a cero.

Designemos por SSRC_r el receptor que emite este informe de receptor. El SSRC_n de origen puede computar el retardo de propagación de ida y vuelta al SSRC_r registrando el tiempo en que se recibió este bloque de informe de recepción. Calcula el tiempo total de ida y vuelta A – LSR utilizando el último campo de indicación de tiempo SR (LSR), y restando luego este campo para dejar el retardo de propagación de ida y vuelta como (A – LSR – DLSR). Esto se ilustra en la figura A.2.

Puede utilizarse como una medición aproximada de la distancia a receptores agrupados, aunque algunos enlaces tienen retardos muy asimétricos.

A.6.3.2 Paquete RTCP de informe de receptor

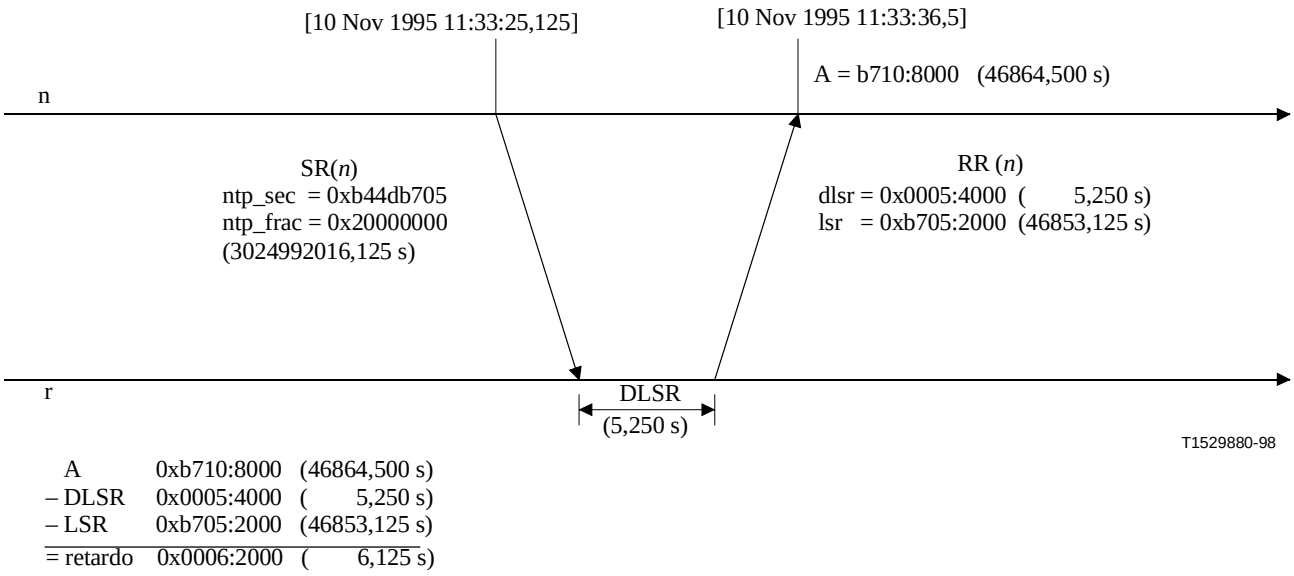
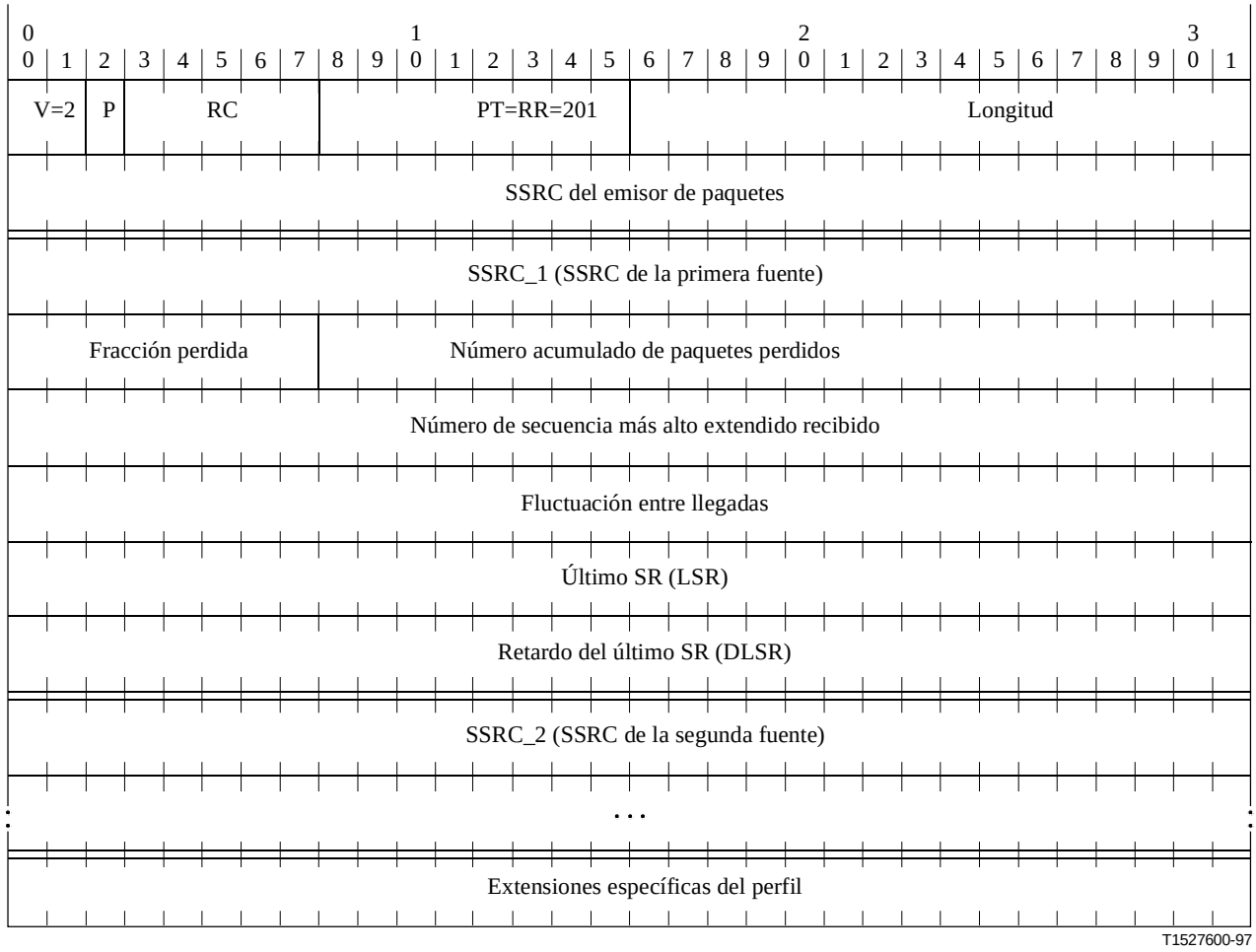


Figura A.2/H.225.0 – Ejemplo de cálculo del tiempo de ida y vuelta



El formato del paquete de informe de receptor (RR) es el mismo que el paquete SR, salvo en que el campo de tipo de paquete contiene la constante 201 y se omiten las cinco palabras de la información de emisor (son éstas las indicaciones de tiempo NTP y RTP y las cuentas de paquetes y octetos del emisor). Los campos restantes tienen el mismo significado que para el paquete SR.

Un paquete RR vacío (RC = 0) se pone a la cabeza de un paquete RTCP compuesto cuando no existe transmisión ni recepción de datos de la que informar.

A.6.3.3 Extensión de los informes de emisor y de receptor

Un perfil debe definir extensiones específicas del perfil o de la aplicación del informe de emisor y de receptor, si hay información adicional que deba comunicarse regularmente acerca del emisor o los receptores. Este método debe ampliarse preferentemente a definir otro tipo de paquete RTCP porque requiere menos tara:

- menor número de octetos en el paquete (ningún encabezamiento RTCP o campo de SSRC);
- análisis sintáctico más sencillo y rápido debido a que las aplicaciones que funcionan bajo ese perfil estarían programadas para esperar siempre los campos de extensión en la ubicación directamente accesible después de los informes de recepción.

Si se requiere información de usuario adicional, debe incluirse primero en la extensión de los informes de emisor, pero no estaría presente en los informes de receptor. Se ha de incluir información sobre receptores, estos datos se estructurarían como una formación de bloques paralela a la formación existente de bloques de informe de recepción; es decir el número de bloques se indicaría mediante el campo RC.

A.6.3.4 Análisis de los informes de emisor y de receptor

Se cree que una realimentación de calidad de recepción será de utilidad no sólo para el emisor sino también para otros receptores y monitores de terceras partes. El emisor puede modificar sus transmisores sobre la base de la realimentación; los receptores pueden determinar si los problemas son locales, regionales o globales; los gestores de redes pueden utilizar monitores independientes del perfil que reciban sólo los paquetes RTCP y no los correspondientes paquetes de datos RTP para evaluar las prestaciones de sus redes en la distribución multidifusión.

Se utilizan cuentas acumulativas en los bloques de información de emisor y de informe de receptor de manera que puedan calcularse diferencias entre cualesquiera dos informes para hacer mediciones tanto en periodos breves como largos, y para proporcionar resistencia contra la pérdida de un informe. La diferencia entre los dos últimos informes recibidos puede utilizarse para estimar la calidad reciente de la distribución. La indicación de tiempo NTP se incluye de manera que las velocidades puedan calcularse a partir de estas diferencias en el intervalo entre dos informes. Como la indicación de tiempo es independiente de la velocidad de reloj para la codificación de datos, es posible implementar monitores independientes de la codificación y del perfil.

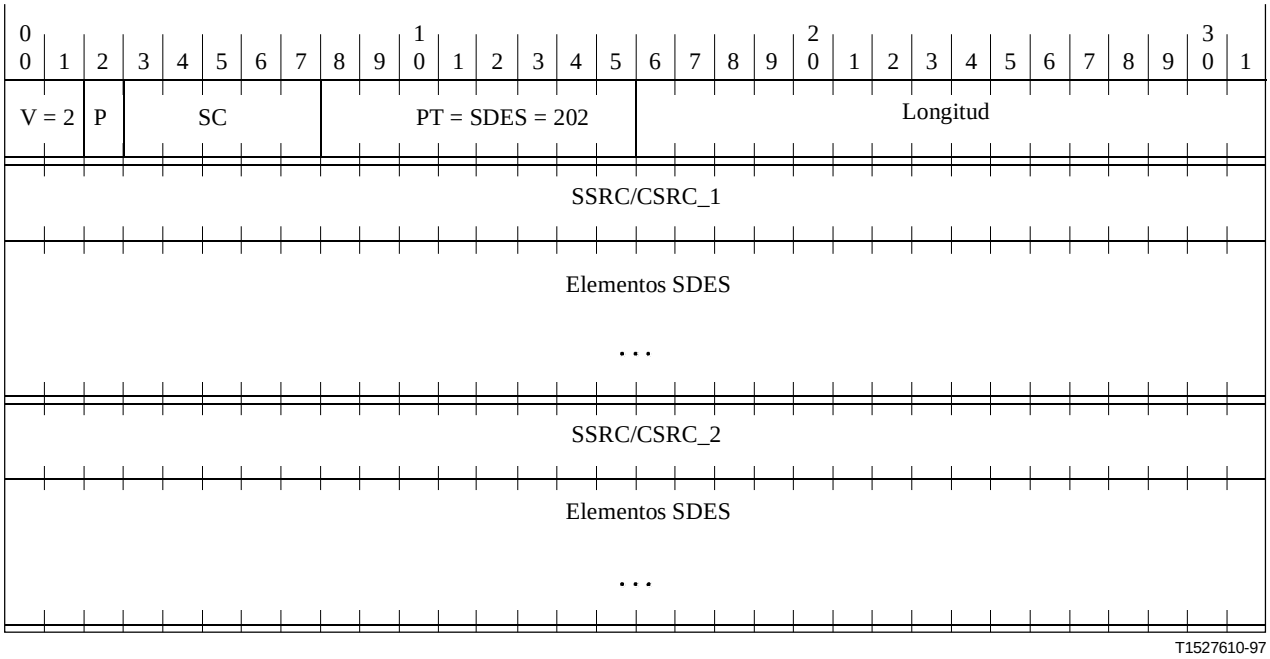
Un ejemplo de cálculo es la velocidad perdida de paquetes en el intervalo entre dos informes de recepción. La diferencia en el número acumulativo de paquetes perdidos da el número perdido durante ese intervalo. La diferencia en los últimos números de secuencia extendidos recibidos da el número de paquetes esperado durante el intervalo. La relación entre estos dos es la fracción de pérdida de paquetes en el intervalo. Esta relación debe ser igual al campo de fracción perdida si los dos informes son consecutivos, pero no en otro caso. La velocidad de pérdida por segundo puede obtenerse dividiendo la fracción de pérdida por la diferencia en indicaciones de tiempo NTP, expresada en segundos. El número de paquetes recibidos es el número de paquetes esperados menos el número de paquetes perdidos. El número de paquetes esperados puede también utilizarse para juzgar la validez estadística de cualesquiera estimaciones de pérdida. Por ejemplo, uno de cinco paquetes perdidos tiene una significación menor que 200 entre 1000.

A partir de la información del emisor, un monitor de tercera parte puede calcular la velocidad de datos media de cabida útil y la velocidad de paquetes media en un intervalo sin recibir los datos. La relación entre las dos da el tamaño medio de cabida útil. Puede suponerse que la pérdida de paquetes es independiente del tamaño de paquete, por lo que el número de paquetes recibidos por un receptor determinado multiplicado por el tamaño de cabida útil media (o el tamaño de paquete correspondiente) da el caudal aparente disponible para ese receptor.

Además de las cuentas acumulativas que permiten mediciones de pérdidas de paquetes de larga duración utilizando diferencias entre informes, el campo de fracción de pérdida proporciona una medición de corta duración a partir de un único informe. Esto resulta más importante a medida que el tamaño de una sesión se sobreescala lo bastante para que la información de estado de recepción no pueda mantenerse para todos los receptores o el intervalo entre informes resulta suficientemente grande para que sólo un informe pueda haber sido recibido de un determinado receptor.

El campo de fluctuación entre llegadas proporciona una segunda medición de corta duración de la congestión de red. La pérdida de paquetes es un índice de congestión persistente mientras que la medición de fluctuación lo es de la congestión transitoria. La medida de fluctuación puede indicar congestión antes que produzca pérdida de paquetes. Como el campo de fluctuación entre llegadas es sólo una instantánea de la fluctuación en el momento de un informe, puede ser necesario analizar cierto número de informes procedentes de un receptor en el tiempo o procedentes de múltiples receptores, por ejemplo, dentro de una única red.

A.6.4 SDES: Paquete RTCP de descripción de fuente



El paquete SDES es una estructura trinivel compuesta de un encabezamiento o cero o más pedazos, cada uno de los cuales se compone de elementos que describen la fuente identificada en ese pedazo. Los elementos se describen individualmente en las subcláusulas que siguen.

versión (V), relleno (P), longitud: Como se describe para el paquete SR (véase A.6.3.1, SR: paquete RTCP de informe de emisor).

tipo de paquetes (PT): 8 bits. Contiene la constante 202 para identificar éste como un paquete RTCP SDES.

cuenta de fuente (SC): 5 bits. Número de pedazos SSRC/CSRC contenido en el paquete SDES. Un valor cero es válido pero inútil.

Cada pedazo consta de un identificador de SSRC/CSRC seguido por una lista de cero o más elementos, que transportan información sobre la SSRC/CSRC. Cada pedazo empieza en una frontera de 32 bits. Cada elemento consta de un campo de tipo de 8 bits, una cuenta de octetos de 8 bits que describe la longitud del texto (por tanto, sin incluir este encabezador de dos octetos) y del propio texto. Adviértase que el texto no puede tener más de 255 octetos, lo que no obstante es consecuente con la necesidad de limitar el consumo de anchura de banda RTCP.

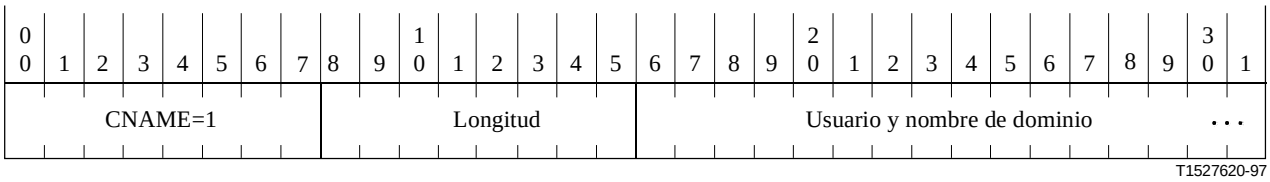
El texto se codifica según la codificación UTF-2 especificada en el anexo F de ISO/CEI 10646-1 [A-10]. Esta codificación también se conoce como UTF-8 o UTF-FSS. Se describe en "File System Safe UCS Transformation Format (FSS_UTF)", X/Open Preliminary Specification, Document Number P316 y Unicode Technical Report N.° 4. US-ASCII es un subconjunto de esta codificación y no requiere codificación adicional. La presencia de codificaciones multiocteto se indica fijando el bit más significativo de un carácter al valor uno.

Los elementos son consecutivos, es decir, los elementos no son individualmente rellenados a una frontera de 32 bits. El texto no termina en nulos, ya que algunas codificaciones multioctetos incluyen octetos nulos. La lista de elementos en cada pedazo está terminada por uno o más octetos nulos, el primero de los cuales es interpretado como un tipo de elemento cero que indica el fin de la lista, y el resto se necesita como relleno hasta la siguiente frontera de 32 bits. Un pedazo con cero elementos (cuatro octetos nulos) es válido pero inútil.

Los sistemas de extremo envían un paquete SDES que contiene su propio identificador de fuente (el mismo que el SSRC en el encabezamiento RTP fijo). Un mezclador envía un paquete SDES que contiene un pedazo para cada fuente contribuyente de la cual está recibiendo información SDES, o múltiples paquetes SDES completos en el formato citado si hay más de 31 de esas fuentes (véase A.2.3, Mezcladores y traductores).

Los elementos SDES actualmente definidos se describen en las subcláusulas siguientes. Sólo el elemento CNAME es obligatorio. Algunos elementos aquí mostrados pueden ser útiles sólo en determinados perfiles, pero los tipos de elemento se asignan todos a partir de un espacio común para favorecer el uso compartido y para simplificar las aplicaciones independientes del perfil. Pueden definirse elementos adicionales en un perfil registrando los números de tipo en el IANA.

A.6.4.1 CNAME: Elemento SDES identificador de punto extremo canónico



El identificador de CNAME tiene las siguientes propiedades:

- Dado que el identificador SSRC aleatoriamente atribuido puede cambiar si se descubre un conflicto o si se reinicia un programa, el elemento CNAME ha de proporcionar la vinculación del identificador de SSRC a un identificador para la fuente, que permanece constante.
- Como el identificador de SSRC, el identificador CNAME debe también ser exclusivo entre todos los participantes dentro de una sesión RTP.
- Para proporcionar una vinculación entre múltiples herramientas de medios utilizadas por un participante en un conjunto de sesiones RTP relacionadas, el CNAME debe ser fijado para ese participante.
- Para facilitar la supervisión por terceros, el CNAME debe ser adecuado para que un programa o una persona localice la fuente.

Por tanto, el CNAME debe derivarse algorítmicamente y no introducirse manualmente, a ser posible. Para cumplir estos requisitos, debe utilizarse el siguiente formato a menos que un perfil especifique una sintaxis o semántica alternativa. El elemento CNAME debe tener el formato "user@host" u "host" si un nombre de usuario no está disponible como en sistemas de un solo usuario. Para ambos formatos "host" es el nombre de dominio perfectamente cualificado del computador principal del cual se originan los datos en tiempo real, formatados según las reglas especificadas en RFC 1034 [A-11], RFC 1035 [A-12] y 2.1 de RFC 1123 [A-13]; o la representación ASCII normalizada de la dirección numérica del computador principal en la interfaz utilizada para la comunicación RTP. Por ejemplo, la representación ASCII normalizada de una dirección IP versión 4 es "decimal de puntos", también conocida como cuadrete de puntos. Otros tipos de dirección se cree que tendrán representaciones ASCII que sean mutuamente exclusivas. El nombre de dominio plenamente cualificado es más conveniente para un observador humano y puede evitar la necesidad de enviar un elemento NAME además, pero puede ser difícil o imposible de obtener fiablemente en algunos entornos operativos. Las aplicaciones que pueden operarse en dichos entornos deben utilizar en su lugar la representación ASCII de la dirección.

Ejemplos son "doe@sleepy.megacorp.com" o "doe@192.0.2.89" para un sistema multiusuario. En un sistema sin ningún nombre de usuario, los ejemplos serían "sleepy.megacorp.com" o "192.0.2.89".

El nombre de usuario debe estar en una forma que un programa tal como "finger" o "talk" pueda utilizar, es decir, suele ser el nombre de registro cronológico en lugar del nombre personal. El nombre del computador principal no es necesariamente idéntico al de la dirección de correo electrónico del participante.

Esta sintaxis no proporcionará identificadores exclusivos para cada fuente si una aplicación permite a un usuario generar múltiples fuentes a partir de un computador principal. Dicha aplicación tendría que basarse en la SSRC para seguir identificando la fuente, o el perfil para esa aplicación tendría que especificar sintaxis adicional para el identificador CNAME.

Si cada aplicación crea su CNAME independientemente, los CNAME resultantes pueden no ser idénticos, ya que se necesitaría que proporcionasen una vinculación a través de múltiples herramientas de medios pertenecientes a un participante en un conjunto de sesiones RTP relacionadas. Si se requiere una vinculación de medios recíprocos, puede ser necesario que el CNAME de cada herramienta sea configurado externamente con el mismo valor por una herramienta de coordinación. Los redactores de aplicaciones deben ser conscientes de que las asignaciones de dirección de red privada tales como la asignación Net-10 propuesta en RFC 1597 [A-14], pueden crear direcciones de red que no sean globalmente exclusivas. Esto conduciría a CNAME no exclusivos si a los computadores principales con direcciones privadas y sin conectividad IP directa con la red Internet, se les remiten sus paquetes RTP a Internet pública mediante un traductor a nivel RTP. (Véase también RFC 1627 [A-15].) Para tratar este caso, las aplicaciones pueden proporcionar un medio de configurar un CNAME exclusivo, pero recae en el traductor la carga de trasladar CNAME de direcciones privadas a direcciones públicas si es necesario para impedir que queden expuestas direcciones privadas.

A.6.4.2 NAME: Elemento SDES nombre de usuario

Véase el apéndice I.

A.6.4.3 EMAIL: Elemento SDES dirección de correo electrónico

Véase el apéndice I.

A.6.4.4 PHONE: Elemento SDES número telefónico

Véase el apéndice I.

A.6.4.5 LOC: Elemento SDES ubicación de usuario geográfico

Véase el apéndice I.

A.6.4.6 TOOL: Elemento SDES nombre de aplicación o de herramienta

Véase el apéndice I.

A.6.4.7 NOTE: Elemento SDES notificación/situación

Véase el apéndice I.

A.6.4.8 PRIV: Elemento SDES extensiones privadas

Véase el apéndice I.

A.6.5 BYE: Paquete RTCP de despedida

Véase el apéndice I.

A.6.6 APP: Paquete RTCP definido por la aplicación

Véase el apéndice I.

A.7 Traductores y mezcladores RTP

Además de los sistemas de extremo, el RTP soporta la noción de "traductores" y "mezcladores", que podrían ser considerados como "sistemas intermedios" al nivel RTP. Aunque este soporte añade alguna complejidad al protocolo, la necesidad de estas funciones ha sido claramente establecida por experimentos con aplicaciones audio y vídeo multidifusión en Internet. En esta subcláusula figuran ejemplos de uso de traductores y mezcladores. Los mezcladores y traductores son resultado de la presencia de cortafuegos y conexiones de baja anchura de banda, los cuales es probable que permanezcan ambos.

A.7.1 Descripción general

Un traductor/mezclador RTP conecta dos o más "nubes" a nivel transporte. Cada nube suele definirse mediante un protocolo de red y de transporte común (por ejemplo, IP/UDP), dirección multidifusión o par de direcciones unidifusión, y puerto de destino a nivel transporte. (Los traductores de protocolo a nivel red, tales como IP versión 4 a IP versión 6, pueden estar presentes dentro de una nube invisiblemente al RTP.) Un sistema puede servir de traductor o mezclador para cierto número de sesiones RTP, pero cada uno se considera una entidad lógicamente separada.

A fin de evitar crear un bucle cuando se instala un traductor o un mezclador, deben observarse las siguientes reglas:

- Cada una de las nubes conectadas por traductores y mezcladores que participan en una sesión RTP deben ser distintas de todas las demás en al menos uno de estos parámetros (protocolo, dirección, puerto) o bien deben estar aisladas de las demás a nivel de red.
- Una derivación de la primera regla es que no debe haber múltiples traductores ni mezcladores conectados en paralelo a menos que por cierto arreglo dividan el conjunto de fuentes a remitir.

Análogamente, todos los sistemas de extremo RTP que puedan comunicar mediante uno o más traductores o mezcladores RTP comparten el mismo espacio SSRC, es decir, los identificadores de SSRC deben ser exclusivos entre todos estos sistemas de extremo. En A.8.2, Resolución de colisiones y detección de bucles, se describe el algoritmo de resolución de colisiones mediante el cual los identificadores SSRC se mantienen exclusivos y se detectan bucles.

Puede haber muchas variedades de traductores y mezcladores diseñados para diferentes fines y aplicaciones. Algunos ejemplos son añadir o suprimir criptación, cambiar la codificación de los datos o los protocolos subyacentes, o hacer reproducciones entre una dirección multidifusión y una o más direcciones unidifusión. La distinción entre traductores y mezcladores es que un traductor pasa a través de las corrientes de datos desde diferentes fuentes por separado, mientras que un mezclador los combina para formar un nuevo tren:

Traductor: Remite paquetes RTP con su identificador de SSRC intacto; esto hace posible que los receptores identifiquen fuentes individuales aun cuando paquetes procedentes de todas las fuentes atraviesen el mismo traductor y transporten la misma dirección de fuente de red del traductor. Algunas clases de traductores harán pasar los datos intactos, pero otras pueden cambiar la codificación de los datos y por tanto el tipo de cabida útil de datos RTP y la indicación de tiempo. Si se recodifican múltiples paquetes de datos en uno, o viceversa, un traductor debe asignar nuevos números de secuencia a los paquetes salientes. Las pérdidas en el tren de paquetes entrante puede producir vacíos correspondientes en los números secuenciales salientes. Los receptores no pueden detectar la presencia de un traductor a menos que conozcan por algún otro medio qué tipo de cabida útil o dirección de transporte utilizó la fuente original.

Mezclador: Recibe trenes de paquetes de datos RTP de una o más fuentes, posiblemente cambia el formato de datos, combina los trenes de alguna manera y luego remite el tren combinado. Dado que la temporización entre múltiples fuentes de entrada no estará generalmente sincronizada, el mezclador hará ajustes de temporización entre los trenes y generará su propia temporización para el

tren combinado, por lo que es la fuente de sincronización. Así, todos los paquetes de datos remitidos por un mezclador estarán marcados con el propio identificador de SSRC del mezclador. A fin de preservar la identidad de las fuentes originales que contribuyen al paquete mixto, el mezclador debe insertar sus propios identificadores de SSRC en la lista de identificadores a continuación del encabezamiento RTP fijo del paquete. Un mezclador que tiene también él mismo una fuente contribuyente para algún paquete debe explícitamente incluir sus propios identificador de SSRC en la lista de CSRC para ese paquete.

En algunas aplicaciones, puede ser aceptable que un mezclador no identifique fuentes en la lista de CSRC. Sin embargo, esto introduce el peligro de que no puedan detectarse los bucles en los que intervienen estas fuentes.

La ventaja de un mezclador sobre un traductor para aplicaciones como el audio es que la anchura de banda de salida estará limitada a la de una fuente, aun cuando haya múltiples fuentes activas en el lado entrada. Esto puede ser importante en los enlaces de pequeña anchura de banda. El inconveniente es que los receptores en el lado salida no tienen ningún control sobre qué fuentes se dejan pasar o son silenciadas, a menos que se introduzca algún mecanismo de control distante del mezclador. La regeneración de la información de sincronización por los mezcladores también significa que los receptores no pueden hacer sincronización intermedios de los trenes originales. Un mezclador multimedios podría hacerlo.

En la figura A.3 se muestra un conjunto de mezcladores y traductores para ilustrar su efecto en los identificadores de SSRC y CSRC. En la figura, los sistemas de extremo se muestran como rectángulos (denominados E), los traductores como rombos (denominados T) y los mezcladores como óvalos (denominados M). La notación "M1:48(1, 17)" designa un paquete que origina un mezclador M1, identificado con un valor SSRC de M1 (aleatorio) de 48 y dos identificadores CSRC (1 y 17), copiados de los identificadores SSRC de paquetes procedentes de E1 y E2.

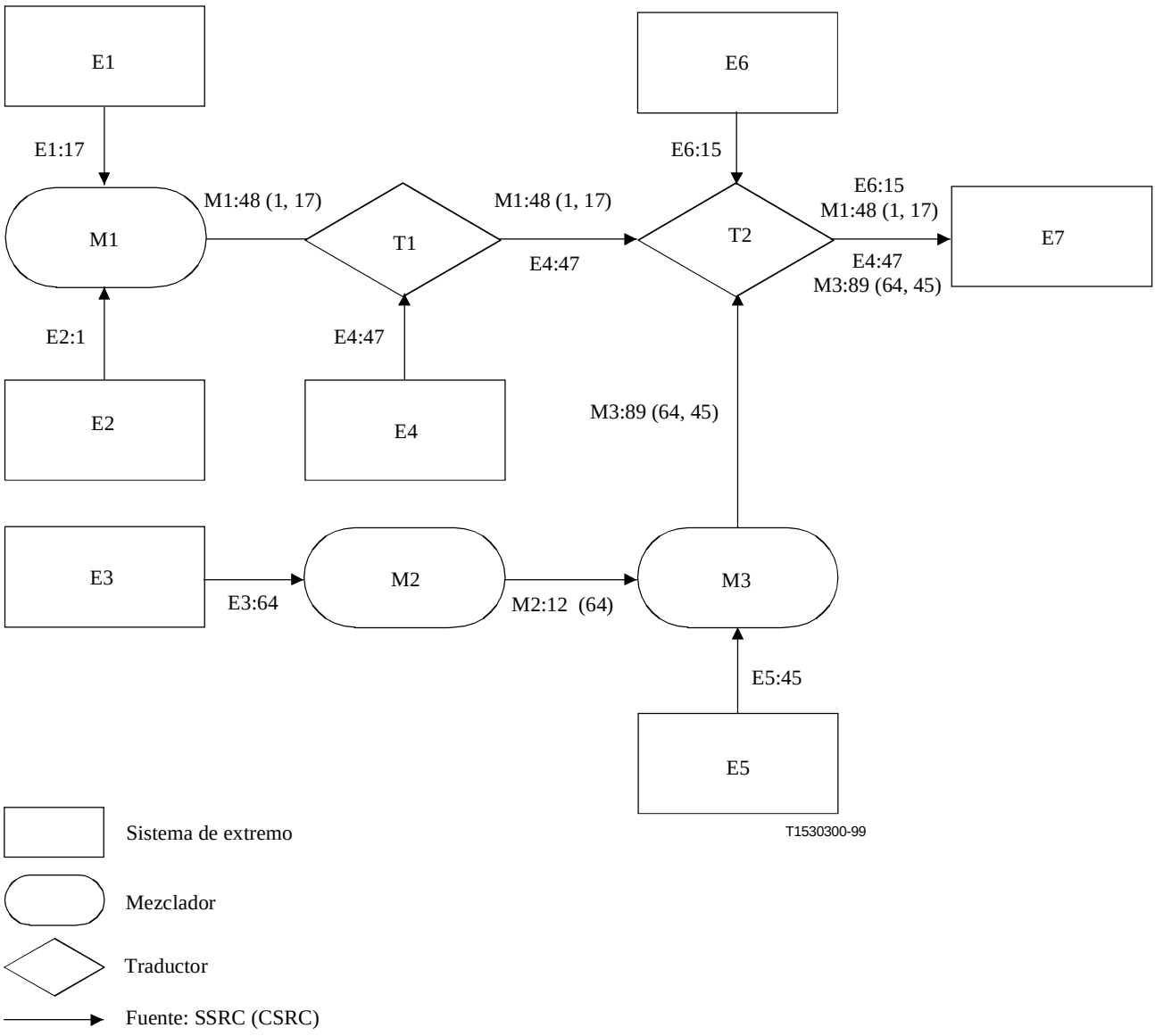


Figura A.3/H.225.0 – Ejemplo de red RTP con sistemas de extremo, mezcladores y traductores

A.7.2 Procesamiento RTCP en los traductores

Además de enviar paquetes de datos, quizá modificados, los traductores y mezcladores deben también procesar paquetes RTCP. En muchos casos tomarán aparte los paquetes RTCP compuestos recibidos de los sistemas de extremo para agregar información SDES y modificar los paquetes RR o RR. La retransmisión de esta información puede ser desencadenada por la llegada de paquetes o por el temporizador de intervalos RTCP del traductor o del propio mezclador.

Un traductor que no modifique los paquetes de datos, por ejemplo, uno que simplemente hace una reproducción entre una dirección multidifusión y una dirección unidifusión, puede simplemente remitir paquetes RTCP también sin modificar. Un traductor que transforma la cabida útil debe de alguna manera hacer transformaciones correspondientes en la información SR y RR para que siga reflejando las características de los datos y la calidad de recepción. Estos traductores no deben simplemente enviar paquetes RTCP. En general, un traductor no debe combinar paquetes SR y RR de diferentes fuentes en un paquete ya que ello reduciría la exactitud de las mediciones de retardo de propagación basadas en los campos de LSR y DLSR.

Información de emisor SR: Un traductor no genera su propia información de emisor, pero remite los paquetes SR recibidos de una nube a las demás. La SSRC se deja intacta, pero la información de emisor debe ser modificada si es necesario por la traducción. Si un traductor cambia la codificación de datos, debe cambiar el campo "cuenta de bytes del emisor". Si también combina varios paquetes de datos en un paquete de salida, debe cambiar el campo "cuenta de paquetes del emisor". Si cambia la frecuencia de indicación de tiempo, debe cambiar el campo "indicación de tiempo RTP" en el paquete SR.

Bloques de informe de recepción SR/RR: Un traductor remite informes de recepción recibidos de una nube a las demás. Adviértase que éstos fluyen en sentido opuesto a los datos. La SSRC se deja intacta. Si un traductor combina varios paquetes de datos en un paquete de salida, y cambia por tanto los números secuenciales, debe hacer la manipulación inversa para los campos de pérdida de paquetes y el campo "último número secuencial extendido". Éste puede ser complejo. En el caso extremo, puede no haber ningún modo significativo de traducir los informes de recepción, por lo que el traductor no debe pasar ningún informe de recepción en absoluto ni un informe sintético basado en su propia recepción. La regla general es hacer lo que tenga sentido en una traducción determinada.

Un traductor no necesita un identificador de SSRC propio, pero puede decidir asignar uno con el fin de enviar informes sobre lo que ha recibido. Éstos se enviarían a todas las nubes conectadas, cada una correspondiente a la traducción del tren de datos que se envía a esa nube, ya que los informes de recepción son normalmente multidifundidos a todos los participantes.

SDES: Los traductores suelen transmitir sin modificación la información SDES que reciben de una nube a las demás, pero pueden, por ejemplo, decidir filtrar información no CNAME SDES si la anchura de banda es limitada. Los CNAME deben transmitirse para permitir que funcione la detección de colisiones de identificadores SSRC. Un traductor que genere sus propios paquetes RR debe enviar información SDES CNAME sobre sí mismo a las mismas nubes a las que envía esos paquetes RR.

BYE: Los traductores transmiten los paquetes BYE sin modificación. Los traductores con su propia SSRC deben generar paquetes BYE con ese identificador SSRC si están a punto de cesar el envío de paquetes.

APP: Los traductores envían los paquetes APP sin modificación.

A.7.3 Procesamiento RTCP en los mezcladores

Dado que un mezclador genera un nuevo tren de datos propio, no pasa paquetes SR ni RR en absoluto, y en su lugar genera nueva información para ambos lados.

Información de emisor SR: Un mezclador no pasa información de emisor de las fuentes que mezcla debido a que las características de los trenes de fuentes se pierden en la mezcla. Como fuente de sincronización, el mezclador genera sus propios paquetes SR con información de emisor acerca del tren de datos mixto y los envía en el mismo sentido que el tren mixto.

Bloques de informes de recepción SR/RR: Un mezclador genera sus propios informes de recepción para las fuentes en cada nube y los envía solamente a la misma nube. No envía estos informes de recepción a las demás nubes ni remite informes de recepción de una nube a las demás, debido a que las fuentes no serían SSRC (sólo CSRC).

SDES: Los mezcladores suelen remitir sin modificación la información SDES que reciben de una nube a las demás, pero pueden, por ejemplo, decidir si filtrar información no CNAME SDES si la anchura de banda es limitada. Los CNAME deben remitirse para permitir funcionar la detección de colisiones de identificadores de SSRC. (Un identificador de una lista de CSRC generado por un mezclador podría colisionar con un identificador de SSRC generado por un sistema de extremo.) Un mezclador debe enviar información SDES CNAME sobre sí mismo a las mismas nubes a las que envía paquetes SR o RR.

Dado que los mezcladores no remiten paquetes SR o RR, normalmente extraerán paquetes SDES de un paquete RTCP compuesto. Para reducir al mínimo la tara, los pedazos de los paquetes SDES pueden combinarse en un único paquete SDES, que es entonces apilado en un paquete SR o RR originario del mezclador. La velocidad de paquetes RTCP puede ser diferente en cada lado del mezclador.

Un mezclador que no inserta identificadores CSRC puede también abstenerse de remitir SDES CNAME. En este caso, los espacios de identificador de SSRC en las dos nubes son independientes. Como se ha indicado antes, este modo de operación crea peligro de que no puedan detectarse los bucles.

BYE: Los mezcladores necesitan remitir paquetes BYE. Deberían generar paquetes BYE con sus propios identificadores SSRC si están a punto de cesar el envío de paquetes.

APP: El tratamiento de los paquetes APP por los mezcladores es específico de la aplicación.

A.7.4 Mezcladores en cascada

En una sesión RTP puede intervenir un conjunto de mezcladores y traductores, como se muestra en la figura A.3. Si dos mezcladores están en cascada, tales como M2 y M3 en la figura, los paquetes recibidos por un mezclador pueden ya haberse mezclado y pueden incluir una lista de CSRC con múltiples identificadores. El segundo mezclador debe construir la lista de CSRC para el segundo paquete saliente utilizando los identificadores de CSRC de paquetes de entrada ya mezclados y los identificadores de SSRC de paquetes de entrada no mezclados. Esto se muestra en el arco de salida del mezclador M3 designado por M3:89 (64, 45) en la figura A.3. Como en el caso de mezcladores que no están en cascada, si la lista de CSRC resultante tiene más de 15 identificadores, no puede incluirse el resto.

A.8 Asignación y utilización de identificadores de SSRC

El identificador de SSRC transportado en el encabezamiento RTP y en diversos campos de paquetes RTCP es un número de 32 bits aleatorio que es necesario que sea globalmente exclusivo dentro de una sesión RTP. Es crucial que el número se elija con cuidado a fin de que sea improbable que participantes en la misma red o que comienzan al mismo tiempo elijan el mismo número.

No es suficiente utilizar la dirección de red local (tal como una dirección IPv4) para el identificador, debido a que la dirección puede no ser exclusiva. Dado que los traductores y mezcladores RTP posibilitan la interoperación entre múltiples redes con diferentes espacios de dirección, los patrones de asignación para direcciones dentro de dos espacios podrían producir una tasa mucho más alta de colisiones que la que se produciría con asignación aleatoria.

Habría también dificultades si múltiples fuentes operan en un computador principal.

No es suficiente obtener un identificador de SSRC simplemente mediante llamada aleatoria () sin inicializar cuidadosamente el estado. En A.8.2 se presenta un ejemplo de cómo generar un identificador aleatorio.

A.8.1 Probabilidad de colisión

Como los identificadores se eligen aleatoriamente, es posible que dos o más fuentes elijan el mismo número. Se producen colisiones con la máxima probabilidad cuando todas las fuentes se arrancan simultáneamente, por ejemplo, cuando son desencadenados automáticamente por algún evento de gestión de sesión. Si N es el número de fuentes y L la longitud del identificador (aquí, 32 bits), la probabilidad de que dos fuentes escojan independientemente el mismo valor puede aproximarse para grandes N [20] a $1-\exp\left(-\frac{N^2}{2^{L+1}}\right)$. Para N = 1000, la probabilidad es aproximadamente 10^{-4}.

La probabilidad de colisión típica es mucho menor que el caso más desfavorable arriba citado. Cuando se incorpora una nueva fuente a una sesión RTP en las que todas las demás fuentes ya tienen identificadores exclusivos, la probabilidad de colisión es simplemente la fracción de los números utilizados fuera del espacio. También ahora, si N es el número de fuentes y L la longitud del identificador, la probabilidad de colisión es $\frac{N}{2^L}$. Para N = 1000, la probabilidad es aproximadamente $2 \cdot 10^{-7}$. La probabilidad de colisión es reducida aún más por la oportunidad de que una nueva fuente reciba paquetes de otros participantes antes de enviar su primer paquete (de datos o de control). Si la nueva fuente sigue la pista de los demás participantes (mediante un identificador de SSRC), antes de transmitir su primer paquete la nueva fuente puede entonces verificar que su identificador no está en contradicción con cualquiera que ha sido recibido, o de otro modo elige de nuevo.

A.8.2 Resolución de colisiones y detección de bucles

Aunque la probabilidad de colisión de los identificadores de SSRC es baja, todas las implementaciones RTP deben estar preparadas para detectar colisiones y ejercer las acciones apropiadas para resolverlas. Si una fuente descubre en cualquier momento que otra fuente está utilizando su mismo identificador de SSRC, debe enviar un paquete RTCP BYE para el antiguo identificador y elegir otro aleatorio. Si un receptor descubre que otras dos fuentes están en colisión, puede conservar los paquetes de una y descartar los paquetes de la otra cuando esto se detecte mediante diferentes direcciones de transporte de fuente o CNAME. Se espera que las dos fuentes resuelvan la colisión para que la situación no dure.

Debido a que los identificadores aleatorios se mantienen globalmente exclusivos para cada sesión RTP, pueden también utilizarse para detectar bucles que puedan ser introducidos por mezcladores o traductores. Un bucle produce duplicación de la información de datos y de control, ya sea no modificada o posiblemente mixta, como en los siguientes ejemplos:

- Un traductor puede incorrectamente remitir un paquete al mismo grupo multidifusión del cual ha recibido el paquete, sea directamente o a través de una cadena de traductores. En ese caso, el mismo paquete aparece varias veces, originario de diferentes fuentes de red.
- Dos traductores incorrectamente establecidos en paralelo, es decir, con los mismos grupos multidifusión en ambos lados, transmitirían ambos paquetes de un grupo multidifusión al otro; los traductores unidireccionales producirían dos copias; los traductores bidireccionales formarían un bucle.
- Un mezclador puede cerrar un bucle transmitiendo al mismo destino de transporte que recibe paquetes, sea directamente o a través de otro mezclador o traductor. En este caso, una fuente podría aparecer como una SSRC en un paquete de datos o como una CSRC en un paquete de datos.

Una fuente puede descubrir que sus propios paquetes están siendo bucleados, o que lo están siendo paquetes de otra fuente (un bucle de terceros). Tanto los bucles como las colisiones en la selección aleatoria de un identificador de fuente dan lugar a paquetes que llegan con el mismo identificador de SSRC, pero una dirección de transporte de fuente diferente, que puede ser la del sistema de extremo que origina el paquete o un sistema intermedio. Consiguientemente, si una fuente cambia su dirección de transporte de fuente, debe también elegir un nuevo identificador de SSRC para evitar que se interprete como una fuente bucleada. Los bucles o colisiones que se producen en el extremo distante de un traductor o un mezclador no pueden detectarse utilizando la dirección de transporte de fuente si todas las copias de los paquetes pasan por el traductor o el mezclador, pero sin embargo pueden todavía detectarse colisiones cuando los pedazos de los paquetes RTCP SDES contienen el mismo identificador de SSRC pero diferentes CNAME.

Para detectar y resolver estos conflictos, una implementación RTP debe incluir un algoritmo similar al descrito a continuación. Ignora los paquetes procedentes de una nueva fuente o bucle que colisionan con una fuente establecida. Resuelve colisiones con el propio identificador de SSRC del

participante enviando un RTCP BYE para el antiguo identificador y eligiendo uno nuevo. Sin embargo, cuando la colisión fue inducida por un bucle de los propios paquetes del participante, el algoritmo elegirá un nuevo identificador sólo una vez, y después ignorará los paquetes procedentes de la dirección de transporte de la fuente bucleante. Esto es necesario para evitar una riada de paquetes BYE.

Este algoritmo depende de que la dirección de transporte de fuente sea la misma para ambos paquetes RTP y RTCP procedentes de una fuente. El algoritmo exigiría modificaciones para soportar aplicaciones que no cumplan esta constricción.

Este algoritmo exige mantener un cuadro indexado por identificadores de fuente y que contenga la dirección de transporte de fuente a partir de la cual se recibió (primero) el identificador, junto con otro estado para esa fuente. Cada identificador de SSRC o CSRC recibido en un paquete de datos o de control se consulta en este cuadro a fin de procesar esa información de datos o de control. Para los paquetes de control, cada elemento con su propia SSRC, por ejemplo un pedazo de SDES, requiere una consulta separada. (La SSRC de un bloque de informe de recepción es una excepción.) Si no se halla la SSRC o CSRC, se crea una nueva entrada. Estas entradas del cuadro se suprimen cuando se recibe un paquete RTCP BYE con la correspondiente SSRC, o después de que no hayan llegado paquetes durante un tiempo relativamente largo (véase A.6.2.1, Mantenimiento del número de miembros de sesión).

A fin de seguir la pista de los paquetes de datos propios del participante, es también necesario mantener una lista separada de direcciones de transporte de fuente (no identificadores), que se haya visto que están en contradicción. Adviértase que ésta sería una lista corta, normalmente vacía. Cada elemento de esta lista almacena la dirección de fuente más la hora a la que se recibió el paquete contradictorio más reciente. Un elemento puede suprimirse de la lista cuando no haya llegado ningún paquete contradictorio procedente de esa lista durante un tiempo del orden de 10 intervalos de informe RTCP (véase A.6.2, Intervalo de transmisión RTCP).

Para el algoritmo que se presenta, se supone que el propio identificador de fuente del participante se incluye en el cuadro de identificadores de fuente. El algoritmo podría reestructurarse para establecer primero una comparación separada con el propio identificador de fuente del participante.

Si el identificador de SSRC o CSRC no se halla en el cuadro de identificadores de fuente:

ENTONCES crear una nueva entrada que almacena la dirección de transporte de origen y SSRC o CSRC junto con otro estado.

CONTINUAR con el procesamiento normal.

(El identificador se halla en el cuadro.)

SI la dirección de transporte de origen del paquete concuerda con la salvada en la entrada del cuadro para este identificador:

CONTINUAR ENTONCES con el procesamiento normal.

(Se indica una colisión de identificadores o un bucle.)

SI el identificador de origen no es el propio del participante:

ENTONCES SI el identificador de origen es de un pedazo RTCP SDES que contiene un elemento CNAME que difiere del CNAME en la entrada del cuadro.

ENTONCES (facultativamente) contar una colisión de un tercero.

EN OTRO CASO (facultativamente) contar una colisión de un tercero.

ABORTAR el procesamiento del paquete de datos o del elemento de control.

(Una colisión o bucle de los propios datos del participante.)

SI la dirección de transporte de origen figura en la lista de direcciones contradictorias:

ENTONCES SI el identificador de origen no es de un pedazo RTCP SDES que contiene un elemento CNAME o si ese CNAME es propio del participante:

> ENTONCES (facultativamente) contar la ocurrencia del propio tráfico bucleado. Marcar la hora en la entrada de la lista de direcciones contradictorias.
>
> ABORTAR el procesamiento de paquete de datos o elemento de control.

Registrar cronológicamente la ocurrencia de una colisión.

Crear una nueva entrada en la lista de direcciones contradictorias y marcar la hora.

Enviar un paquete RTCP BYE con el antiguo identificador SSRC.

Elegir un nuevo identificador.

Crear una nueva entrada en el cuadro de identificadores de origen con el antiguo SSRC más la dirección de transporte de origen del paquete que se está procesando.

CONTINUAR con el procesamiento normal.

En este algoritmo, los paquetes de una dirección de fuente recientemente contrapuesta serán ignorados y se conservarán los paquetes procedentes de la fuente original. (Si la fuente original era a través de un mezclador y posteriormente la misma fuente se recibe directamente, puede aconsejarse al receptor que conmute, a menos que se hubieran perdido otras fuentes en la mezcla.) Si no llegan paquetes desde la fuente original durante un largo periodo, la entrada del cuadro será destemporizada y la nueva fuente podrá tomar el relevo. Esto podría ocurrir si la fuente original detecta la colisión y pasa a un nuevo identificador de fuente, pero en el caso ordinario se recibirá un paquete RTCP BYE de la fuente original para suprimir el estado sin tener que esperar un tiempo muerto.

Cuando se elige un nuevo identificador de SSRC debido a una colisión, el identificador candidato debe consultarse primero en el cuadro de identificadores de fuente para ver si ya estaba en uso por alguna otra fuente, en cuyo caso, debe generarse otro candidato y repetirse el proceso.

Un bucle de paquetes de datos a un destino multidifusión puede causar una grave riada en la red. Todos los mezcladores y traductores necesitan implementar un algoritmo de detección de bucles como el aquí indicado para que puedan interrumpir los bucles. Esto debe limitar el tráfico excedente a no más de una copia duplicada del tráfico original, lo que puede permitir que la sesión continúe para que pueda determinarse y fijarse con precisión la causa del bucle. Sin embargo, en casos extremos en los que un mezclador o un traductor no interrumpen adecuadamente el bucle y se producen niveles de tráfico elevados, puede ser necesario que los sistemas de extremo cesen de transmitir datos o de controlar paquetes completamente. Esta decisión puede depender de la aplicación. Debe indicarse una condición de error si así conviene. La transmisión podría intentarse una vez más periódicamente después de un largo tiempo aleatorio (del orden de minutos).

A.9 Seguridad

En el apéndice I puede verse una descripción informativa de algunos métodos de seguridad Internet. La privacidad H.323 y los métodos de intercambio de claves se describen en la Recomendación H.323.

A.10 RTP sobre los protocolos de red y de transporte

Esta subcláusula describe aspectos específicos del transporte de paquetes RTP dentro de determinados protocolos de red y de transporte. Se aplican las siguientes reglas a menos que sean anuladas y reemplazadas por definiciones específicas del protocolo que quedan fuera del alcance de esta especificación.

El RTP se sirve del protocolo o protocolos subyacentes para proporcionar demultiplexación de trenes de datos RTP y de control RTCP. Para el UDP y protocolos similares, el RTP utiliza un número de puerto par y el correspondiente tren RTCP utiliza el número de puerto inmediatamente superior (impar). Si a una aplicación se le suministra un número impar para su utilización como el puerto RTP, debe sustituir este número por el inmediatamente inferior (par).

Los paquetes de datos RTP no contienen ningún campo de longitud ni otra descripción, por lo cual el RTP se sirve de los protocolos subyacentes para proporcionar una indicación de longitud. La longitud máxima de los paquetes RTP está limitada únicamente por los protocolos subyacentes.

Si han de transportarse paquetes RTP en un protocolo subyacente que permite la abstracción de un tren de octetos continuo en vez de mensajes (paquetes), debe definirse un encapsulado de los paquetes RTP para proporcionar un mecanismo de entramación. La entramación también es necesaria si el protocolo subyacente puede contener relleno a fin de que no pueda determinarse la extensión de la cabida útil RTP. El mecanismo de entramación no se define aquí.

Un perfil puede especificar un método de entramación a utilizar aun cuando el RTP sea transportado en protocolos que no permiten entramación a fin de transportar varios paquetes RTP en una unidad de datos de protocolo de capa inferior, como es un paquete UDP. Transportar varios paquetes RTP en un paquete de red o de transporte reduce la tara y puede simplificar la sincronización entre diferentes trenes.

A.11 Sumario de constantes de protocolo

Se incluye en esta subcláusula un sumario de las constantes definidas en esta especificación.

Las constantes de tipo de cabida útil (PT, *payload type*) RTP se definen en perfiles más que en esta Recomendación. Sin embargo, el octeto del encabezamiento RTP que contiene el bit (o bits) marcador y el tipo de cabida útil debe evitar los valores reservados 200 y 201 (decimales) para distinguir los paquetes RTP de los tipos de paquetes RTCP SR y RR para el procedimiento de validación de encabezamiento descrito en A.6.3. Para la definición normalizada de un bit marcador y un campo de tipo de cabida útil de 7 bits, que se muestra en esta especificación, esta restricción significa que se reservan los tipos de cabida útil 72 y 73.

A.11.1 Tipos de paquetes RTCP

Abreviatura	Nombre	Valor
SR	Informe de emisor	200
RR	Informe de receptor	201
SDES	Descripción de fuente	202
BYE	Adiós	203
APP	Definido por la aplicación	204

Estos valores tipo se eligieron en la gama 200-204 para una mejor comprobación de la validez del encabezamiento de los paquetes RTCP en comparación con paquetes RTP u otros paquetes no relacionados. Cuando el campo de tipo de paquete RTCP se compara con el octeto correspondiente del encabezamiento RTP, esta gama corresponde a que el bit marcador sea 1 (lo que no suele ocurrir en los paquetes de datos) y que el bit superior del campo tipo de cabida útil normalizado sea 1 (ya que los tipos de cabida útil estática suelen definirse en la mitad baja). Esta gama también se eligió para que esté a cierta distancia numéricamente de 0 y 255, ya que todos ceros y todos unos son patrones de datos comunes.

Como todos los paquetes RTCP compuestos deben comenzar por SR o RR, estos códigos se eligieron como una pareja par/impar para permitir la comprobación de validez RTCP para probar el número máximo de bits con plantilla y valor.

Otras constantes son asignadas por IANA. Se alienta a los experimentadores a registrar los números que necesiten para sus experimentos, y luego desregistren los que resulten innecesarios.

A.11.2 Tipos de SDES

Abreviatura	Nombre	Valor
END	Fin de lista de SDES	0
CNAME	Nombre canónico	1
NAME	Nombre de usuario	2
EMAIL	Dirección de correo electrónico del usuario	3
PHONE	Número de teléfono del usuario	4
LOC	Ubicación geográfica del usuario	5
TOOL	Nombre de aplicación o herramienta	6
NOTE	Información sobre la fuente	7
PRIV	Extensiones privadas	8

Otras constantes son asignadas por IANA. Se alienta a los experimentadores a registrar los números que necesiten para sus experimentos, y luego desregistren los que resulten innecesarios.

A.12 Perfiles RTP y especificaciones de formato de cabida útil

Una especificación completa de RTP para una determinada aplicación exigirá uno o más documentos acompañantes de dos tipos aquí descritos: perfiles, y especificaciones de formato de cabida útil.

RTP puede utilizarse para una variedad de aplicaciones con requisitos algo diferentes. La flexibilidad para adaptarse a estos requisitos se proporciona permitiendo múltiples opciones en la especificación de protocolo principal, y luego seleccionando las opciones adecuadas o definiendo extensiones para un entorno determinado y clases de aplicaciones en un documento de perfil separado. Una aplicación solerá operar bajo un solo perfil, por lo que no hay ninguna explicación explícita de qué perfil se utiliza. En el anexo B puede verse un perfil para aplicaciones audio y vídeo.

El segundo tipo de documento acompañante es una especificación de formato de cabida útil, que define cómo debe transportarse en RTP una determinada clase de datos de cabida útil, tales como vídeo con codificación H.261. Estos documentos suelen titularse "RTP Payload Format for XYZ Audio/Video Encoding" ("Formato de cabida útil RTP para codificación audio/vídeo XYZ"). Los formatos de cabida útil pueden ser de utilidad bajo múltiples perfiles y pueden por tanto definirse independientemente de cualquier perfil determinado. Los documentos de perfiles se encargan entonces de asignar una correspondencia por defecto de ese formato a un valor de tipo de cabida útil si es necesario. Véase esta información en el anexo C.

Dentro de esta especificación, se han identificado los siguientes elementos para su posible definición dentro de un perfil, pero esta lista no pretende ser exhaustiva:

Encabezamiento de datos RTP: El octeto del encabezamiento de datos RTP que contiene el bit marcador y el campo de tipo de cabida útil pueden ser definidos por un perfil para adecuarse a requisitos diferentes, por ejemplo, con más o menos bits marcadores (véase A.5.3, Modificaciones específicas del perfil en el encabezamiento RTP).

Tipos de cabida útil: Suponiendo que se incluya un campo de tipo de cabida útil, el perfil definirá usualmente un conjunto de formatos de cabida útil (por ejemplo, codificaciones de medios) y una

correspondencia estática por defecto de estos formatos a valores de tipo de cabida útil. Algunos de los formatos de cabida útil pueden definirse por referencia a especificaciones de formatos de cabida útil separados. Para cada tipo de cabida útil definido, el perfil debe especificar la cadencia de indicaciones de tiempo a utilizar (véase A.5.1, Campos de encabezamiento fijo RTP).

Adiciones al encabezamiento de datos RTP: Pueden agregarse campos adicionales al encabezamiento de datos RTP si se requiere alguna funcionalidad adicional en la clase de aplicaciones de perfil independientes del tipo de cabida útil (véase A.5.3, Modificaciones específicas del perfil en el encabezamiento RTP).

Extensiones del encabezamiento de datos RTP: El contenido de los 16 primeros bits de la estructura de extensión del encabezamiento de datos RTP debe definirse si ha de permitirse el uso de ese mecanismo bajo el perfil para las extensiones específicas de la implementación (véase A.5.3, Modificaciones específicas del perfil en el encabezamiento RTP).

Tipos de paquetes RTCP: Pueden definirse y registrarse en IANA nuevos tipos de paquetes RTCP específicos de la clase de aplicación.

Intervalo de informe RTCP: Un perfil debe especificar que se utilizarán los valores sugeridos en A.6.2, Intervalo de transmisión RTCP, para las constantes empleadas en el cálculo del intervalo de informe RTCP. Son éstos la fracción RTCP de la anchura de banda de sesión, el mínimo intervalo de informe, y la anchura de banda dividida entre emisores y receptores. Un perfil puede especificar valores alternativos si han demostrado que operan de manera escalable.

Extensión de SR/RR: Puede definirse una sección de extensión para los paquetes RTCP SR y RR si existe información adicional que deba informarse regularmente acerca del emisor o los receptores (véase A.6.3.3, Extensión de los informes de emisor y de receptor).

Utilización de SDES: Este perfil puede especificar las prioridades relativas para que los RTCP SDES sean transmitidos o excluidos totalmente (véase A.6.2.2, Asignación de anchura de banda de descripción de fuente); una sintaxis o una semántica alternativas para el elemento CNAME (véase A.6.4.1, CNAME: Elemento SDES identificador de punto extremo canónico); el formato del elemento LOC (véase A.6.4.5, LOC: Elemento SDES ubicación de usuario geográfico); la semántica y utilización del elemento NOTE (véase A.6.4.7, NOTE: Elemento SDES notificación/situación); o nuevos tipos de elemento SDES a registrar en IANA.

Seguridad: Un perfil puede especificar qué servicios de seguridad y algoritmos deben ser ofrecidos por las aplicaciones, y puede proporcionar orientación en cuanto a su uso apropiado (véase A.9, Seguridad).

Correspondencia cadena-clave: Un perfil puede especificar cómo se hace corresponder una contraseña o locución de paso en una clave de criptación.

Protocolo subyacente: Utilización de un determinado protocolo de capa de red o de transporte subyacente para transportar paquetes RTP.

Correspondencia de transporte: Puede especificarse una correspondencia de direcciones RTP y RTCP a direcciones de nivel transporte, por ejemplo, puertos UDP, distinta de la correspondencia normalizada definida en el anexo B.

Encapsulado: Puede definirse un encapsulado de paquetes RTP para permitir el transporte de múltiples paquetes de datos RTP en un paquete de capa inferior o para proporcionar entramación sobre protocolos subyacentes que todavía no lo hacen (véase A.10, RTP sobre los protocolos de red y de transporte).

A.13 Algoritmos

Esta subcláusula puede verse en el apéndice I. Todas esas implementaciones de muestra son no normativas, por lo cual no se incluyen aquí.

A.14 Bibliografía

Adviértase que el material de esta bibliografía es informativo, y no es necesario implementarlo en este anexo.

[A-1] CLARK (D.D.), TENNENHOUSE (D.L.): Architectural considerations for a new generation of protocols, *SIGCOMM Symposium on Communications Architectures and Protocols*, (Philadelphia, Pennsylvania), pp. 200-208, *IEEE*, septiembre de 1990. Computer Communications Review, Vol. 20 (4), septiembre de 1990.

[A-2] COMER (D.E.): Internetworking with TCP/IP, Vol. 1, *Prentice Hall*, Englewood Cliffs, New Jersey 1991.

[A-3] POSTEL (J.): Internet protocol, RFC 791, *Internet Engineering Task Force*, septiembre de 1981.

[A-4] MILLS (D.): Network time protocol (v3), RFC 1305, *Internet Engineering Task Force*, abril de 1992.

[A-5] EASTLAKE (D.), CROCKER (S.), SCHILLER (J.): Randomness recommendations for security, RFC 1750, *Internet Engineering Task Force*, diciembre de 1994.

[A-6] BOLOT (J.-C.), TURLETTI (T.), WAKEMAN (I.): Scalable feedback control for multicast video distribution in the internet, *SIGCOMM Symposium on Communications Architectures and Protocols*, pp. 58-67, ACM, London, agosto de 1994.

[A-7] BUSSE (I.), DEFFNER (B.), SCHULZRINNE (H.): Dynamic QOS control of multimedia applications based on RTP, *Computer Communications*, enero de 1996.

[A-8] FLOYD (S.), JACOBSON (V.): The synchronization of periodic routing messages, *SIGCOMM Symposium on Communications Architectures and Protocols* (D. P. Sidhu, ed.,), pp. 33-44, ACM, (San Francisco, California) septiembre de 1993.

[A-9] CADZOW (J.A.): Foundations of digital signal processing and data analysis, *Macmillan* New York 1987.

[A-10] ISO/CEI 10646-1:1993, *Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane.*

[A-11] MOCKAPETRIS (P.): Domain names – Concepts and facilities, STD 13, RFC 1034, *Internet Engineering Task Force*, noviembre de 1987.

[A-12] MOCKAPETRIS (P.): Domain names – Implementation and specification, STD 13, RFC 1035, *Internet Engineering Task Force*, noviembre de 1987.

[A-13] BRADEN (R.): Requirements for internet hosts – Application and support, STD 3, RFC 1123, *Internet Engineering Task Force*, octubre de 1989.

[A-14] REKHTER (Y.), MOSKOWITZ (R.), KARRENBERG (D.), DE GROOT (G.): Address allocation for private internets, RFC 1597, *Internet Engineering Task Force*, marzo de 1994.

[A-15] LEAR (E.), FAIR (E.), CROCKER (D.), KESSLER (T.): Network 10 considered harmful (some practices should not be codified), RFC 1627, *Internet Engineering Task Force*, julio de 1994.

[A-16] CROCKER (D.): Standard for the format of ARPA internet text messages, STD 11, RFC 822, *Internet Engineering Task Force*, agosto de 1982.

[A-17] FELLER (W.): An Introduction to Probability Theory and its Applications, Vol. 1, John Wiley and Sons, third ed., New York 1968.

[A-18] BALENSON (D.): Privacy enhancement for internet electronic mail: Part III: algorithms, modes, and identifiers, RFC 1423, *Internet Engineering Task Force*, febrero de 1993.

[A-19] VOYDOCK (V.L.), KENT (S.T.): Security mechanisms in high-level network protocols, *ACM Computing Surveys* , Vol. 15, pp. 135-171, junio de 1983.

[A-20] RIVEST (R.): The MD5 message-digest algorithm, RFC 1321, *Internet Engineering Task Force*, abril de 1992.

ANEXO B

Perfil RTP

Véase la introducción al anexo A; todas las advertencias allí indicadas se aplican igualmente a este anexo. En el apéndice II puede verse una referencia normativa al documento IETF completo; sin embargo, este anexo contiene toda la información necesaria para la implementación de la Recomendación H.323.

B.1 Introducción

Este perfil define aspectos de RTP que no se especificaron en el anexo A. Este perfil está destinado a su utilización en audioconferencias y videoconferencias con mínimo control de sesión. En particular, no soporta la negociación de parámetros ni el control de los participantes. El perfil se cree que será de utilidad en sesiones en las que no se aplica negociación ni control de participantes (por ejemplo, utilizando los tipos de cabida útil estática y las indicaciones de participación proporcionadas por el RTCP), pero este perfil puede ser de utilidad en unión de un protocolo de control de nivel superior.

La utilización de este perfil se produce mediante el uso de aplicaciones apropiadas; no hay ninguna indicación implícita por número de puerto, identificador de protocolo o similar.

Otros perfiles pueden hacer diferentes selecciones para los elementos aquí especificados.

B.2 Formas de paquetes RTP y RTCP y comportamiento de protocolo

La subcláusula "Perfiles RTP y especificación de formato de cabida útil" enumera cierto número de elementos que pueden especificarse o modificarse en un perfil. En esta subcláusula se tratan estos elementos. Generalmente, este perfil sigue los aspectos por defecto y/o recomendados de la especificación RTP.

Encabezamiento de datos RTP: Se utiliza el formato normalizado del encabezamiento de datos RTP (un bit marcador).

Tipos de cabida útil: Los tipos de cabida útil estática se definen en B.6, Definiciones de tipos de cabida útil.

Adiciones al encabezamiento de datos RTP: No se agregan campos fijos adicionales al encabezamiento de datos RTP.

Extensiones del encabezamiento de datos RTP: No se definen extensiones del encabezamiento RTP, pero aplicaciones que operan bajo este perfil pueden utilizar dichas extensiones. Así, las aplicaciones no deben suponer que el bit X de encabezamiento RTP es siempre cero, y deben estar preparadas para ignorar la extensión del encabezamiento. Si se define en el futuro una extensión del encabezamiento, esa definición debe especificar el contenido de los 16 primeros bits de modo que puedan identificarse múltiples extensiones diferentes.

Tipos de paquetes RTCP: No se definen tipos de paquetes RTCP adicionales en esta especificación de perfil.

Intervalo de informe RTCP: Las constantes sugeridas han de utilizarse para el cálculo del intervalo de informe RTCP.

Extensión de SR/RR: No se define ninguna sección de extensión para el paquete RTCP SR o RR.

Utilización de SDES: Las aplicaciones pueden utilizar cualquiera de los elementos SDES descritos. Aunque la información CNAME se envía cada intervalo de informe, otros elementos deben enviarse sólo cada quinto intervalo de informe.

Seguridad: Los servicios de seguridad por defecto RTP no son el valor por defecto bajo este perfil.

Correspondencia cadena-clave: Véase en el apéndice II esta información.

Protocolo subyacente: En el apéndice IV se describe todo protocolo subyacente admitido que cumple ciertos requisitos.

Correspondencia de transporte: Se utiliza la correspondencia normalizada de direcciones RTP y RTCP a direcciones a nivel de transporte.

Encapsulado: No se especifica ningún encapsulado de paquetes RTP.

B.3 Tipos de cabida útil

Véase en el apéndice II información sobre el registro de nuevos tipos de cabida útil.

Adviértase que no es necesario asignar un tipo de cabida útil estática a todas las comunicaciones que haya de utilizar el RTP. Pueden utilizarse medios no RTP que caen fuera del alcance de este anexo (tales como protocolos de servicios de directorio o protocolos de invitación) para establecer una correspondencia dinámica entre un tipo de cabida útil extraído de la gama 96-127 y una codificación. Por conveniencia del implementador, este perfil contiene descripciones de codificaciones a las que actualmente no se les ha asignado un tipo de cabida útil estática.

El espacio de tipo de cabida útil disponible es relativamente pequeño. Por tanto, sólo se asignan nuevos tipos de cabida útil estática si se cumplen las siguientes condiciones:

- La codificación es de interés para la comunidad Internet en general.
- Presenta ventajas en comparación con codificaciones existentes y/o es necesaria para el interfuncionamiento con sistemas de conferencia o multimedios existentes de implantación muy extendida.
- La descripción es suficiente para construir un decodificador.

B.4 Audio

B.4.1 Recomendaciones independientes de la codificación

En las aplicaciones que no envían paquetes durante el silencio, el primer paquete de un arranque de palabra (primer paquete tras un periodo de silencio) se distingue fijando el bit marcador en el encabezamiento de datos RTP. Las aplicaciones sin supresión de silencio ponen el bit a cero.

La velocidad de reloj RTP utilizada para generar la indicación de tiempo RTP es independiente del número de canales y de la codificación; es igual al número de periodos de muestreo por segundo. Para codificaciones de canal N, cada periodo de muestra (por ejemplo, 1/8000 de un segundo) genera N muestras. (Ésta es terminología estándar, pero algo confusa, ya que el número total de muestras generadas por segundo es entonces la velocidad de muestreo que temporiza la cuenta de canales.)

Si se utilizan múltiples canales audio, los canales se enumeran de izquierda a derecha, empezando por uno. En los paquetes de audio RTP, la información de los canales de número más bajo precede a la de los canales de número más alto.

Para más de dos canales, el convenio exige utilizar la siguiente notación:

1 izquierda (*left*)

r derecha (*right*)

c centro (*center*)

S alrededor (*surround*)

F delante (*front*)

R detrás (*rear*)

Canales	Descripción	Canal					
		1	**2**	**3**	**4**	**5**	**6**
2	Estéreo	l	r				
3		l	r	c			
4	Cuadrifónico	Fl	Fr	R1	Rr		
4		l	c	r	S		
5		Fl	Fr	Fc	Sl	Sr	
6		l	lc	c	r	rc	S

Las muestras para todos los canales pertenecientes a un solo instante de muestreo deben estar dentro del mismo paquete. El entrelazado de muestras de diferentes canales depende de la codificación. Se dan directrices generales en B.4.2, Directrices para codificaciones de audio efectuadas con muestras.

La frecuencia de muestreo debe extraerse del conjunto: 8000, 11 025, 16 000, 22 050, 24 000, 32 000, 44 100 y 48 000 Hz. (Los computadores Apple Macintosh tienen velocidades de muestreo iniciales de 22 254,54 y 11 127,27, que pueden convertirse en 22 050 y 11 025 con calidad aceptable abandonando 4 ó 2 muestras en una trama de 20 ms.) Sin embargo, la mayoría de las codificaciones de audio se definen para un conjunto más restringido de frecuencias de muestreo. Los receptores deben estar preparados para aceptar audio multicanal, pero pueden decidir reproducir únicamente un solo canal.

Las recomendaciones siguientes son parámetros operativos por defecto. Las aplicaciones deben estar preparadas para tratar otros valores. Los valores indicados se destinan a dar orientación a los preparadores de aplicaciones, permitiendo a un conjunto de aplicaciones conformes con estas directrices interfuncionar sin negociación adicional. Estas directrices no están destinadas a restringir los parámetros operativos en aplicaciones que puedan negociar un conjunto de parámetros interoperables, por ejemplo, mediante un protocolo de control de conferencia.

Para audio paquetizado, el intervalo de paquetización por defecto debe tener una duración de 20 ms, a menos que se indique otra cosa al describir la codificación. El intervalo de paquetización determina el mínimo retardo de extremo a extremo; paquetes más largos introducen menos tara de encabezamiento pero mayor retardo, y hacen la pérdida de paquetes más apreciable. En las aplicaciones no interactivas tales como conferencias o enlaces con constricciones rigurosas de anchura de banda, puede ser apropiado un retardo de paquetización mayor. Un receptor debe aceptar paquetes que representen entre 0 y 200 ms de datos de audio. Esta restricción permite un dimensionamiento de memoria intermedia razonable en el receptor.

B.4.2 Directrices para codificaciones de audio efectuadas con muestras

En las codificaciones que utilizan muestras, cada muestra de audio se representa por un número fijo de bits. Dentro de los datos de audio comprimidos, los códigos de muestras individuales pueden abarcar fronteras de octetos. Un paquete de audio RTP puede contener cualquier número de muestras de audio, con la condición de que el número de bits por muestra multiplicado por el número de muestras por paquete arroje una cuenta de octetos entera. Las codificaciones fraccionarias producen menos de un octeto por muestra.

La duración de un paquete de audio viene determinada por el número de muestras en el paquete.

En las codificaciones efectuadas con muestras que producen uno o más octetos por muestra, las muestras procedentes de diferentes canales muestreados en el mismo instante de muestreo se

empacan en octetos consecutivos. Por ejemplo, en una codificación de dos canales, la secuencia de octetos es (canal izquierdo, primera muestra), (canal derecho, primera muestra), (canal izquierdo, segunda muestra), (canal derecho, segunda muestra). En las codificaciones multioctetos, los octetos se transmiten en orden de bytes (es decir, octeto más significativo primero).

El empacamiento de codificaciones basadas en muestras que produzcan menos de un octeto por muestra es específico de la codificación.

B.4.3 Directrices para codificaciones de audio efectuadas con tramas

Las codificaciones que utilizan tramas codifican un bloque de longitud fija de audio en otro bloque de datos comprimidos, que también suele ser de longitud fija. En las codificaciones con tramas, el emisor puede decidir combinar varias de dichas muestras en un solo mensaje. El receptor puede saber el número de tramas contenido en un mensaje, ya que la duración de trama se define como parte de la codificación.

En los códecs que utilizan tramas, el orden de los canales se define para el bloque completo. Es decir, en audio bicanal, las muestras derecha e izquierda se codifican independientemente, con la trama codificada del canal izquierdo precediendo a la del canal derecho.

Todos los códecs audio que utilizan tramas deben poder codificar y decodificar varias tramas consecutivas dentro de un solo paquete. Como el tamaño de trama en los códecs que utilizan tramas viene dado, no hay necesidad de utilizar una designación separada para la misma codificación, pero con diferente número de tramas por paquete.

B.4.4 Codificaciones de audio

Las características de las codificaciones de audio normalizadas se muestran en el cuadro B.1, y sus tipos de cabida útil se indican en el cuadro B.2.

Véase en el apéndice II información sobre cualquier codificación no enumerada en el cuadro B.1. El soporte de dichas codificaciones no forma parte de la Recomendación H.323.

Cuadro B.1/H.225.0 – Propiedades de las codificaciones de audio

Codificación	Muestra/trama	Bits/muestra	ms/trama
G722	Muestra	8	
G728	Trama	N/A	2,5
PCMA	Muestra	8	
PCMU	Muestra	8	
G723	Trama	N/A	30
G729	Trama	N/A	10
GSM	Trama	N/A	20

B.4.4.1 G722

G722 se especifica en la Recomendación G.722, "Codificación de audio de 7 kHz dentro de 64 kbit/s".

B.4.4.2 G728

G728 se especifica en la Recomendación G.728, "Codificación de señales vocales a 16 kbit/s utilizando predicción lineal con excitación por código de bajo retardo".

B.4.4.3 PCMA (MIC-A)

PCMA se especifica en la Recomendación G.711. Los datos de audio se codifican con ocho bits por muestra, previo escalamiento logarítmico.

B.4.4.4 PCMU (MIC-μ)

PCMU se especifica en la Recomendación G.711. Los datos de audio se codifican con ocho bits por muestra, previo escalamiento logarítmico.

B.5 Vídeo

Se definen actualmente las siguientes codificaciones de vídeo, con los nombres abreviados que se utilizan para su identificación. Véase en el apéndice II cualquier codificación no descrita aquí. Dicha codificación no forma parte de la Recomendación H.323.

H261

La codificación se especifica en la Recomendación H.261. La paquetización y las propiedades específicas RTP se describen en el anexo C.

H263

La codificación se especifica en la Recomendación H.263. La paquetización y las propiedades específicas RTP se describen en el anexo E.

Las entidades H.323 que deseen transmitir trenes de vídeo H.263 (1996 ó 1998) han de seguir el procedimiento que a continuación se indica:

- En un mensaje **OpenLogicalChannel (apertura de canal lógico)** de la Recomendación H.245, el emisor que desea utilizar el formato de cabida útil de legado para H.263 (1996) ampliamente utilizado en la industria señalará características H.263 (1996) solamente y omitirá h2250LogicalChannelParameters.mediaPacketization.
- En un mensaje **OpenLogicalChannel (apertura de canal lógico)** de la Recomendación H.245, el emisor que desea utilizar el formato de cabida útil para H.263 (1996) definido en RFC 2190 especificará **h2250LogicalChannelParameters.rtpPayloadType** como sigue: {rfc-number = 2190, payloadType = 34}.
- En general, en un mensaje **OpenLogicalChannel (apertura de canal lógico)** de la Recomendación H.245, el emisor especificará el formato de cabida útil de acuerdo con la semántica definida en la Recomendación H.245. Esto es algo que ha de cumplirse sobre todo al señalizar el formato de cabida útil H.263 + (1998) (definido en el anexo A) y sus posibles sucesores.

B.6 Definiciones de tipos de cabida útil

El cuadro B.2 define los valores de tipos de cabida útil estática del perfil para el campo PT del encabezamiento de datos RTP. Además, pueden definirse valores de tipo de cabida útil en la gama 96-127 dinámicamente a través de un protocolo de control de conferencia, lo que cae fuera del alcance de esta Recomendación. Por ejemplo, un directorio de sesión podría especificar que para una sesión dada, el tipo de carga útil 96 indica codificación PCMU, frecuencia de muestreo de 8000 Hz, 2 canales. La gama de tipo de cabida útil declarada "reservada" se ha dejado a un lado para que los paquetes RTCP y RTP puedan distinguirse fiablemente. (Véase A.11, Sumario de constantes de protocolo.)

Una fuente RTP emite un solo tipo de cabida útil RTP en un momento dado; no está autorizado el entrelazado de varios tipos de cabida útil RTP en una sola sesión, pero pueden utilizarse múltiples sesiones RTP en paralelo para transmitir múltiples medios. Los tipos de cabida útil actualmente definidos en este perfil transportan audio o vídeo, pero no ambos. Sin embargo, está permitido

definir tipos de cabida útil que combinen varios medios, por ejemplo, audio y vídeo, con separación apropiada en el formato de cabida útil. Los participantes en una sesión acuerdan mediante mecanismos que se salen del alcance de esta especificación el conjunto de tipos de cabida útil autorizados en una sesión. Este conjunto puede, por ejemplo, ser definido por las capacidades de las aplicaciones utilizadas, negociado por un protocolo de control de conferencia o establecido por acuerdo entre los individuos participantes.

Todas las codificaciones de vídeo utilizan una frecuencia de indicación de tiempo de 90 000 Hz, que es la misma que la frecuencia de indicación de tiempo de presentación. Esta frecuencia arroja incrementos de indicación de tiempo enteros exactos para las frecuencias de trama típicas de 24 (HDTV), 25 (PAL) y 29,97 (NTSC) y 30 Hz (HDTV) y las velocidades de campo 50, 59,94 y 60 Hz. Aunque 90 kHz es la frecuencia recomendada para futuras codificaciones de vídeo utilizadas dentro de este perfil, son posibles otras. Sin embargo, no es suficiente utilizar la frecuencia de trama (normalmente entre 15 y 30 Hz) porque no proporciona resolución adecuada para requisitos de sincronización típicos cuando se calcula la indicación de tiempo RTP correspondiente a la indicación de tiempo NTP en un paquete RTCP SR (véase el anexo A). La resolución de indicación de tiempo debe también ser suficiente para la estimación de fluctuación de fase contenida en los informes de receptor.

Las codificaciones de vídeo normalizadas y sus tipos de cabida útil se enumeran en el cuadro B.2.

Cuadro B.2/H.225.0 – Tipos de cabida útil (PT) para codificación audio y vídeo normalizadas

PT	Nombre de codificación	Audio/vídeo (A/V)	Frecuencia de reloj (Hz)	Canales (audio)
0	PCMU	A	8 000	1
8	PCMA	A	8 000	1
9	G722	A	8 000	1
4	G723	A	8 000	1
15	G728	A	8 000	1
18	G729	A	8 000	1
31	H261	V	90 000	
34	H263	V	90 000	
3	GSM	A	8 000	1
96-127	Dinámica	?		
NOTA – Los tipos de cabida útil 1-7, 10-14, 16-30 y 30-95 están reservados. Para más información véase el apéndice II.				

B.7 Asignación de puertos

Como se especifica en la definición de protocolo RTP, los datos RTP han de ser transportados sobre un número de puerto UDP par y los paquetes RTCP correspondientes han de transportarse en el número de puerto inmediatamente superior (impar).

Las aplicaciones que operan bajo este perfil pueden utilizar cualquier de dichos pares de puertos UDP. Por ejemplo, el par de puertos puede ser asignado aleatoriamente por un programa de gestión de sesión. No puede necesitarse un único par de números de puertos fijos porque múltiples aplicaciones que utilizan este perfil es probable que se pasen por el mismo computador principal, y hay algunos sistemas operativos que no permiten que múltiples procesos utilicen el mismo puerto UDP con diferentes direcciones multidifusión.

Sin embargo, los números de puertos 5004 y 5005 se han registrado para uso con este perfil en aquellas aplicaciones que deciden utilizarlos como el par por defecto. Las aplicaciones que operan bajo múltiples perfiles pueden utilizar este par de puertos como una indicación para seleccionar este perfil si no están sujetos a la limitación del párrafo anterior. Las aplicaciones no necesitan tener un valor por defecto y pueden requerir que el par de puertos se especifique explícitamente. Los números de puertos concretos se eligieron para que queden en la gama superior a 5000 para acomodar la práctica de asignación de números de puerto dentro del sistema operativo Unix, en el que los números de puerto por debajo de 1024 sólo pueden ser utilizados por procesos privilegiados y los números de puerto entre 1024 y 5000 son automáticamente asignados por el sistema operativo.

ANEXO C

Formato de cabida útil RTP para trenes de vídeo H.261

Véase la introducción al anexo A; todas las advertencias allí indicadas se aplican igualmente a este anexo. En el apéndice III puede verse una referencia informativa al documento IETF completo; sin embargo, este anexo contiene toda la información necesaria para la implementación de la Recomendación H.323.

C.1 Introducción

La Recomendación H.261 [C-2] especifica las codificaciones utilizadas por los códecs de videoconferencia conformes del UIT-T. Aunque estas codificaciones se especificaron originalmente para circuitos RDSI a velocidades de datos fijas, los experimentos han demostrado que pueden también ser utilizadas por redes con conmutación de paquetes tales como Internet.

El objeto de este anexo es especificar el formato de cabida útil RTP para encapsular trenes de vídeo H.261 en RTP (véase el anexo A).

C.2 Estructura del tren de paquetes

C.2.1 Sinopsis de la Recomendación H.261

La codificación H.261 se organiza como una jerarquía de agrupamientos. El tren de vídeo se compone de una secuencia de imágenes, o de tramas, que a su vez se organizan como un conjunto de grupos de bloques (GOB, *groups of blocks*). Adviértase que las "imágenes" H.261 se denominan "tramas" en esta Recomendación. Cada GOB contiene un conjunto de 3 líneas de 11 macrobloques (MB, *macro blocks*). Cada MB transporta información en un grupo de 16×16 píxels: se especifica información de luminancia para 4 bloques de 8×8 píxels, mientras que la información de crominancia viene dada por dos componentes de diferencia de color "rojo" y "azul" a una resolución de sólo 8×8 píxels. Estas componentes y los códigos que representan sus valores muestreados son los definidos en la Recomendación UIT-R BT.601 [C-3].

Este agrupamiento se utiliza para especificar información a cada nivel de la jerarquía:

– Al nivel de trama, se especifica información tal como el retardo con respecto a la anterior trama, el formato de imagen, y diversos indicadores.

– Al nivel GOB, se especifica el número GOB y el cuantificador por defecto que se utilizará para los MB.

– Al nivel MB, se especifica qué bloques están presentes y cuáles no cambiaron, y opcionalmente un cuantificador y vectores de movimiento.

Los bloques que han cambiado se codifican computando la transformada discreta de coseno (DCT, *discrete cosine transform*) de sus coeficientes, que entonces se cuantifican y se someten a codificación Huffman (códigos de longitud variable).

La codificación Huffman H.261 incluye un patrón "comienzo de GOB", compuesto de 15 ceros seguidos por un solo 1, que no puede ser imitado por cualesquiera otras palabras de código. Este patrón se incluye al comienzo de cada encabezamiento GOB (y también al comienzo de cada encabezamiento de trama) para marcar la separación entre dos GOB, y se utiliza de hecho como un indicador de que el actual GOB ha terminado. La codificación también incluye un patrón de relleno, compuesto de siete ceros seguidos por cuatro unos; ese patrón de relleno sólo puede introducirse entre la codificación de MB, o inmediatamente antes del separador de GOB.

C.2.2 Consideraciones para la paquetización

Los códecs H.261 diseñados para la operación por circuitos RDSI producen un tren de bits compuesto de varios niveles de codificación especificados por la Recomendación H.261 y Recomendaciones acompañantes. Los bits resultantes de la codificación Huffman se codifican en tramas de 512 bits, que contienen 2 bits de sincronización, 492 bits de datos y 18 bits de código de corrección de errores. Las tramas de 512 bits se entrelazan entonces con un tren de audio y se transmiten por circuitos a $p \times 64$ kbit/s de acuerdo con la Recomendación H.221 [C-1].

Cuando se transmita por Internet, consideraremos directamente la salida de la codificación Huffman. Todos los bits producidos por la etapa de codificación Huffman se incluirán en el paquete. No transportaremos las tramas de 512 bits, ya que la protección contra los errores de bit puede obtenerse por otros medios. Análogamente, no intentaremos multiplexar las señales de audio y de vídeo en los mismos paquetes, ya que UDP y RTP proporcionarán un modo mucho más eficaz de obtener multiplexación.

Transmitiendo directamente el resultado de la codificación Huffman sobre un tren no fiable de datagramas UDP se obtendrían, sin embargo, características mediocres de resistencia a los errores. El resultado de esta estructura jerárquica del tren de bits H.261 es que se necesita recibir la información presente en el encabezamiento de trama para decodificar los GOB, así como la información presente en el encabezamiento de GOB para decodificar los MB. Sin precauciones, esto significaría que han de recibirse todos los paquetes que transportan una imagen a fin de decodificar adecuadamente sus componentes.

Si cada imagen pudiera transportarse en un único paquete, este requisito no crearía ningún problema. Sin embargo, una imagen de vídeo o incluso un GOB puede a veces ser demasiado grande para caber en un solo paquete. Por tanto, el MB se toma como la unidad de fragmentación. Los paquetes deben comenzar y terminar en una frontera de MB, es decir, un MB no puede repartirse entre múltiples paquetes. Pueden transportarse múltiples MB en un solo paquete cuando quepan dentro del máximo tamaño de paquete permitido. Esta práctica se recomienda para reducir la velocidad de emisión de paquetes y la tara de paquetes.

Para permitir que cada paquete se procese independientemente para que haya una resincronización eficaz en presencia de pérdidas de paquetes, se transporta alguna información de estado procedente del encabezamiento de trama y del encabezamiento de GOB con cada paquete para permitir la codificación de los MB de ese paquete. Esta información de estado incluye el número de GOB en efecto al comienzo del paquete, el predictor de dirección de macrobloque (es decir, el último MBA codificado en el paquete anterior), el valor de cuantificador en efecto antes del comienzo de este paquete (GQUANT, MQUANT o cero en caso de un comienzo de GOB) y los datos de vectores de movimiento (MVD, *motion vector data*) de referencia para calcular los verdaderos MVD contenidos dentro de este paquete. El tren de bits no puede fragmentarse entre un encabezamiento de GOB y el MB 1 de ese GOB.

Además, dado que el MB comprimido no puede llenar un número entero de octetos, el encabezamiento de datos contiene dos enteros de tres bits, SBIT y EBIT, para indicar el número de bits no utilizados en los octetos primero y último de los datos H.261, respectivamente.

C.3 Especificación del esquema de paquetización

C.3.1 Utilización del RTP

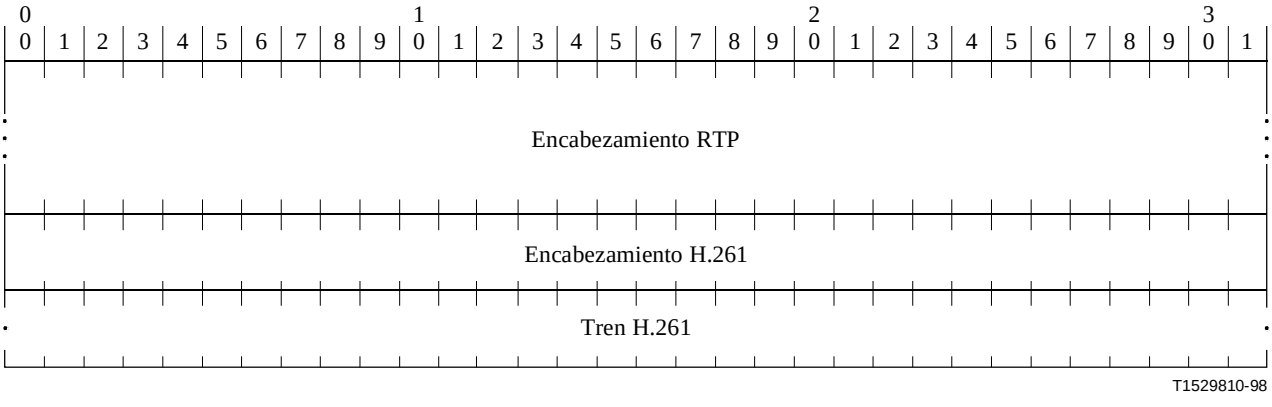
La información H.261 es transmitida como datos de cabida útil dentro del protocolo RTP. Se especifican los siguientes campos del encabezamiento RTP:

– El tipo de cabida útil debe especificar el formato de cabida útil H.261 (véase el anexo B).

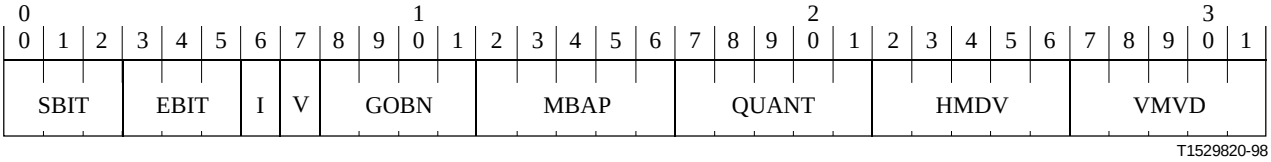
– La indicación de tiempo RTP codifica en el instante de muestreo de la primera imagen de vídeo contenida en el paquete de datos RTP. La indicación de tiempo RTP será la misma en paquetes sucesivos si una imagen de vídeo de la misma imagen de vídeo ocupa más de un paquete. Para trenes de vídeo H.261, la indicación de tiempo RTP se basa en un reloj de 90 kHz. Esta frecuencia de reloj es un múltiplo de la velocidad de trama H.261 (es decir, 30 000/1001 o aproximadamente 29,97 Hz). De este modo, para cada tiempo de trama, el reloj se incrementa simplemente en el múltiplo y se elimina así la inexactitud al calcular la indicación de tiempo. Además, el valor inicial de la indicación de tiempo es aleatorio (impredecible) para dificultar los ataques de texto claro conocido sobre la criptación, véase RTP (anexo A). Adviértase que si se codifican múltiples tramas en un paquete (por ejemplo, cuando hay muy pocos cambios entre dos imágenes), es necesario calcular los tiempos de visualización para las tramas después del primero que utiliza la información de temporización en el encabezamiento de trama H.261. Esto es necesario porque la indicación de tiempo RTP sólo da el tiempo de visualización de la primera trama del paquete.

– El bit marcador del encabezamiento RTP se pone a uno en el último paquete de una trama de vídeo, y en otro caso, debe ser cero. Así, no es necesario esperar un paquete siguiente (que contenga el código de arranque que termina la trama actual) para detectar que debe visualizarse una nueva trama.

Los datos H.261 seguirán el encabezamiento RTP, como en:



El encabezamiento H.261 se define como sigue:



Los campos del encabezamiento H.261 tienen los siguientes significados:

Posición de bit inicial (SBIT, start bit position): 3 bits – Número de bits que debe ser ignorado en el primer octeto de datos.

Posición de bit final (EBIT, end bit position): 3 bits – Número de bits que debe ser ignorado en el último octeto de datos.

Datos codificados INTRAtrama (I): 1 bit – Se pone a 1 si este tren contiene sólo bloques codificados INTRAtrama. Se pone a 0 si este tren puede o no contener bloques codificados INTRAtrama. El sentido de este bit puede no cambiar en el transcurso de la sesión.

Bandera de vectores de movimiento (V, motion vector flag): 1 bit – Se pone a 0 si no se utilizan en este tren vectores de movimiento. Se pone a 1 si los vectores de movimiento pueden o no utilizarse en este tren. El sentido de este bit puede no cambiar durante el curso de la sesión.

Número de GOB (GOBN, GOB number): 4 bits – Codifica el número de GOB en efecto al comienzo del paquete. Se pone a 0 si el paquete comienza con un encabezador de GOB.

Predictor de dirección de macrobloque (MBAP, macroblock address predictor): 5 bits – Codifica el predictor de dirección de macrobloque (es decir, el último MBA codificado en el paquete anterior). Este predictor varía de 0 a 32 (para predecir los MBA 1-33), pero debido a que el tren de bits no puede ser fragmentado entre un encabezamiento de GOB y MB 1, el predictor al comienzo del paquete nunca puede ser 0. Por tanto, la gama es 1-32, que es sesgada en −1 para caber en 5 bits. Por ejemplo, si MBAP es 0, el valor del predictor de MBA es 1. Se fija a 0 si el paquete comienza por un encabezamiento GOB.

Cuantificador (QUANT, quantizer): 5 bits – Valor del cuantificador (MQUANT o GQUANT) en efecto antes del comienzo de este paquete. Se pone a 0 si el paquete comienza con un encabezamiento de GOB.

Datos de vectores de movimiento horizontal (HMVD, horizontal motion vector data): 5 bits – Datos de vectores de movimiento (MVD) horizontal de referencia. Se pone a 0 si la bandera V es 0 o si el paquete empieza por un encabezamiento de GOB. Los valores HMVD son número de complemento de 2 de 5 bits que representan directamente los valores [−16, +15], donde −16 no se utiliza.

Datos de vectores de movimiento vertical (VMVD, vertical motion vector data): 5 bits – Datos de vectores de movimiento (MVD) vertical de referencia. Se pone a 0 si la bandera V es 0 o si el paquete empieza por un encabezamiento de GOB. Los valores VMVD son número de complemento de 2 de 5 bits que representan directamente los valores [−16, +15], donde −16 no se utiliza.

Adviértase que las banderas I y V son banderas de indicaciones, es decir, pueden deducirse del tren de bits. Se incluyen para permitir que los decodificadores hagan optimizaciones que no serían posibles si no se proporcionasen estas indicaciones antes de que se decodifique el tren de bits. Por tanto, estos bits no pueden cambiar mientras dure el tren. Una implementación conforme puede siempre poner V = 1 e I = 0.

Los datos de los vectores de movimiento horizontal y vertical se deben poner a cero cuando MTYPE del último MB codificado en el paquete anterior no tiene movimiento compensado.

C.3.2 Recomendaciones para la operación con códecs de soporte físico

Los paquetizadores de códecs de soporte físico pueden determinar cómodamente las fronteras de GOB utilizando el patrón de comienzo de GOB incluido en los datos H.261. (Adviértase que los codificadores de soporte lógico ya conocen las fronteras.) La implementación de paquetización más barata consiste en paquetizar al nivel de GOB todos los GOB que caben en un paquete. Cuando un GOB es demasiado grande, el paquetizador lo tiene que analizar sintácticamente para efectuar la fragmentación de MB. (Adviértase que sólo debe analizarse la codificación Huffman y que no es necesario descomprimir completamente el tren, por lo que esto exige relación entre poco procesamiento; en el apéndice III pueden verse ejemplos de implementaciones.) Se recomienda que la fragmentación a nivel de MB se utilice cuando sea viable a fin de obtener una paquetización más eficaz. Utilizar este esquema de fragmentación reduce la velocidad de paquetes de salida, y por tanto reduce la tara.

En el receptor, el tren de datos puede despaquetizarse y ser dirigido a una entrada de códec de soporte físico. Si el decodificador de soporte físico opera a una velocidad binaria fija, la sincronización puede mantenerse insertando el patrón de relleno entre MB (es decir, entre paquetes) cuando la velocidad de llegada de paquetes es menor que la velocidad binaria.

C.3.3 Aspectos de pérdida de paquetes

En Internet, la mayoría de las pérdidas de paquetes se deben a congestión de la red más que a errores de transmisión. Utilizando UDP, no hay disponible ningún mecanismo en el emisor para saber si un paquete se ha recibido correctamente. Corresponde a la aplicación, es decir, al codificador y al decodificador, tratar la pérdida de paquetes. Cada paquete RTP incluye un campo de número secuencial que puede utilizarse para detectar la pérdida de paquetes.

La Recomendación H.261 utiliza redundancia temporal de vídeo para realizar la compresión. Esta codificación diferencial (o codificación INTERtrama) es sensible a la pérdida de paquetes. Tras una pérdida de paquetes, partes de la imagen pueden permanecer deterioradas hasta que todos los MB correspondientes hayan sido codificados en modo INTRAtrama (es decir, codificados independientemente de pasadas tramas). Hay varias formas de aliviar la pérdida de paquetes:

1) Una es utilizar solamente codificación INTRAtrama y renovación condicional a nivel de MB. Es decir, sólo se transmiten los MB que cambian (más allá de algún umbral).

2) Otra forma es ajustar la velocidad de renovación de codificación INTRAtrama de acuerdo con la pérdida de paquetes observada por los receptores. La Recomendación H.261 especifica que un MB es codificado INTRAtrama al menos cada 132 veces que es transmitido. Sin embargo, la velocidad de renovación INTRAtrama puede elevarse a fin de acelerar la recuperación cuando la velocidad de pérdida medida es significativa.

3) La forma más rápida de reparar una imagen deteriorada es pedir la renovación de imagen con codificación INTRAtrama después de detectarse una pérdida de paquetes. Un modo de realizarlo es que el decodificador envíe al codificador una lista de paquetes perdidos. El codificador puede decidir codificar cada MB de cada GOB en la trama de vídeo siguiente en modo INTRAtrama (es decir, con codificación INTRAtrama completa), o si el codificador puede deducir de los números secuenciales de los paquetes cuyos MB fueron afectados por la pérdida, puede ahorrarse anchura de banda enviando sólo aquellos MB en modo INTRAtrama. Este modo es particularmente eficaz en conexión punto a punto o cuando el número de codificadores es bajo. En la subcláusula siguiente se especifica cómo puede implementarse la función de renovación.

C.3.4 Utilización de paquetes de control específicos H.261 opcionales

Esta especificación define dos paquetes de control RTCP específicos H.261, "Petición INTRAtrama completa" y "Acuse de recibo negativo", descritos en la subcláusula siguiente. Su finalidad es acelerar la renovación del vídeo en las situaciones en las que su uso es viable. El soporte de estos

paquetes de control específicos por el emisor H.261 es opcional; en particular, los primeros experimentos han revelado que la utilización de esta característica podría tener efectos muy negativos cuando el número de puestos es muy grande. Así, estos paquetes de control deben utilizarse con precaución.

Los paquetes de control específicos H.261 difieren de los paquetes RTCP normales en que no son transmitidos a la dirección de transporte de destino RTCP normal para la sesión RTP (que es a menudo una dirección multidifusión). En vez de ello, estos paquetes de control se envían directamente por unidifusión del decodificador al codificador. El puerto de destino de estos paquetes de control es el mismo puerto que el codificador utiliza como puerto fuente para transmitir paquetes RTP (de datos). Por tanto, estos paquetes pueden considerarse paquetes de control "inversos".

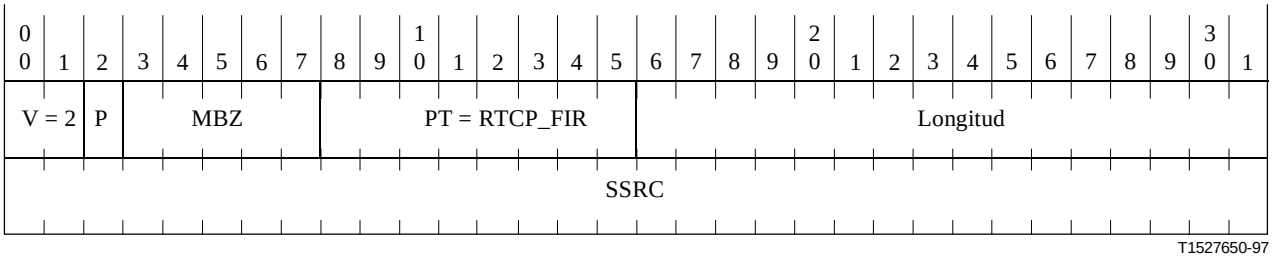
En consecuencia, estos paquetes de control sólo pueden utilizarse cuando no intervienen mezcladores ni traductores RTP en el trayecto del codificador al decodificador. Si intervienen dichos sistemas intermedios, la dirección del codificador dejaría de estar presente, ya que la dirección de fuente a nivel red dejaría de estar presente en los paquetes recibidos por el decodificador, y de hecho, podría serle imposible al decodificador enviar paquetes directamente al codificador.

Algunos protocolos multidifusión fiables utilizan paquetes de control NACK transmitidos sobre el canal de distribución multidifusión normal, pero suelen utilizar retardos aleatorios para evitar un problema de implosión NACK. El objeto de tales protocolos es proporcionar entrega fiable de paquetes multidifusión a costa de un retardo, lo que es apropiado en aplicaciones tales como la pizarra compartida.

En cambio, la transmisión vídeo interactiva es más sensible al retardo y no requiere plena fiabilidad. En las aplicaciones de vídeo, es más eficaz enviar paquetes de control NACK tan pronto como resulte posible, es decir, tan pronto como se detecte una pérdida, sin añadir retardos aleatorios. En este caso, los paquetes de control NACK generarían tráfico inútil entre receptores, ya que sólo el codificador los utilizaría. Pero este método es sólo efectivo cuando el número de receptores es pequeño, es decir, si los paquetes de control específicos H.261 se utilizan solamente en conexiones punto a punto o en conexiones punto a multipunto cuando hay menos de 10 participantes en la conferencia.

C.3.5 Definición de paquetes de control

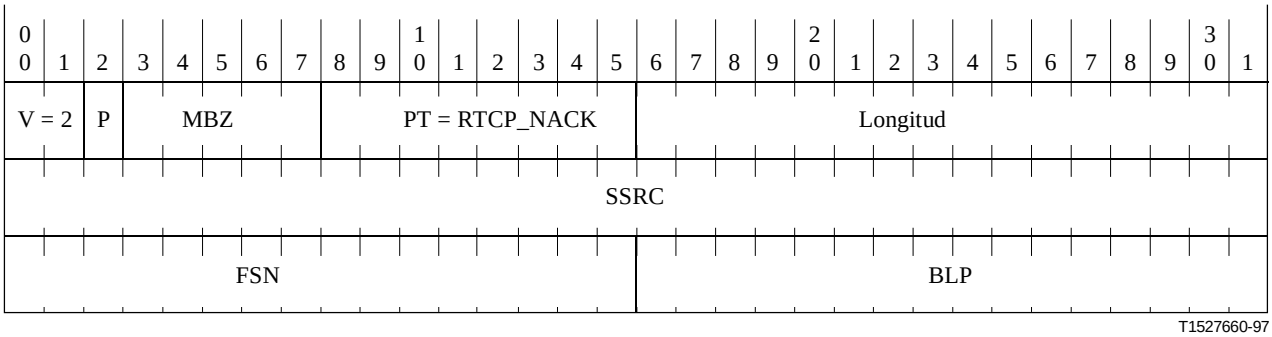
C.3.5.1 Paquete de petición INTRAtrama completa (FIR, *full INTRA-frame request*)



Este paquete indica que un receptor requiere una imagen codificada completa a fin de iniciar la decodificación con una imagen entera o renovar su imagen y acelerar la recuperación después de una ráfaga de paquetes perdidos. El receptor pide a la fuente que fuerce la siguiente imagen en modo codificación "INTRAtrama" completo, es decir, sin utilizar codificación diferencial. Los diversos campos se definen en la especificación RTP (anexo A). SSRC es el identificador de fuente de sincronización para el emisor de este paquete. El valor del tipo de paquete (PT, *packet type*) es la constante RTCP_FIR (192).

C.3.5.2 Paquete de acuse de recibo negativo (NACK, *negative ACKnowledgement*)

El formato del paquete NACK es el siguiente:



Los diversos campos T, P, PT, longitud y SSRC se definen en la especificación RTP (véase el anexo A). El valor del identificador de tipo de paquete (PT) es el RTCP_NACK (193) constante. SSRC es el identificador de fuente de sincronización para el emisor de este paquete.

Los dos campos restantes tienen los significados siguientes:

Primer número secuencial (FSN, first sequence number): 16 bits – Identifica el primer número secuencial perdido.

Plantilla de bits de los siguientes paquetes perdidos (BLP): 16 bits – El bit A se pone a 1 si el paquete correspondiente se ha perdido, y se pone a 0 en otro caso. BLP se pone a 0 sólo si no se ha perdido ningún paquete distinto de aquel del que se hace acuse negativo NACKed (utilizando el campo FSN). BLP se pone a 0x00001 si se han perdido el paquete correspondiente al FSN y el paquete siguiente, etc.

C.4 Bibliografía

[C-1] Recomendación UIT-T H.221 (1997), *Estructura de trama para un canal de 64 a 1920 kbit/s en teleservicios audiovisuales*.

[C-2] Recomendación UIT-T H.261 (1993), *Códec vídeo para servicios audiovisuales a p ×64 kbit/s*.

[C-3] Recomendación UIT-R BT.601 (1997), *Métodos digitales de transmisión de información de televisión*.

ANEXO D

Formato de la cabida útil del RTP para trenes de vídeo H.261A

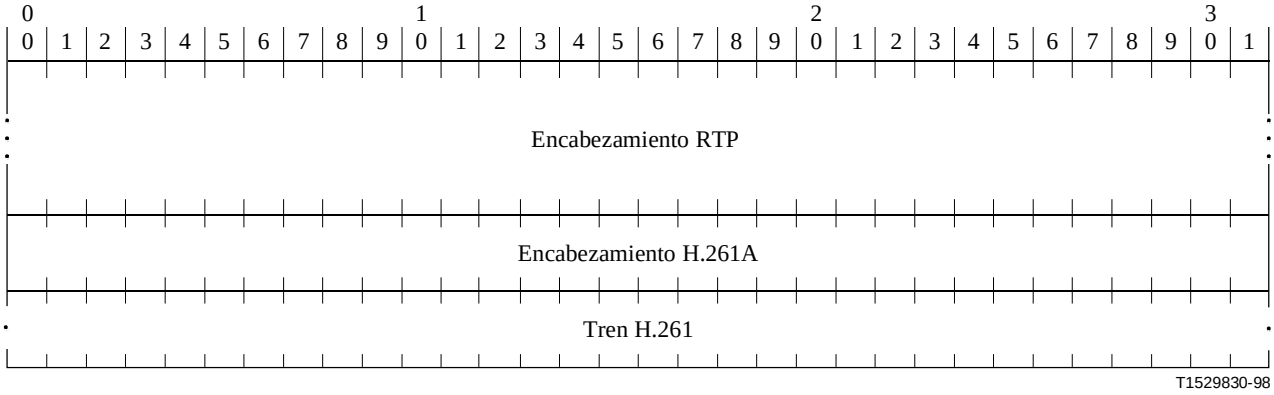
D.1 Introducción

Para facilitar la interconexión de trenes vídeo H.323 con la RCC a través de las pasarelas, la Recomendación H.323 define una forma modificada de la cabida útil de vídeo H.261 RTP. Esto facilita la gestión de la memoria tampón y el interfuncionamiento con códecs RCC distantes. El soporte del tipo de cabida útil H.261A se señaliza utilizando conjuntos de capacidades H.245 y en el mensaje **apertura de canal lógico** utilizando tipos de cabida útil dinámica RTP.

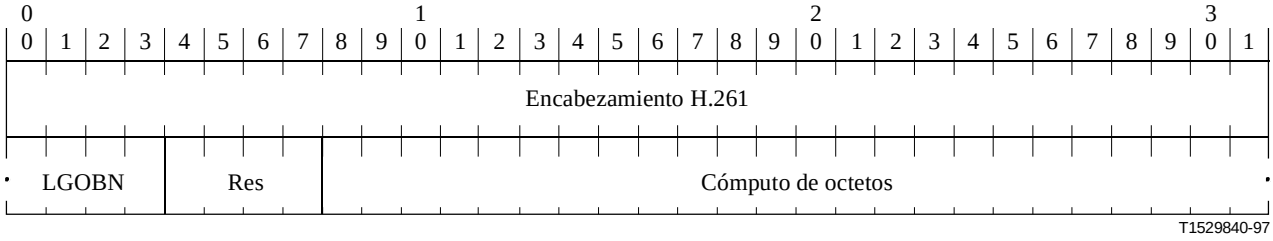
D.2 Paquetización RTP H.261A

Esta versión es una ampliación de la versión descrita en el anexo C, salvo que se añade una palabra adicional de 32 bits al encabezamiento H.261. Los procedimientos que se describen en el anexo C se aplican también a este anexo.

Los datos H.261A seguirán al encabezamiento RTP como se indica a continuación:



El encabezamiento H.261A se define como:



Los campos de encabezamiento H.261A tienen los siguientes significados:

Encabezamiento H.261: 32 bits – Como se describe en el anexo C.

Último número GOB (LGOBN, last GOB number): 4 bits – El número GOB del último GOB en el paquete RTP (el número GOB máximo es 12 para H.261).

Reservado (RES): Reservado.

Cómputo de octetos: 24 bits – Indica el número acumulado de octetos que han sido enviados en la parte de tren H.261 de los paquetes RTP. Si el último octeto de un paquete está rellenado sólo parcialmente (como es indicado por EBIT), entonces no se cuenta en el cómputo acumulado de octetos. Este cómputo de octetos módulo 2^{24} comienza en un valor aleatorio y no se reinicia nunca.

Se puede utilizar ambos campos adicionales cuando se pierden paquetes o se entregan fuera de orden. El cómputo de octetos se puede utilizar para determinar cuánto relleno se necesitará en el tren RCC y facilitar la gestión de la memoria tampón. El último número GOB simplifica la determinación de cuáles GOB se han perdido debido a pérdida de paquetes.

ANEXO E

Paquetización de vídeo

En IETF RFC 2190 se especifica un formato de cabida útil del RTP para vídeo H.263 para trenes de bits de vídeo H.263 que no contienen las características nuevas adoptadas en la versión 2 (la versión de 1998) de la Recomendación H.263 (las características que utilizan PLUSPTYPE o los anexos que siguen al anexo H/H.263). Más adelante se especificará un formato de cabida útil adicional que soporte las características mejoradas de los trenes de bits de la versión 2 de la Recomendación H.263. Un formato de paquetización de legado ampliamente utilizado en la industria (no especificado en IETF RFC 2190) sólo se puede emplear si la entidad par ha indicado que soporta ese formato en el intercambio de capacidades.

En la subcláusula B.5 se describe el procedimiento a seguir para señalizar trenes de vídeo H.263.

ANEXO F

Paquetización de audio

En este anexo se describen los detalles de la paquetización RTP para códecs de audio normalizados por la UIT-T.

F.1 G.723.1

La Recomendación G.723.1 especifica una representación codificada que puede utilizarse para la compresión del componente señal vocal de los servicios multimedios a una velocidad binaria muy baja. Una trama G.723.1 puede tener uno de los tres siguientes tamaños: 24 bytes (trama de 6,3 kbit/s), 20 bytes (trama de 5,3 kbit/s) ó 4 bytes. Las tramas de 4 bytes se denominan tramas SID (descriptor de inserción de silencio) y se utilizan para especificar parámetros de nivelación de ruido. No hay ninguna restricción con respecto a la forma en que se combinan entre sí las tramas de 4, 20 y 24 bytes. Los dos bits menos significativos del primer octeto de la trama determinan el tamaño de la trama y el tipo de códec (para más información sobre el orden de los bits, véanse los cuadros 5 y 6/G.723.1). Es posible pasar de una a otra de las dos velocidades en cualquier frontera de trama de 30 ms. Ambas velocidades (5,3 kbit/s y 6,4 kbit/s) son parte obligatoria del codificador y decodificador. Este codificador se optimizó para representar la señal vocal con una calidad próxima a la de los enlaces de larga distancia en las velocidades mencionadas utilizando un grado de complejidad limitado.

Todos los bits del tren de bits codificado se transmiten siempre desde el bit menos significativo al bit más significativo.

NOTA – Esto se refiere al orden de los bits presentados a la capa de transporte y no al orden de los bits en el hilo conductor.

La paquetización G.723.1 es conforme al anexo B excepto en lo que respecta al intervalo de paquetización (30 ms frente a los 20 ms por defecto):

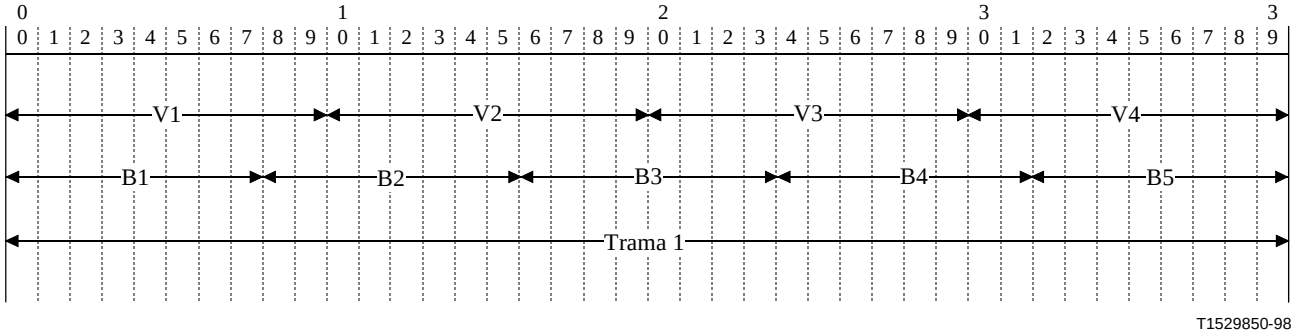
1) El primer paquete de un arranque de palabra (primer paquete tras un periodo de silencio) se distingue fijando el bit marcador en el encabezamiento de datos RTP.
2) La frecuencia de muestreo (frecuencia de reloj RTP) es 8000 Hz.
3) El intervalo de paquetización tendrá una duración de 30 ms (una trama) a diferencia de la paquetización por defecto de 20 ms.
4) Los códecs deben poder codificar y decodificar varias tramas consecutivas dentro de un solo paquete.

5) Un receptor debe aceptar paquetes que representen entre 0 y 180 ms de datos de audio a diferencia de los 0 y 200 ms por defecto.

F.2 G.728

1) *Paquetización de trama*

Una trama G.728 (4 vectores: V1-V4, 10 bits cada uno, V1 es el más antiguo, el que se ha de reproducir primero) se organiza en 5 bytes (B1-B5). Respecto a la figura que aparece a continuación, el principio para el orden de bits es el de "mantenimiento de la importancia de los bits". Los bits de los vectores más antiguos son más significativos que los bits de los vectores más recientes. El bit más significativo (MSB, *most significant bit*) de la trama pasa a ser el MSB de B1 y el bit menos significativo (LSB, *least significant bit*) de la trama pasa a ser el LSB de B5. Para mayor claridad: los bits más significativos de cada vector se colocan en los bits más significativos de B1-B5 (los bits más significativos del B de número más bajo).



Por ejemplo:

B1 contiene 8 bits más significativos de V1, el MSB de V1 es el MSB de B1.

B2 contiene 2 bits menos significativos de V2, el más significativo de los dos en su MSB, y 6 bits más significativos de V2, el más significativo de ellos es también más significativo de B2.

B1 será el primero del paquete (el byte más significativo en RTP) y B5 el último.

2) *Paquetización de multitrama*

El envío de una sola trama en un paquete RTP puede dar lugar a una tara considerable en la red. Por ello se permite enviar un paquete multitrama de la siguiente manera:

Un paquete RTP G.728 deberá contener un número completo de tramas.

Las tramas más antiguas (las que se han de reproducir primero) deberán colocarse las primeras en el paquete RTP.

La indicación de tiempo reflejará el tiempo de captura de la primera muestra, en el primer vector (V1) de la primera trama (la información más antigua en el paquete).

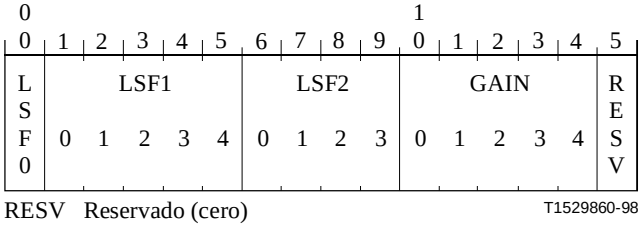
3) El bit marcador mantendrá el mismo significado que se le asigna en la Recomendación H.225.0.

F.3 G.729

La Recomendación G.729 especifica una representación codificada que puede utilizarse para la compresión del componente señal vocal de los servicios multimedios a una velocidad binaria de 8 kbit/s. Este codificador se optimizó para representar la señal vocal con una calidad similar a la de los enlaces de larga distancia o alámbricos en 8 kbit/s. Tiene robustez inherente contra errores aleatorios en los bits así como contra tramas borradas de manera aleatoria y por ráfagas. Representa

la señal vocal con alta calidad cuando funciona en un entorno ruidoso. En el anexo A/G.729 se especifica una versión de complejidad reducida del algoritmo G.729. Los algoritmos de codificación vocal del cuerpo principal de la Recomendación G.729 y del anexo A/G.729 son plenamente interoperables entre sí, por lo que no es necesario distinguir más entre ellos.

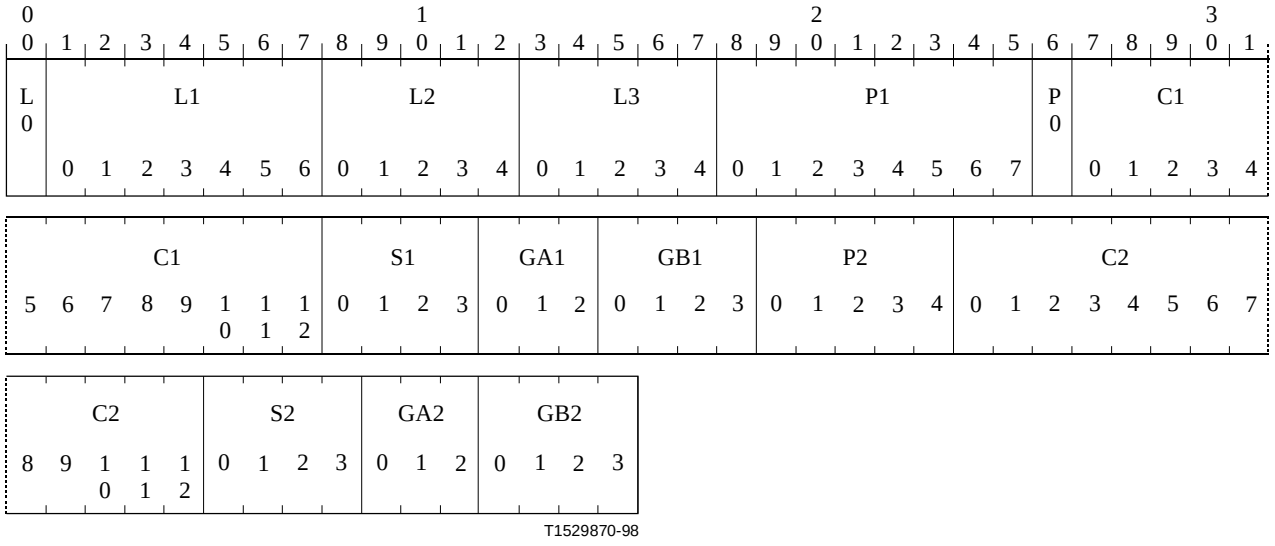
Para aplicaciones simultáneas de voz y datos digitales se recomienda un algoritmo detector de actividad vocal (VAD, *voice activity detector*) y generador de nivelación de ruido (CNG, *comfort noise generator*) indicado en el anexo B/G.729, que puede utilizarse junto con la Recomendación G.729 o el anexo A/G.729. Una trama G.729 o trama del anexo A/G.729 contiene 10 octetos, en tanto que la trama de nivelación de ruido del anexo B/G.729 ocupa dos octetos:



RESV Reservado (cero)

T1529860-98

Los parámetros transmitidos de una trama de 10 ms G.729 y G.729 anexo A, formada por 80 bits, están definidos en el cuadro 8/G.729. A continuación figura la correspondencia de estos parámetros. Los bits están numerados en el mismo orden de Internet, es decir, el bit más significativo es el bit 0.



T1529870-98

Un paquete RTP puede constar de cero o más tramas G.729 o tramas G.729 anexo A, seguidas de cero o una cabida útil del anexo B/G.729. La presencia de una trama de nivelación de ruido puede deducirse a partir de la longitud de la cabida útil RTP.

1) El primer paquete de un arranque de palabra (primer paquete tras un periodo de silencio) se distingue fijando el bit marcador en el encabezamiento RTP.

2) La frecuencia de muestreo (frecuencia de reloj RTP) es 8000 Hz.

3) El intervalo de paquetización por defecto deberá tener una duración de 20 ms. Aunque el valor de 20 ms se recomienda encarecidamente en algunas situaciones quizá convenga enviar paquetes de 10 ms. Considérese, por ejemplo, el tránsito de voz a ausencia de voz en los primeros 10 ms del paquete. Si fuese obligatorio un intervalo de paquetización de 20 ms, el transmisor tendría que esperar hasta que la señal vocal estuviera activa de nuevo.

4) Los códecs deben poder codificar y decodificar varias tramas consecutivas dentro de un solo paquete.

5) Un receptor debe aceptar paquetes que representen entre 0 y 200 ms de datos de audio.

F.4 Supresión de silencio

La Recomendación H.225.0 indica que los codificadores pueden enviar tramas de silencio antes de interrumpir la transmisión durante un periodo de silencio. Dado que no todos los codificadores de audio tienen señalización dentro de banda para silencio, se debe definir un mecanismo general a nivel del RTP. Por ejemplo, se podría enviar un paquete RTP vacío. Este asunto queda en estudio.

F.5 Códecs GSM

Los códecs de señales vocales GSM incluyen: GSM de velocidad total (FR, *GSM full rate*) [F-1], GSM de velocidad media (HR, *GSM half rate*) [F-3] y GSM de velocidad total mejorado (EFR, *GSM enhanced full rate*) [F-2]. Cada códec produce tres tipos de trama de tráfico de señal vocal diferentes, a saber:

- Tramas de señales vocales – Contiene datos de señales vocales verdaderos.
- Tramas inactivas – Indica que no hay actividad vocal; todos los bits de datos están puestos a uno.
- Tramas de descriptor de inserción de silencio (SID, *silence insertion descriptor*) –Indica el comienzo de un periodo de silencio, los datos describen el ruido de fondo. Las tramas SID se marcan dentro de banda con un diagrama de bits fijo.

F.5.1 Paquetización de tramas

Con los tres códecs GSM, los bits de trama de tráfico de señales vocales se empacan con el bit más significativo (MSB) de trama del RTP primero. Un paquete RTP puede contener una o más tramas de tráfico de señales vocales GSM. Todos los puntos extremos deben poder recibir e identificar una trama inactiva. Una trama de señales vocales GSM inactiva se rellena con unos binarios.

Si un punto extremo fija el parámetro `comfortNoise` en `VERDADERO`, enviará tramas SID como se indica en las especificaciones de ruido nivelador y transmisión discontinua (DTX, *discontinuous transmission*) de un código GSM particular. Durante un periodo de silencio, se envía periódicamente una nueva trama SID con información de ruido (posiblemente) actualizada, es decir, cada 24 tramas. Después de un periodo de silencio, el bit marcador se pone a 1 en el encabezamiento RTP.

Códec de velocidad plena

El códec de velocidad plena GSM envía una trama de 260 bits (32,5 octetos) cada 20 ms. Esta información será empacada en la trama RTP con un prefijo de cuatro bits (0xD o 1101 binarios), denominado signatura. Por tanto, la cabida útil FR del GSM dentro del transporte en tiempo real RTP comprenderá 33 octetos. La trama del descriptor de inserción de silencio (SID) viene marcada dentro de banda por una palabra de código SID almacenada en parámetros códec como se describe en la referencia [F-4]. La dimensión de la cabida útil de una trama SID es de 33 octetos. La signatura de una trama SID de velocidad plena será la misma que la trama de señales vocales de velocidad plena (0xD). Las señales vocales de velocidad plena codificadas en RTP tendrán una velocidad binaria de 13 200 bit/s, no incluida la tara de paquetización.

Códec de velocidad media

El códec de velocidad media GSM envía una trama de 112 bits (14 octetos) cada 20 ms. Esta información se empacará dentro de un encabezamiento RTP sin ningún prefijo ni signatura. La trama SID está marcada dentro de banda por una palabra de código SID almacenada en parámetros códec como se describe en la referencia [F-4]. El tamaño de la cabida útil de una trama SID es de 14 octetos. Las señales vocales codificadas del RTP tendrán una velocidad binaria de 5600 bit/s, no incluida la tara de paquetización.

Velocidad plena mejorada

El códec de velocidad plena mejorada (EFR) del GSM envía una trama de 244 bits (30,5 octetos) cada 20 ms. Esta información será empacada dentro de un encabezamiento RTP con un prefijo de 4 bits (0xC o 1100 binarios), denominado signatura. La cabida útil de la velocidad plena mejorada del GSM dentro del RTP constará de 31 octetos. La trama SID viene marcada dentro de banda por una palabra de código SID almacenada en parámetros códec como se describe en la referencia [F-4]. El tamaño de la cabida útil de una trama SID es de 31 octetos. Las señales vocales de velocidad plena mejorada codificadas en el RTP tendrán una velocidad binaria de 12 400 bit/s, no incluida la tara de paquetización.

F.5.2 Referencias informativas

[F-1] GSM 06.10 (ETS 300 961), *Full rate speech; transcoding*.

[F-2] GSM 06.60 (ETS 300 726), *Enhanced Full Rate (EFR) speech transcoding*.

[F-3] GSM 06.20 (ETS 300 969), *Half rate speech; Half rate speech transcoding*.

[F-4] ETSI, TIPHON 03 001 (TS 101 318), *Using GSM speech codecs within H.323*.

[F-5] GSM 06.31 (ETS 300 963), *Full rate speech; Discontinuous Transmission (DTX) for full rate speech traffic channels*.

[F-6] GSM 06.81 (ETS 300 729), *Discontinuous Transmission (DTX) for Enhanced Full Rate (EFR) speech traffic channels*.

[F-7] GSM 06.41 (ETS 300 972), *Half rate speech; Discontinuous Transmission (DTX) for half rate speech traffic channels*.

[F-8] GSM 06.12 (ETS 300 963), *Full rate speech; Comfort noise aspect for full rate speech traffic channels*.

[F-9] GSM 06.62 (ETS 300 728), *Comfort noise aspects for Enhanced Full rate (EFR) speech traffic channels*.

[F-10] GSM 06.22 (ETS 300 971), *Half rate speech; Comfort noise aspect for half rate speech traffic channels*.

[F-11] GSM 08.60 (ETS 300 737), *Inband control of remote transcoders and rate adaptors for Enhanced Full Rate (EFR) and full rate channels*.

ANEXO G

Comunicación entre dominios administrativos

G.1 Alcance

Se prevé que la red H.323 global consistirá en subconjuntos más pequeños de equipos organizados, por ejemplo por dominio administrativo. Debido al número posiblemente importante de equipos H.323 que existirán en las redes H.323, se necesita un protocolo eficaz para poder completar las llamadas entre dominios administrativos. El ejemplo más elemental es el de un usuario (un punto extremo) perteneciente a un dominio administrativo que se comunica con un usuario (un punto extremo) perteneciente a otro dominio administrativo. Aunque el protocolo RAS H.225.0 puede atender muchas de las necesidades de comunicación entre dominios administrativos, no es completo ni eficaz a estos efectos.

El presente anexo describe métodos que permiten la resolución de dirección, la autorización del acceso y la notificación de uso entre dominios administrativos en los sistemas H.323 para completar

llamadas entre dominios administrativos. Un determinado dominio administrativo se presenta a los otros dominios administrativos a través de un tipo de elemento lógico conocido como elemento de frontera. Un elemento de frontera puede estar coubicado con cualquier otra entidad, (por ejemplo, con un guardián de puerto). De conformidad con el anexo G, un dominio administrativo no necesita revelar detalles acerca de su organización o arquitectura. El anexo G no impone una arquitectura de sistema específica en un dominio administrativo. Además, el anexo G soporta la utilización de cualquier modelo de llamada (encaminada a través de un guardián de puerto o directamente hasta el punto extremo).

De acuerdo con el procedimiento general, los elementos de frontera intercambian información sobre las direcciones que cada dominio administrativo puede resolver. Las direcciones pueden especificarse de manera general o con especificidad creciente. Gracias a la información adicional, los elementos de un dominio administrativo pueden determinar el dominio administrativo más apropiado como destino de la llamada. Los elementos de frontera pueden controlar el acceso a sus direcciones presentadas y requerir notificaciones sobre el uso efectuado durante las llamadas realizadas a dichas direcciones.

En la figura G.1 se indican varios puntos de referencia que representan la señalización entre varios elementos en una red H.323. En esta figura, los dominios administrativos forman parte de una red de paquetes global sin bordes. Obsérvese que esta figura no es una definición explícita de una arquitectura de sistema H.323, sino que ilustra los puntos de referencia de señalización.

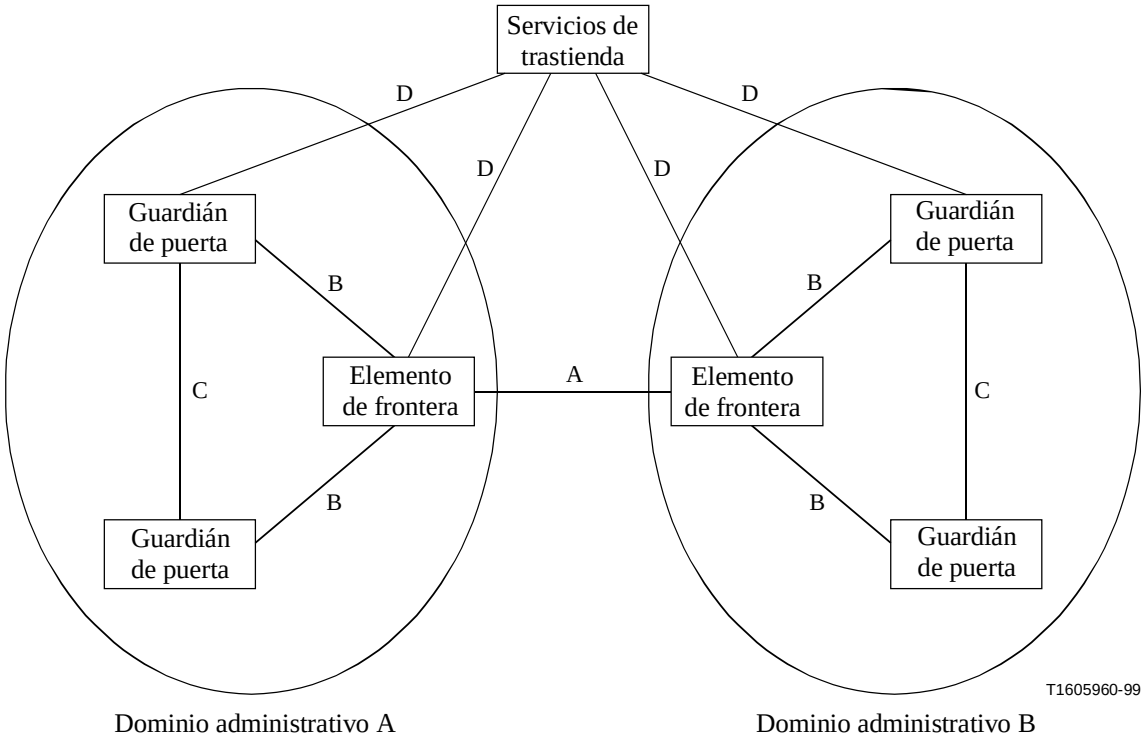


Figura G.1/H.225.0 – Puntos de referencia del sistema

En la figura se indican los siguientes puntos de referencia:

A – Entre elementos de frontera.

B – Entre elementos de frontera y guardianes de puerta.

C – Entre guardianes de puerta.

D – Entre elementos H.323 y servicios de trastienda (fuera del alcance del presente anexo).

El punto de referencia A es el tema central del anexo G. El empleo del protocolo descrito en el anexo G para la comunicación entre guardianes de puerta dentro de un dominio administrativo queda en estudio. El punto de referencia B queda en estudio dado que actualmente se supone que el elemento de frontera estará coubicado con otro elemento H.323.

En la subcláusula G.9, Ejemplos de señalización, se dan algunos ejemplos de señalización que pueden facilitar la comprensión.

G.2 Definiciones

En esta Recomendación se definen los términos siguientes.

G.2.1 dominio administrativo: Conjunto de entidades H.323 administradas por una entidad administrativa. Un dominio administrativo puede constar de uno o varios guardianes de puerta (es decir, una o varias zonas).

G.2.2 servicios de trastienda: Funciones tales como autenticación o autorización de usuario, contabilidad, facturación, tasación/tarificación, etc. Los servicios de trastienda y el protocolo para intercambiar información con dichos servicios (si es diferente del que figura en este anexo) están fuera del ámbito del presente anexo.

G.2.3 elemento de frontera: Elemento funcional que soporta el acceso público a un dominio administrativo para completar las llamadas o para cualquier otro servicio en el que interviene la comunicación multimedios con otros elementos del dominio administrativo. El elemento de frontera controla la visión externa del dominio administrativo. Un elemento de frontera se comunica con otros elementos de frontera utilizando el protocolo definido en este anexo. Además, un elemento de frontera puede, según la implementación, comunicarse con otras entidades de su dominio administrativo. Este elemento puede existir en combinación con otros elementos H.323, por ejemplo puede haber una combinación de elemento de frontera, guardián de puerta y pasarela. Un dominio administrativo puede tener cualquier número de elementos de frontera.

G.2.4 centro de resolución: Servicio (posiblemente en forma de elemento de frontera) que puede resolver todas las direcciones (es decir, un tipo de punto de agregación).

G.3 Abreviaturas

En esta Recomendación se utilizan las siguientes siglas.

AD Dominio administrativo (*administrative domain*)

BE Elemento de frontera (*border element*)

CH Centro de resolución (*clearing house*)

DST Diferencia por la hora de verano (*daylight saving time*)

EP Punto extremo (*endpoint*)

GK Guardián de puerta (*gatekeeper*)

GW Pasarela (*gateway*)

T Terminal (*terminal*)

G.4 Referencias

[1] Recomendación UIT-T H.225.0 (1998), *Protocolos de señalización de llamada y paquetización de trenes de medios para sistemas de comunicación multimedios por paquetes*.

[2] Recomendación UIT-T H.235 (1998), *Seguridad y criptado para terminales multimedios de la serie H (basados en las Recomendaciones H.323 y H.245)*.

[3] Recomendación UIT-T H.323 (1998), *Sistemas de comunicación multimedios basados en paquetes*.

[4] Recomendación UIT-T X.680 (1997) | ISO/CEI 8824-1:1998, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica*.

[5] Recomendación UIT-T X.680 (1997)/enm.1 (1999) | ISO/CEI 8824-1:1998/enm.1:1999, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica – Enmienda 1: Identificadores de objeto relativos*.

[6] Recomendación UIT-T X.691 (1997) | ISO/CEI 8825-2:1998, *Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno: Especificación de las reglas de codificación compacta*.

G.5 Modelos de sistema

El anexo G no impone una arquitectura de sistema específica entre dominios administrativos o dentro de un dominio administrativo. En las subcláusulas siguientes se ofrecen algunas arquitecturas, a título ilustrativo, más no exhaustivo.

Por lo general, se considera que un dominio administrativo consta de un número arbitrario de zonas y de un número arbitrario de elementos de frontera. Recuérdese que un elemento de frontera es un elemento funcional que puede existir junto con cualquier otro elemento H.323. En la figura G.2 se muestran algunos ejemplos de implementaciones de elementos de frontera en combinación con otros elementos.

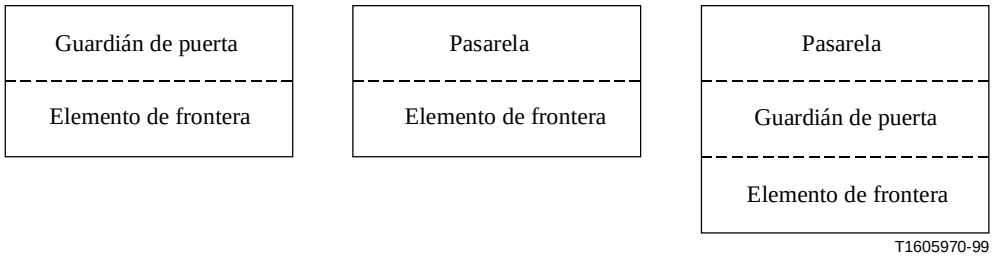


Figura G.2/H.225.0 – Ejemplos de colocación de elementos de frontera

La relación entre dominios administrativos puede ser cualquiera de toda una serie de organizaciones. En las siguientes subcláusulas se dan algunos ejemplos.

G.5.1 Jerárquica

En la figura G.3 se muestra una disposición jerárquica simple entre dominios administrativos. En este caso, para resolver una dirección, un elemento de frontera de un determinado dominio administrativo consultaría a un elemento de frontera de un dominio administrativo superior en la jerarquía.

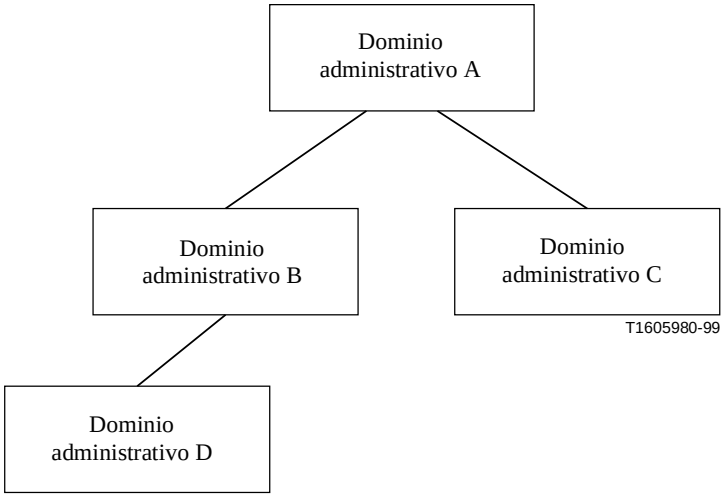


Figura G.3/H.225.0 – Ejemplo de organización jerárquica

G.5.2 Distribuida o en malla completa

En la figura G.4 se ilustra un modelo totalmente distribuido o en malla completa. En este ejemplo, un elemento de frontera de cada dominio administrativo se comunica con los elementos de frontera de los otros dominios administrativos conocidos.

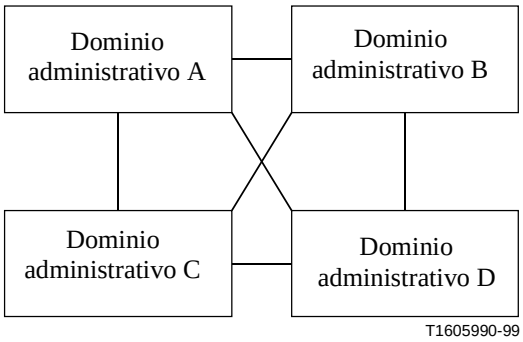


Figura G.4/H.225.0 – Ejemplo de organización distribuida

G.5.3 Centro de resolución

En la figura G.5 se muestra un ejemplo de disposición con centro de resolución. En esta disposición, cada dominio administrativo consulta al centro de resolución para resolver las direcciones.

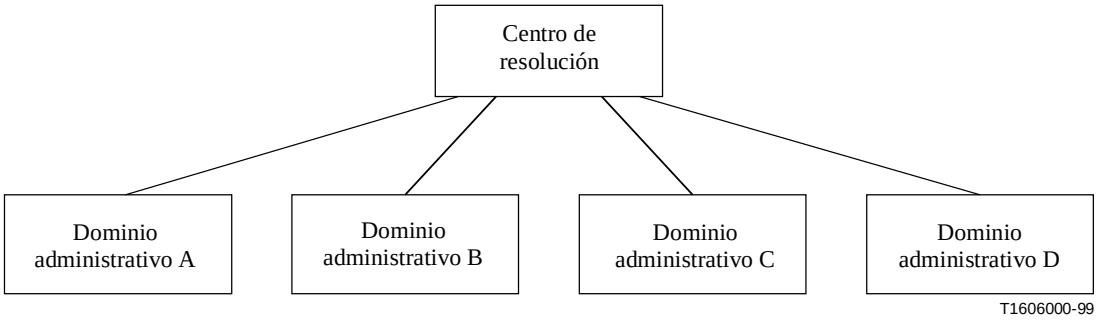


Figura G.5/H.225.0 – Ejemplo de organización con centro de resolución

G.5.4 Punto de agregación

En la figura G.6 se muestra un ejemplo de punto de agregación. En este ejemplo, el dominio administrativo B es un punto de agregación que puede resolver direcciones para sí mismo y para los dominios administrativos C y D. Por ejemplo, el dominio administrativo B puede transmitir peticiones de resolución del dominio administrativo A al dominio administrativo C, o puede indicar al dominio A que se dirija al dominio C directamente para ciertos destinos. Si el dominio administrativo B transmite una petición del dominio administrativo A al dominio administrativo C, el dominio administrativo B puede almacenar la respuesta del dominio administrativo C.

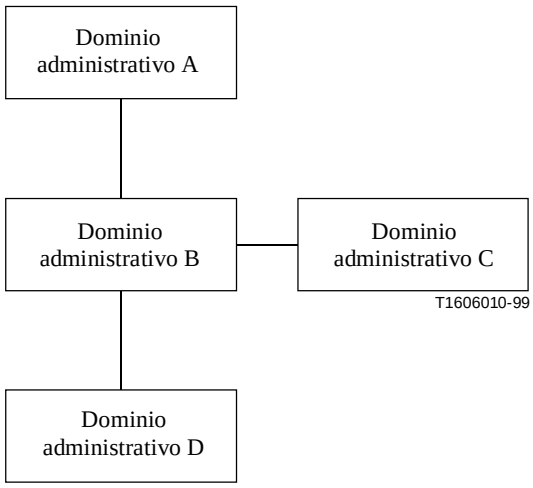


Figura G.6/H.225.0 – Ejemplo de punto de agregación

G.5.5 Dominios administrativos superpuestos

Más de un dominio administrativo puede ser capaz de resolver una determinada dirección. Por ejemplo, varios dominios administrativos podrían contener pasarelas que pueden completar una llamada a un terminal en la RTGC. La selección del dominio administrativo de destino apropiado incumbe al dominio administrativo de origen. El algoritmo empleado para seleccionar el dominio administrativo de destino incumbe a la implementación.

G.6 Convenios de direccionamiento

Para que haya interfuncionamiento entre los dominios es importante que el sistema receptor entienda los formatos de direccionamiento enviados en los mensajes H.323. Un elemento de frontera soportará las direcciones de alias (AliasAddress), tanto de tipo id de correo electrónico (email-id) como número de parte (partyNumber) (utilizando PublicNumber con PublicTypeOfNumber de internationalNumber). Obsérvese que este requisito implica que se soporta una versión ulterior de H.225.0 (1998). Al comunicar con otros elementos de frontera, deben utilizarse únicamente las direcciones de alias de tipo email-id y partyNumber en el campo dirección de destino (destinationAddress) de un mensaje LRQ o Setup (Establecimiento), a menos que haya habido un acuerdo previo entre los dominios administrativos pertinentes. Por ejemplo, si un grupo de dominios administrativos se han puesto de acuerdo sobre la interpretación de los números locales privados, estos números pueden utilizarse en los mensajes entre dichos dominios administrativos.

G.7 Funcionamiento

G.7.1 Plantillas y descriptores de dirección

Una plantilla de dirección ("plantilla", para abreviar) define un conjunto de identificadores de AliasAddress, información sobre precios para completar las llamadas a esas direcciones y el protocolo que debe utilizarse para llegar a las direcciones de ese conjunto. Un dominio administrativo utiliza plantillas para indicar las llamadas que puede resolver. Las plantillas se agrupan mediante un identificador conocido como "descriptor". Una vez agrupada una plantilla por un descriptor, todo cambio a esa plantilla implica una modificación del "grupo" del descriptor. La información que figura en la plantilla puede permitir la agregación de información de direccionamiento si el sistema de direccionamiento está dispuesto de alguna manera jerárquica o encaminable (por ejemplo, una determinada zona podría tratar el 1303538*, es decir todos los números telefónicos que empiezan por 1303538). (Obsérvese que, debido a que "*" es un carácter significativo, la plantilla incluye en la práctica una bandera booleana para indicar si la dirección es específica o no. En estos ejemplos se utiliza "*" para indicar un comodín, pero la representación efectiva en la plantilla se hace a través de la bandera booleana.)

Ejemplos de plantillas:

"Para 1 555 123 4567	enviar mensaje AccessRequest (petición de acceso) al elemento de frontera A."
"Para 1 555 987*	enviar mensaje AccessRequest al elemento de frontera B."
"Para 1 555 987 6543	enviar mensaje Setup a la pasarela X."
"Para*@example.org	enviar mensaje AccessRequest al elemento de frontera A."
"Para 1*	enviar mensaje AccessRequest al elemento de frontera B."
"Para private 31*	enviar mensaje AccessRequest al elemento de frontera C."
"Para 44 171 112*"	no existe."

Un elemento de frontera obtiene plantillas de las maneras siguientes:

- configuración estática;
- recibiendo descriptores de otros elementos de frontera en respuesta a peticiones generales;
- recibiendo respuestas a peticiones específicas.

G.7.1.1 Configuración estática

Un elemento de frontera mantendrá plantillas para todas las zonas de las que es responsable. Estas plantillas pueden realizarse explícitamente en el elemento de frontera, o pueden formarse resumiendo información obtenida de los guardianes de puerta de su dominio. El elemento de frontera puede poner esta información a disposición de otros elementos de frontera a través de respuestas a peticiones. Un dominio administrativo puede escoger el nivel de detalle que deben proporcionar sus elementos de frontera. A continuación figuran algunos ejemplos:

- Un elemento de frontera que desee ocultar la estructura interna podría proporcionar un descriptor (con una indicación de enviar un mensaje AccessRequest) que describa toda su zona y se refiera a un guardián de puerta que tratará todas las llamadas entrantes.
- Un elemento de frontera que no tenga inconveniente en revelar su estructura interna podría suministrar un conjunto de plantillas, cada una con la descripción del guardián de puerta de una zona del dominio.
- Un elemento de frontera que esté en una pasarela de seguridad o cortafuegos (o uno que utilice el modelo de encaminamiento con guardián de puerta) podría suministrar una plantilla para toda la zona con una indicación de enviar un mensaje Setup.

- Un elemento de frontera con vacíos en su dominio (debido a que se han trasladado números a otro dominio administrativo) suministra plantillas marcadas "enviar AccessRequest" que indican el elemento de frontera que debe utilizarse para dirigirse al otro dominio administrativo.
- Un elemento de frontera centro de resolución (por ejemplo, uno que tenga una copia completa de 44) podría tener una plantilla marcada "enviar AccessRequest" para cada dominio administrativo de 44.

Los elementos de frontera no necesitan mantener una copia de toda la base de datos. Si un elemento de frontera no tiene una copia de toda la base de datos, debe contener plantillas "enviar AccessRequest", configuradas estáticamente, que indiquen un elemento de frontera centro de resolución que se utilizará para resolver otras consultas.

G.7.1.2 Recepción de descriptores

Un elemento de frontera puede solicitar las plantillas configuradas estáticamente de otro elemento de frontera. La respuesta a la petición es decidida por el elemento de frontera al que se solicitan dichas plantillas.

Para solicitar una transferencia, el elemento de frontera envía un mensaje Petición de descriptor (DescriptorRequest) que especifica los descriptores que desea recibir. Si el elemento de frontera propietario puede transferirlos, responde con un mensaje Confirmación de descriptor (DescriptorConfirmation) en el que se especifican todas las plantillas.

El elemento de frontera solicitante puede almacenar una copia de una plantilla recibida de esta manera hasta el final de la vida útil de dicha plantilla; en ese momento el elemento de frontera debe eliminar dicha copia. Si el elemento de frontera propietario cambia sus plantillas configuradas estáticamente antes de que haya finalizado su vida útil, enviará un mensaje Actualización de descriptor (DescriptorUpdate) a los elementos de frontera de los que tiene conocimiento. Al recibir un mensaje DescriptorUpdate, un elemento de frontera debe suprimir, añadir o modificar todas las plantillas indicadas que tiene almacenadas, o debe solicitar al propietario copias de los descriptores indicados.

Un elemento de frontera intermedio (es decir un elemento de frontera que está entre los dominios administrativos de origen y de destino, como por ejemplo un centro de resolución o un punto de agregación) puede publicar sus propios descriptores basándose en los descriptores que recibe. Por ejemplo, un centro de resolución puede indicarse a sí mismo como contacto para un mensaje AccessRequest aunque los descriptores que haya recibido de otro elemento de frontera indiquen a ese otro elemento de frontera como contacto.

Un elemento de frontera puede indicar en una plantilla el requisito para que un originador reciba la autorización de realizar una llamada en un dominio administrativo. Cuando la bandera **Especificación de llamada** (**callSpecific**) se coloca en una plantilla y el tipo de mensaje indica que se enviará un mensaje AccessRequest, el originador proporcionará información por llamada en el mensaje AccessRequest. Si un elemento de frontera recibe el mensaje AccessRequest sin información por llamada y la política es solicitar información por llamada, el elemento de frontera responderá con un mensaje Rechazo de acceso (AccessRejection) con el motivo **Información necesaria sobre la llamada** (**needCallInformation**).

Un elemento de frontera puede enviar un mensaje DescriptorUpdate a otros elementos de frontera conocidos, o puede multidistribuir un mensaje DescriptorUpdate. En este último caso, el elemento de frontera debe considerar el ámbito de la multidifusión. El mensaje DescriptorUpdate puede contener los descriptores que han cambiado. Alternativamente, el mensaje DescriptorUpdate puede indicar únicamente la identificación de los descriptores que cambiaron, pudiendo el destinatario solicitar la nueva información. Si han cambiado muchos descriptores, la información debe enviarse en varios mensajes DescriptorUpdate de modo que un determinado mensaje DescriptorUpdate no exceda del tamaño máximo de paquete de transporte.

G.7.1.3 Recepción de respuestas a consultas específicas

Un elemento de frontera puede enviar un mensaje AccessRequest a otro elemento de frontera pidiéndole la resolución de una dirección total o parcialmente calificada. El mensaje AccessRequest se envía generalmente por un medio de transporte no fiable (por ejemplo, UDP), aunque puede enviarse por un medio de transporte fiable (por ejemplo, TCP).

Al recibir un mensaje AccessRequest, un elemento de frontera hace una búsqueda en su base de datos y responde con la plantilla más específica para el destino. Si varias plantillas satisfacen la petición, el elemento de frontera devolverá todas las plantillas pertinentes. Si el elemento de frontera de destino es responsable de la dirección de alias especificada, el elemento de frontera responderá generalmente con una plantilla que indica que debe enviarse un mensaje AccessRequest o Setup. Si el elemento de frontera de destino es un centro de resolución, responderá generalmente con una plantilla que indica que debe enviarse el mensaje AccessRequest.

El elemento de frontera de destino puede también añadir a la respuesta las plantillas que considera que serán útiles en el futuro. La adición de estas plantillas no debe alargar la respuesta de manera que la red de transporte tenga que fragmentarla (por ejemplo, 576 octetos para IPv4 ó 1200 octetos para IPv6).

Por ejemplo, un elemento de frontera estrechamente acoplado con una pasarela de seguridad puede suministrar dos plantillas en su respuesta a los mensajes AccessRequest: una plantilla de corta duración (algunos minutos o segundos) que especifica a dónde debe enviarse un mensaje Setup y plantillas adicionales que especifican que los mensajes deben enviarse al elemento de frontera para otras direcciones de alias en el dominio administrativo.

Un elemento de frontera puede conservar una plantilla recibida en un mensaje Confirmación de acceso (AccessConfirmation) hasta que deje de ser válida.

G.7.2 Localización de un elemento de frontera o de un conjunto de elementos de frontera

G.7.2.1 Estática

Un elemento de frontera puede tener un conjunto administrado de otros elementos de frontera a los que puede dirigirse para la resolución de direcciones. Este conjunto administrado puede definirse mediante un conjunto de acuerdos bilaterales entre el dominio administrativo y otros dominios administrativos. Los dominios administrativos pueden utilizar opcionalmente el servicio de un centro de resolución.

G.7.2.2 Dinámica

En las redes IP, la propiedad de direcciones de tipo email-ID (ID de correo electrónico) es definida por el sistema DNS. Así, en ausencia de mejor información, un elemento de frontera puede examinar los registros SRV del DNS en la parte del ID de correo electrónico situada a la derecha del signo "@" (por ejemplo, la búsqueda en los registros **_h2250-annex-g._udp.example.org** para **person@example.org**). La respuesta de esta búsqueda debe emplearse para sintetizar una plantilla "enviar AccessRequest" que pueda utilizarse durante el proceso de resolución. Las plantillas sintetizadas a partir de peticiones DNS no deben conservarse más allá de la vida útil indicada en la respuesta del DNS.

G.7.2.3 Otros métodos

Queda en estudio la utilización de otros métodos para localizar otro elemento de frontera.

G.7.3 Procedimientos de resolución

G.7.3.1 Procedimiento de resolución en un dominio administrativo

Cuando se pide a un elemento de frontera que resuelva una dirección de alias (por ejemplo, mediante una pasarela o un guardián de puerta coubicados), encuentra las plantillas pertinentes en su lugar de almacenamiento.

Si hay más de una plantilla pertinente, se seleccionan y ordenan las plantillas apropiadas de acuerdo con la política local. Por ejemplo, las plantillas pueden ordenarse primero según la longitud del comodín (es mejor suministrar plantillas más específicas), luego según el tipo de protocolo especificado ("enviar Setup" es mejor que "enviar AccessRequest").

Si varias plantillas satisfacen la petición, el elemento de frontera indicará todas las plantillas que corresponden.

Si el proceso de selección de plantilla no arroja ninguna plantilla marcada "enviar Setup", el elemento de frontera envía un mensaje AccessRequest con una dirección de destino específica a la dirección especificada en la plantilla. Cuando obtiene una respuesta del elemento de frontera, puede almacenarla e indicar al solicitante la dirección a la que debe enviar el mensaje Setup.

G.7.3.2 Procedimiento de resolución entre dominios administrativos

Cuando un elemento de frontera recibe un mensaje AccessRequest, busca en las plantillas que tiene almacenadas y encuentra las que corresponden a la dirección que figura en la consulta.

Si hay más de una plantilla pertinente, las plantillas se ordenan primero según la longitud del comodín (es mejor utilizar plantillas más específicas). Luego se ordenan de acuerdo con el tipo de mensaje especificado ("enviar Setup" es mejor que "enviar AccessRequest"). En cada caso se descartan todas las plantillas distintas de las que corresponden a la búsqueda más específica.

Si las plantillas que corresponden vienen marcadas "enviar AccessRequest", el elemento de frontera puede reenviar el mensaje AccessRequest al(a los) elemento(s) de frontera especificado(s) en la(s) plantilla(s), o puede devolver las plantillas tal como están. Si el contador de saltos que figura en el mensaje AccessRequest recibido ha llegado a cero, el elemento de frontera no puede reenviar el mensaje AccessRequest a otro elemento de frontera; en lugar de ello, debe devolver las plantillas que corresponden. Si el contador ha llegado a cero y el elemento de frontera no tiene ninguna información para proporcionar en un mensaje AccessConfirmation, el elemento de frontera debe responder con un mensaje AccessRejection que indique que se ha rebasado el cómputo de saltos.

En esta etapa y para autorizar la petición de acceso, el elemento de frontera puede utilizar un elemento de frontera de un tercer dominio administrativo (por ejemplo, un centro de resolución). Para ello, envía un mensaje Petición de validación (ValidationRequest), que lleva los testigos de acceso suministrados por el elemento de frontera solicitante en AccessRequest. El elemento de frontera destinatario valida los testigos y devuelve el mensaje Confirmación de validación (ValidationConfirmation).

El elemento de frontera devuelve entonces un mensaje AccessConfirmation con las plantillas que ha hallado (éstas tendrán la misma dirección y los mismos tipos de mensaje) así como cualesquiera otras plantillas que considere útiles.

Si varias plantillas satisfacen la petición, el elemento de frontera devolverá todas las plantillas que corresponden.

Si la petición de acceso contiene información de llamada específica, las plantillas devueltas son válidas únicamente para la llamada solicitada. Esto se utiliza cuando un dominio administrativo desea conceder el acceso llamada por llamada. En este caso, el dominio administrativo puede imponer la inclusión de la información de llamada para cada AccessRequest que le es enviada, colocando una bandera en las plantillas que hacen referencia a él.

G.7.4 Intercambio de información sobre uso

Los dominios administrativos pueden solicitar a otros dominios que les proporcionen información sobre la utilización de recursos en determinadas llamadas. Los mensajes Indicación de uso (UsageIndication) pueden suministrarse en cualquier etapa de la llamada. Además, pueden enviarse múltiples indicaciones de uso para la misma llamada, con información cada vez más actualizada.

Las indicaciones de uso pueden intercambiarse únicamente si ambos elementos de frontera mantienen relaciones de servicio.

Se enviarán peticiones UsageIndication cuando un elemento de frontera lo requiera, ya sea en las plantillas para las que sirve de contacto, o bien indicándolo en cualquiera de los mensajes Petición de uso (UsageRequest), AccessRequest, ValidationRequest y ValidationConfirmation enviados en el contexto de la llamada para la que se solicitó UsageIndication.

G.8 Protocolo

En el protocolo del anexo G los mensajes pueden enviarse a través de un servicio de transporte no fiable (por ejemplo, UDP) o a través de un servicio de transporte fiable (por ejemplo, TCP) a una dirección bien conocida. En las redes IP, el puerto 2099, suficientemente conocido, debería utilizarse tanto para TCP como para UDP, a menos que se haya comunicado otro puerto al emisor. Los elementos de frontera serán atendidos en los dos puertos mencionados.

Cuando los mensajes se envían por el servicio de transporte fiable, pueden enviarse varios mensajes dentro de las fronteras definidas por la unidad de datos de protocolo (PDU) de transporte fiable, siempre que se envíen mensajes enteros. (En implementaciones IP, como se indica en el apéndice IV/H.225.0, esta PDU viene definida por TPKT.)

Cuando se utiliza un servicio de transporte no fiable, los mensajes de petición pueden ser retransmitidos. Un método adaptable sensible al retardo (como el utilizado por el protocolo TCP) debería determinar el valor por defecto del temporizador de retransmisión. En las retransmisiones subsiguientes se utilizará la reducción exponencial. No debe haber más de 5 retransmisiones. Las respuestas no serán retransmitidas.

En implementaciones UDP IP, los mensajes tendrán además como prefijo encabezadores TPKT para permitir múltiples mensajes por paquete. El campo longitud del paquete UDP contendrá la longitud total de la cabida útil, incluidos todos los mensajes y sus encabezamientos TPKT.

G.8.1 Consideraciones en materia de seguridad

Cuando se desea autenticación, integridad y criptación para los mensajes intercambiados entre elementos de frontera, la seguridad IP funcionará según lo indicado en la RFC 1825 del IETF ("Security Architecture for the Internet Protocol"), así como en la RFC 1826 del IETF ("IP Authentication Header") y/o la RFC 1827 del IETF ["IP Encapsulating Security Payload (ESP)"].

Cuando proceda, se utilizarán los procedimientos y modelos de H.235 para la seguridad a nivel de aplicación. Específicamente, se utilizarán los intercambios de autenticación y formatos de testigo. Los testigos y los testigos de criptado recibidos en los mensajes de respuesta deben utilizarse en una petición conexa subsiguiente.

G.8.2 Definiciones de mensaje

Cada mensaje contiene un conjunto de campos comunes además de la información específica del mensaje. Los campos comunes son:

Campo	Descripción
sequenceNumber (número de secuencia)	Cada mensaje de petición o actualización contiene un número de secuencia único. El mensaje enviado en respuesta a un mensaje de petición (un mensaje de confirmación o de rechazo) utiliza el número de secuencia del mensaje de petición. Los mensajes retransmitidos tendrán el mismo número de secuencia.
ReplyAddress (Dirección para la respuesta)	Dirección a la que ha de enviarse la respuesta a un mensaje de petición. Cualquier mensaje de petición incluirá una replyAddress, a menos que la petición se haya enviado por un transporte de conexión bidireccional (por ejemplo, TCP). Cualquier otro mensaje que no sea un mensaje de petición no incluirá una replyAddress.
version (Versión)	Versión del protocolo utilizado por el remitente del mensaje.
hopCount (cómputo de saltos)	Define el número de elementos de frontera a través de los cuales puede propagarse el mensaje. Cuando un elemento de frontera recibe este mensaje y decide que el mensaje debe reenviarse a otro elemento de frontera, empieza por decrementar el *hopCount*. Si éste es mayor que 0, el elemento de frontera inserta el nuevo valor del contador en el mensaje que debe transmitirse. Si el *hopCount* ha llegado a 0, el elemento de frontera no reenviará el mensaje. Si el mensaje es una petición, el elemento de frontera debe responder con un mensaje de confirmación que contenga la información pertinente. Si no la hay, el elemento de frontera debe responder con un mensaje de rechazo.
IntegrityCheckValue (Valor de verificación de integridad)	Proporciona integridad/autenticación mejoradas del mensaje. El emisor calcula el valor de verificación de integridad criptado aplicando a todo el mensaje un algoritmo de integridad negociado y la clave secreta. Antes del cálculo de integrityCheckValue, cada byte de este campo se pondrá en cero. Tras el cálculo, el emisor coloca el valor de verificación de integridad calculado en el campo integrityCheckValue y transmite el mensaje. Ciertos datos pueden ser necesarios para el funcionamiento. Los datos se insertarán en el mensaje, si existe.
Tokens (Testigos)	
CryptoTokens (Testigos de criptado)	Testigos criptados.
nonStandard (No normalizada)	Información no normalizada.

G.8.2.1 Descriptor

El descriptor no es un mensaje, sino un elemento de mensaje utilizado para etiquetar un conjunto de plantillas.

El descriptor contiene la siguiente información:

Campo	Descripción
descriptorInfo (información sobre el descriptor)	Contiene el identificador único del descriptor y el momento en que fue modificado por última vez (véase más adelante Información de descriptor).
Templates (Plantillas)	Conjunto de plantillas que definen las direcciones que puede resolver este descriptor.
GatekeeperID (ID de guardián de puerta)	Identificador (texto) que indica el propietario del descriptor (es decir, el guardián de puerta que creó este mensaje).

G.8.2.2 Información de descriptor

La información de descriptor identifica de manera única el momento en que el descriptor fue modificado por última vez.

Campo	Descripción
descriptorID (ID de descriptor)	Identificador único utilizado para identificar este descriptor entre muchos descriptores posibles.
lastChanged (última modificación)	Fecha y hora en que este descriptor fue modificado por última vez.

G.8.2.3 Plantilla de dirección

La plantilla de dirección describe un conjunto de una o varias direcciones de alias. La plantilla no es un mensaje, sino un elemento utilizado como bloque constitutivo de otros elementos. Consta de otras estructuras, que se describen en las subcláusulas siguientes.

Campo	Descripción
Pattern (Patrón)	Lista de patrones (véase más adelante patrón).
RouteInfo (Información de ruta)	Lista de información de ruta para esta plantilla (véase más adelante Información de ruta).
TimeToLive (Tiempo de vida)	Indica el tiempo, expresado en segundos, durante el cual esta plantilla es válida.

G.8.2.3.1 Información de ruta

La estructura información de ruta (el campo *routeInfo*) encontrada en la *plantilla* contiene lo siguiente:

Campo	Descripción
MessageType (Tipo de mensaje)	Indica el tipo de mensaje que debe enviarse cuando se intenta resolver una dirección específica en esta plantilla. Las posibilidades son Enviar petición de acceso (sendAccessRequest), Enviar establecimiento (sendSetup) o No existente (nonExistent) (indica que la dirección no existe).

CallSpecific (Especificación de llamada)	Si se pone en VERDADERO, quiere decir que se solicita autorización para cada llamada en esta ruta, con lo cual el mensaje AccessRequest incluirá la información de llamada. Este campo booleano es significativo únicamente cuando *messageType* es sendAccessRequest; en caso contrario, *callSpecific* se pondrá en FALSO.
UsageSpec (Especificación de uso)	Si está presente, indica los mensajes UsageIndication que serán enviados en relación con las llamadas efectuadas en esta ruta.
PriceInfo (Información sobre precios)	Lista de información sobre precios para esta ruta en particular (véase más adelante Información sobre precios). Obsérvese que múltiples pasarelas con diferentes estructuras de precios deberían describirse en múltiples estructuras *RouteInformation*.
contacts (contactos)	Información de contacto para el elemento que aceptará el mensaje especificado en el campo *messageType* de routeInfo. La información de contacto puede darse como una lista de contactos posibles (véase más adelante la descripción de la Información de contacto).
type (tipo)	Indica el tipo de punto extremo que puede atender la llamada. Para los casos de encaminamiento por guardián de puerta, indica los tipos de puntos extremos atendidos por el guardián de puerta y no el propio guardián de puerta.

G.8.2.3.2 Información sobre precios

La información sobre precios aparece como un elemento en la estructura Información de ruta (el campo priceInfo). La información sobre precios se define mediante las estructuras PriceInfoSpec y PriceElement.

La estructura especificación de la información sobre precios (PriceInfoSpec) contiene los campos siguientes:

Campo	**Descripción**
currency (moneda)	Designador de moneda ISO 4217.
currencyScale (escala de la moneda)	Número de unidades que hay que desplazar el punto de base implícito hacia la izquierda. Por ejemplo, cuando *currency* es USD, el valor 2 de *currencyScale* significaría que la suma indicada en *priceElement* está expresada en céntimos de USD.
validFrom (válido desde)	Fecha y hora a partir de las cuales esta información es válida.
validUntil (válido hasta)	Fecha y hora en las cuales caduca esta información.
hoursFrom (desde)	Hora del día en que empieza esta tarifa.
hoursUntil (hasta)	Hora del día en que termina esta tarifa. Puede tener un valor inferior a *hoursFrom*, lo que indica que la tarifa abarca la hora 0000.
priceElement (elemento de precio)	Lista opcional de elementos de precio que se suman para constituir el precio.
priceFormula (fórmula de precio)	Cadena opcional que contiene una fórmula de precio utilizada como alternativa al elemento de precio estructurado.

La estructura Elemento de precio (PriceElement) contiene los campos siguientes:

Campo	Descripción
amount (suma)	Incremento del medidor. El medidor se incrementa una vez por cada *cuanto* o fracción de *cuanto*.
quantum (cuanto)	Número de unidades para las cuales se aplica *amount*. Por ejemplo, un valor de 60, con unidades (*units*) en segundos, indica que la llamada se tarifica por minuto o fracción de minuto. Si el campo *units* se pone en cualquiera de los valores *initial*, *minimun* o *maximum*, el campo *quantum* es irrelevante, y el destinatario ignorará su valor.
units (unidades)	Tipo de unidad en la que se expresa el cuanto: • segundos (seconds) – segundos de duración de la llamada; • paquetes (packets) – paquetes transmitidos o recibidos; • bytes – bytes transmitidos o recibidos; • inicial (initial) – tasa de conexión inicial; • mínimo (minimum) – tasa de llamada mínima; • máximo (maximum) – tasa de llamada máxima.

G.8.2.3.3 Información de contacto

La estructura información de contacto (Contact Information) es un elemento (el campo *contacts*) de la estructura Información de ruta (Route Information).

Campo	Descripción
transportAddress (dirección de transporte)	Dirección (por ejemplo, dirección de transporte o URL) a la cual hay que enviar el mensaje especificado en el campo *messageType* de la estructura Route Information. Siempre que sea posible, se utilizará una dirección de transporte.
priority (prioridad)	Cuando se enumeran múltiples contactos, el campo *priority* especifica el orden en el que éstos deben ensayarse. Los contactos que figuran en la lista pueden compartir una prioridad, por ejemplo si no hay preferencia con relación al orden en el que deben ensayarse. El valor 0 indica la máxima prioridad (primera elección).
transportQoS (calidad de servicio de transporte)	Indica donde recae la responsabilidad por la reserva de recursos para todas las llamadas realizadas mediante este contacto.
Security (seguridad)	Mecanismo de seguridad que describe el orden de preferencia que debe utilizarse al establecer la comunicación con el contacto.
AccessTokens (testigos de acceso)	Conjunto de testigos que se enviarán en el mensaje transmitido a este contacto (Setup o AccessRequest). Estos testigos se enviarán también en los mensajes UsageIndication subsiguientes relativos a las llamadas que utilizan esta plantilla.

G.8.2.3.4 Patrón

La estructura de patrón aparece en la plantilla de dirección. Gracias al patrón, se puede indicar una dirección de alias, una dirección de alias de comodín o una gama de direcciones de alias.

Campo	Descripción
Specific (Específica)	Dirección de alias específica
Wildcard (Comodín)	Definición jerárquica que representa una posible expansión de la cadena. Para los números E.164 esta expansión es posible al final del número; para las direcciones de correo electrónico, la expansión es posible al comienzo. Por ejemplo, si *wildcard* es "+1 303", el patrón podría representar cualquier número en la zona de Denver.
Range (Gama)	Gama de direcciones, comprendidos el inicio y fin indicados de la gama.

G.8.2.4 Estructuras comunes

Las estructuras definidas en esta subcláusula aparecen en muchos de los mensajes.

G.8.2.4.1 Elemento de frontera alternativa (AlternateBE)

Campo	Descripción
contactAddress (dirección de contacto)	Dirección de transporte del elemento de frontera alternativo (la dirección a la que hay que enviar los mensajes del anexo G).
Priority (Prioridad)	Cuando se enumeran múltiples alternativas, el campo *priority* especifica el orden en el que éstas deben ensayarse. Las alternativas que figuran en la lista pueden compartir una prioridad, por ejemplo si no hay preferencia en cuanto al orden en el que deben ensayarse. El valor 0 indica la máxima prioridad (primera elección).
ElementIdentifier (Identificador de elemento)	Este elemento de frontera alternativo utiliza esta cadena unicódigo como identificador.

G.8.2.4.2 Información de parte (PartyInformation)

Esta estructura contiene información relativa a una parte de la llamada (origen o destino).

Campo	Descripción
LogicalAddress (Dirección lógica)	Direcciones formateadas de correo electrónico o E.164 que identifican la parte.
DomainIdentifier (Identificador de dominio)	Dirección de alias que identifica el AD que originó o dio por terminada la llamada. Si en la realización de una llamada participan múltiples dominios, debería establecerse el dominio que actúa como origen o terminación de la llamada desde el punto de vista del emisor.
TransportAddress (Dirección de transporte)	Dirección de transporte del punto extremo.
EndpointType (Tipo de punto extremo)	Proporciona detalles sobre el tipo de punto extremo y sus capacidades.

UserInfo (Información de usuario)	Información relativa al usuario que efectúa la llamada. Puede incluir la identificación en el correo electrónico o en el formato del número de identificación personal (PIN) y las posibles credenciales de autenticación.
TimeZone (Huso horario)	Huso horario de la parte, que es importante en relación con la fijación de precios. Si la parte que originó la llamada es una pasarela, debe transmitirse el huso horario de la pasarela. Se describe en segundos con relación a UTC.

G.8.2.4.3 Información sobre la llamada (CallInformation)

Información para identificar una llamada específica.

Campo	Descripción
CallIdentifier (Identificador de llamada)	Identificación única de la llamada. Esta identificación será el callIdentifier asociado con la misma llamada como en RAS y en los mensajes de señalización de llamada.
ConferenceID (ID de conferencia)	Identificación única de la conferencia a la que pertenece la llamada. Esta identificación será el conferenceID asociado con la misma llamada como en RAS y en los mensajes de señalización de llamada.

G.8.2.4.4 Información de usuario (UserInformation)

Información destinada a identificar al usuario en cualquier parte de la llamada.

Campo	Descripción
UserIdentifier (Identificador de usuario)	Identifica de manera única al usuario.
UserAuthenticator (Autenticador de usuario)	Testigos criptados para una autenticación segura.

G.8.2.4.5 Especificación de uso

Este elemento describe los parámetros que es necesario comunicar en los mensajes UsageIndication. El contexto del mensaje que contiene el elemento *UsageSpecification* determina las llamadas en las cuales se aplica esta especificación.

Campo	Descripción
SendTo (Enviar a)	Elemento de frontera al que se envían los mensajes UsageIndication. Como el emisor debería tener relaciones de servicio con el elemento de frontera, éste es el identificador de elemento devuelto en el mensaje Confirmación de servicio (ServiceConfirmation).
When (Cuándo)	Indica las etapas de la llamada y la frecuencia con que deben enviarse las indicaciones: • Nunca – Detener el envío de mensajes. • Inicio – Cuando empieza la llamada. • Fin – Al final de la llamada, o a partir de entonces.

	• Periodo – Periódicamente, mientras dura la llamada. El periodo se mide en segundos. • Fallo – Comunica los intentos de llamada que han fracasado.
Required (Necesario)	Lista de identificadores para los campos que *deben* estar presentes en los mensajes *UsageIndication*. El emisor de la información sobre uso rechazará o ignorará el mensaje que contiene este mensaje, si no puede suministrar estos campos.
Preferred (Preferido)	Lista de identificadores para los campos que *deberían* estar presentes en los mensajes *UsageIndication*.

G.8.2.4.6 Modo seguridad

Este elemento describe un perfil de seguridad específico que debe utilizarse para la comunicación del anexo G.

Campo	Descripción
Authentication (Autenticación)	Indica el mecanismo de autenticación que va a utilizarse. Este mecanismo debe seleccionarse a partir del conjunto proporcionado en el mensaje Petición de servicio (ServiceRequest).
Integrity (Integridad)	Indica el mecanismo de integridad que va a utilizarse. Si está presente, todos los mensajes subsiguientes rellenarán el campo *integrityCheckValue*; en este caso, *Modo de autenticación* (*AuthenticationMode*) describe la forma en que se generan las claves secretas (intercambio DH, o *a priori*).
AlgorithmOID	Indica el algoritmo de criptación del mecanismo de seguridad.

G.8.2.5 Petición de servicio

Un elemento de frontera puede enviar un mensaje ServiceRequest a otro elemento de frontera para establecer una relación de servicio. La relación define los mecanismos de seguridad que deben utilizarse entre los elementos de frontera y permite la identificación de elementos de frontera alternativos o de reserva. Obsérvese que la relación es unidireccional. La seguridad negociada entre los dos elementos de frontera se utiliza para las peticiones enviadas por el elemento de frontera que ha enviado la petición de servicio y para las respuestas enviadas por el receptor de la misma. Durante el proceso del establecimiento de la relación de servicio pueden generarse claves de sesión que serán válidas mientras dure esa relación. A tal fin, pueden utilizarse testigos, tal como se indica en la Recomendación H.235.

El receptor de la petición de servicio puede indicar elementos de frontera alternativos que el emisor de la misma puede ensayar para el servicio de reserva. El establecimiento de una relación de servicio es obligatorio para los intercambios del mensaje indicación de uso. En otros casos, es un procedimiento opcional, aunque la política de un elemento de frontera puede necesitar esa relación.

Un elemento de frontera puede enviar un mensaje petición de servicio a un elemento de frontera con el que ya tiene una relación, con el fin de dar por terminados los términos de la relación original y reemplazarlos por los nuevos términos. Las relaciones de servicio pueden tener un tiempo de vida limitado. Un elemento de frontera puede renovar la relación enviando una nueva petición de servicio.

Campo	Descripción
ElementIdentifier (Identificador de elemento)	Cadena que identifica el BE que envía la petición.
DomainIdentifier (Identificador de dominio)	AD que solicita la relación de servicio.
SecurityCapability (Capacidad de seguridad)	Conjunto de mecanismos de seguridad que este elemento de frontera puede soportar.
TimeToLive (Tiempo de vida)	Tiempo de vida propuesto de la relación de servicio indicado en segundos. Si no está presente, se supone que el tiempo de vida es infinito.

G.8.2.6 Confirmación de servicio

Al recibir un mensaje petición de servicio, un elemento de frontera responde con un mensaje confirmación de servicio para indicar que está de acuerdo en establecer una relación de servicio. Si el elemento de frontera ya tiene una relación de servicio con el elemento de frontera que ha enviado el mensaje petición de servicio, el enviar ServiceConfirmation indica la terminación de los términos de la relación original y su sustitución por los nuevos términos.

Campo	Descripción
elementIdentifier (identificador de elemento)	Cadena que identifica el elemento de frontera.
alternates (alternativos)	Lista de elementos de frontera alternativos con los que puede tomarse contacto en caso de que este elemento de frontera no responda.
DomainIdentifier (Identificador de dominio)	AD que responde a la petición.
SecurityMode (Modo de seguridad)	Indica el mecanismo de seguridad que debe utilizarse para esta relación de servicio. El mecanismo de seguridad debe escogerse en el conjunto suministrado en el mensaje ServiceRequest.
TimeToLive (Tiempo de vida)	Tiempo de vida en segundos de la relación de servicio determinado por el elemento de frontera servidor.

G.8.2.7 Rechazo de servicio

Al recibir un mensaje ServiceRequest, un elemento de frontera responde con un mensaje Rechazo de servicio (ServiceRejection) para indicar que se niega a establecer una relación de servicio. Si el elemento de frontera ya tiene una relación de servicio con elemento de frontera que ha enviado el mensaje ServiceRequest, el envío de ServiceRejection indica que los nuevos términos propuestos han sido rechazados, pero que los términos de la relación original siguen vigentes.

Campo	Descripción
reason (motivo)	Motivo por el cual el elemento de frontera ha rechazado la petición de servicio. Puede ser: • Servicio no disponible (ServiceUnavailable) – Este elemento de frontera no está disponible actualmente para el servicio.

	• Servicio redireccionado (ServiceRedirected) – Debe ensayarse la lista de elementos de frontera alternativos. • Seguridad (Security) – Este elemento de frontera no puede soportar ninguno de los mecanismos de seguridad propuestos en el mensaje ServiceRequest. • Continuación (Continue) – Indica que se envíe el mensaje ServiceRequest subsiguiente a fin de continuar las múltiples etapas del proceso de intercambio de claves. • Indefinido (Undefined) – El motivo del rechazo de ServiceRequest no corresponde a ninguna de las otras posibilidades.
Alternates (Alternativos)	Lista de elementos de frontera alternativos que podrían satisfacer la petición de servicio. Si el motivo es *serviceRedirected*, debe darse por lo menos una alternativa.

G.8.2.8 Liberación de servicio

Cualquiera de los elementos de frontera de una relación de servicio puede terminar la relación enviando el mensaje Liberación de servicio (ServiceRelease).

Campo	**Descripción**
reason (motivo)	Motivo por el cual este elemento de frontera ha terminado la relación de servicio. Puede ser: • Fuera de servicio (OutOfService) – El elemento de frontera va a ponerse fuera de servicio. • Mantenimiento (Maintenance) – El elemento de frontera va a sacarse de servicio para mantenimiento. • Terminado (Terminated) – El elemento de frontera ha decidido terminar la relación. • Expirado (Expired) – El tiempo de vida de la relación de servicio ha terminado.
Alternates (Alternativos)	Lista de elementos de frontera alternativos que podrían establecer una relación de servicio.

G.8.2.9 Petición de Descriptor

Mediante el mensaje DescriptorRequest, una entidad puede solicitar descriptores específicos a un elemento de frontera.

Campo	**Descripción**
descriptorID (ID de descriptor)	Identifica uno o varios descriptores en particular solicitados por el emisor de este mensaje.

G.8.2.10 Confirmación de Descriptor

El mensaje DescriptorConfirmation es la respuesta positiva de un elemento de frontera a una petición de descriptor cuando el elemento de frontera puede interpretar la petición y las reglas de la realización permiten el intercambio de información.

Campo	Descripción
descriptores	Estos son los *descriptores* descritos anteriormente.

G.8.2.11 Rechazo de descriptor

Un elemento de frontera puede rechazar una petición de descriptor por varios motivos.

Campo	Descripción
reason (motivo)	Motivo por el que ha sido rechazada la petición de descriptor. Puede ser: • Tamaño de paquete rebasado (PacketSizeExceeded) – La respuesta excedería del tamaño de paquete máximo, de modo que el solicitante debería enviar la petición utilizando un mecanismo de transporte diferente (por ejemplo, TCP en vez de UDP). • ID ilegal (illegalID) – El receptor de la petición de descriptor no tiene registro del descriptor solicitado. • seguridad (security) – La petición de descriptor no satisfizo los requisitos de seguridad del receptor. • Cómputo de saltos rebasado (HopCountExceeded)– El cómputo de saltos llegó a cero y no hay información disponible. • no disponible (unavailable) – El destinatario no puede proporcionar descriptores. Debería utilizarse el método de suministro estático o fuera de banda. • relación de servicio inexistente (noServiceRelationship) – El destinatario intercambiará esta información únicamente después del establecimiento de una relación de servicio. • indefinido (undefined) – El motivo del rechazo de la petición de descriptor no corresponde a las demás posibilidades.
DescriptorID (ID de descriptor)	Identifica el descriptor específico para esta respuesta.

G.8.2.12 Petición de ID de descriptor

Mediante la Petición de ID de descriptor (DescriptorIDRequest), una entidad puede pedir a un elemento de frontera la lista de identificadores de descriptor dentro del dominio administrativo del elemento de frontera.

G.8.2.13 Confirmación de ID de descriptor

Un mensaje Confirmación de ID de descriptor (DescriptorIDConfirmation) es la respuesta positiva de un elemento de frontera al mensaje DescriptorIDRequest. Al recibir un mensaje DescriptorIDConfirmation, un elemento de frontera puede enviar el mensaje DescriptorRequest para solicitar la transmisión de los descriptores.

Campo	Descripción
descriptorInfo (información sobre los descriptores)	Lista de información sobre los descriptores, en la que cada entrada identifica de manera única el descriptor y el momento que ha cambiado por última vez.

G.8.2.14 Rechazo de ID de descriptor

Un elemento de frontera puede rechazar una DescriptorIDRequest por varios motivos.

Campo	Descripción
reason (motivo)	Indica el motivo del rechazo de la petición. Puede ser: • no hay descriptores (noDescriptors) – Indica que el elemento de frontera no tiene descriptores. • seguridad (security) – La petición de ID de descriptor no satisfizo los requisitos de seguridad del receptor. • cómputo de saltos rebasado (hopCountExceeded) – El cómputo de saltos llegó a cero y no hay información disponible. • no disponible (unavailable) – El destinatario no puede proporcionar descriptores. Debería utilizarse el método de suministro estático o fuera de banda. • Relación de servicio inexistente (NoServiceRelationship) – El destinatario intercambiará esta información únicamente después del establecimiento de una relación de servicio. • indefinido (undefined) – El motivo del rechazo de la petición de ID de descriptor no corresponde a las otras posibilidades.

G.8.2.15 Actualización de descriptor

El mensaje DescriptorUpdate es la notificación de un elemento de frontera de que la información de dirección ha cambiado. Un elemento de frontera puede también enviar el mensaje DescriptorUpdate durante la inicialización. Al recibir el mensaje DescriptorUpdate, un elemento de frontera puede solicitar información del elemento identificado en ese mensaje.

Campo	Descripción
sender (emisor)	Al recibir el mensaje DescriptorUpdate, un elemento puede enviar una petición a esta dirección (por ejemplo, dirección de transporte o URL).
UpdateInfo (Información de actualización)	Lista de las actualizaciones. Cada entrada de la lista da, ya sea el descriptor o el identificador del descriptor que ha sido actualizado. Cada entrada indica también si el descriptor ha sido modificado, añadido o suprimido.

G.8.2.16 Acuse de recibo de actualización de descriptor

Un elemento de frontera debe acusar recibo del mensaje DescriptorUpdate enviando el mensaje Acuse de recibo de actualización de descriptor (DescriptorUpdateAck). El número de secuencia utilizado en el acuse de recibo debe ser el mismo que el número de secuencia recibido en el mensaje

DescriptorUpdate. Un elemento de frontera no debe acusar recibo de un mensaje DescriptorUpdate que llega por multidistribución.

G.8.2.17 Petición de acceso

Un elemento de frontera puede enviar un mensaje Petición de acceso (AccessRequest) a otro elemento de frontera para pedir la resolución de una dirección de alias específica.

Campo	Descripción
DestinationInfo (Información de destino)	Dirección que debe resolverse.
SourceInfo (Información sobre el origen)	Información sobre la parte que dio origen a la llamada cuyo acceso se solicita.
CallInfo (Información sobre la llamada)	Proporciona la identificación de la llamada cuya autorización de acceso se solicita. Si no está presente, la petición corresponde a llamadas indefinidas a destinos especificados.
UsageSpec (Especificación de uso)	Indica los mensajes sobre uso relativos a la llamada solicitada en este mensaje cuyo envío solicita la parte origen de la llamada a la parte que responde a esa llamada. Se aplica únicamente si *CallInfo* está presente.

G.8.2.18 Confirmación de acceso

Un elemento de frontera devuelve en el mensaje AccessConfirmation la información solicitada en el mensaje AccessRequest.

Campo	Descripción
templates (plantillas)	Lista de plantillas que corresponden a los atributos de AccessRequest.
PartialResponse (Respuesta parcial)	Si es VERDADERO, este mensaje contiene una fracción de la información disponible. La información completa no ha sido enviada porque excede del tamaño del paquete. La información completa debe recuperarse utilizando otro medio de transporte (por ejemplo, TCP).

G.8.2.19 Rechazo de acceso

Un elemento de frontera puede rechazar una petición de acceso por varios motivos.

Campo	Descripción
reason (motivo)	Motivo del rechazo de la petición. Puede ser: • No hay correspondencia (NoMatch) – El destino especificado en la petición de acceso no puede ser resuelto. • Tamaño de paquete rebasado (PacketSizeExceeded) – La respuesta excedería del tamaño de paquete máximo, de modo que el solicitante debe enviar la petición utilizando un mecanismo de transporte diferente (por ejemplo, TCP en lugar de UDP). • seguridad (security) – La petición de acceso no satisfizo los requisitos de seguridad del receptor.

	• Cómputo de saltos rebasado (HopCountExceeded) – El cómputo de saltos llegó a cero y no hay información disponible.• Relación de servicio inexistente (NoServiceRelationship) – El destinatario intercambiará esta información únicamente después del establecimiento de una relación de servicio. • Información necesaria sobre la llamada (CallInfoNeeded) – La información específica sobre la llamada no estaba presente en la petición. • Indefinido (Undefined) – El motivo del rechazo de la petición de acceso no corresponde a las demás posibilidades de elección.

G.8.2.20 Petición en proceso

Un elemento de frontera puede devolver el mensaje Petición en progreso (RequestInProgress) para indicar que el tiempo necesario por el elemento de frontera para responder a una petición puede exceder de los intervalos de respuesta normalmente esperados. El número de secuencia será el mismo que el número de secuencia hallado en la petición para la cual se enviará este mensaje.

Campo	Descripción
delay (plazo)	Longitud de tiempo prevista, en milisegundos, para que el elemento de frontera responda a la petición original.

G.8.2.21 Petición no normalizada

El mensaje Petición no normalizada (NonStandardRequest) puede ser enviado desde un elemento de frontera para indicar un mensaje de petición no definido en el anexo G. La información no normalizada es transportada en el elemento *no normalizada* (*nonStandard*) de *información común del anexo* (*G AnnexGCommonInfo*).

G.8.2.22 Confirmación no normalizada

El mensaje Confirmación no normalizada (NonStandardConfirmation) puede ser enviado desde un elemento de frontera en respuesta a un mensaje NonStandardRequest. La información no normalizada es transportada en el elemento *no normalizada* de *información común del anexo G*.

G.8.2.23 Rechazo no normalizado

El mensaje Rechazo no normalizado (NonStandardRejection) puede ser enviado desde un elemento de frontera en respuesta a un mensaje NonStandardRequest. La información no normalizada es transportada en el elemento *nonStandard* de *AnnexGCommonInfo*.

Campo	Descripción
Reason (motivo)	Indica el motivo del rechazo de la petición. Puede ser: • no soportado (notSupported) – El destinatario comprende que se trata de un mensaje NonStandardRequest, pero no comprende ni admite los datos no normalizados. • relación de servicio inexistente (noServiceRelationship) – El destinatario intercambiará esta información únicamente después del establecimiento de una relación de servicio.

	• indefinido (undefined) – El motivo del rechazo de NonStandardRequest no corresponde a las otras posibilidades.

G.8.2.24 Respuesta a mensaje desconocido

Al recibir un mensaje que no entiende, un elemento de frontera debe responder al emisor de dicho mensaje con el mensaje Respuesta a mensaje desconocido (UnknownMessageResponse). El elemento de frontera no debe utilizar este mensaje si otro mensaje del anexo G puede constituir una respuesta apropiada (por ejemplo, un rechazo de descriptor sería la respuesta apropiada a una petición de descriptor cuyo identificador de descriptor no es válido).

Campo	Descripción
unknownMessage (mensaje desconocido)	Contenido del mensaje desconocido.
Reason (motivo)	Motivo por el que se utiliza el mensaje UnknownMessageResponse. Puede ser: • no entendido (notUnderstood) – El mensaje no ha sido entendido. • indefinido (undefined) – El motivo del envío del mensaje UnknownMessageResponse no corresponde a ninguna de las otras posibilidades.

G.8.2.25 Petición de uso

Petición dirigida al destinatario para que envíe mensajes UsageIndication relativos a una determinada llamada.

Campo	Descripción
CallInfo (información sobre la llamada)	Llamada a la cual se envía la indicación.
UsageSpec (especificación de uso)	Indica cuándo deben recibirse las indicaciones y qué deben contener.

G.8.2.26 Confirmación de uso

El mensaje UsageConfirmation se envía en respuesta a un mensaje UsageRequest para indicar que el destinatario aceptó la petición y enviará indicaciones de uso.

G.8.2.27 Rechazo de uso

El mensaje UsageRejection se envía en respuesta a un mensaje UsageRequest para indicar que el destinatario rechazó la petición y no enviará las indicaciones de uso.

Campo	Descripción
Reason (motivo)	Indica el motivo por el cual el elemento de frontera rechazó la petición de uso. Puede ser: • llamada no válida (InvalidCall). • seguridad (Security). • no disponible (Unavailable). • relación de servicio inexistente (noServiceRelationship). • indefinido (Undefined).

G.8.2.28 Indicación de uso

Este mensaje comunica los detalles de la llamada y la información sobre uso. Se envía con respecto al último elemento *UsageSpecification* recibido por el BE relativo a la llamada.

Campo	Descripción
CallInfo (Información sobre la llamada)	Llamada a la cual se aplica la indicación.
AccessTokens (Testigos de acceso)	Testigos de acceso para la llamada que fueron recibidos en la plantilla de dirección utilizada para esa llamada y propagados en el mensaje AccessRequest/Setup para la misma llamada.
SendRole (Función del emisor)	Indica la función del emisor de la indicación: • origen (originator) – parte que origina la llamada; • destino (destination) – parte que termina la llamada; • no normalizada (nonStandard) – otros.
UsageCallStatus (Situación de la llamada)	Indica la situación actual de la llamada: • antes de la conexión (preConnect); • llamada en curso (callInProgress); • llamada terminada (callEnded).
SourceAddress (Dirección del origen)	Dirección de correo electrónico o E.164 de la parte llamante. En el caso de E.164, designa la identificación de la línea llamante/identificación del número (ANI/CLI).
DestAddress (Dirección del destino)	Dirección de correo electrónico o E.164 de la parte llamada.
StartTime (Hora de inicio)	Hora en que se inicia la llamada en el formato UTC. Se aplica únicamente a las llamadas que cumplieron la etapa de establecimiento.
EndTime (Hora de terminación)	Hora en que termina la llamada en el formato UTC. Se aplica únicamente a llamadas terminadas.
TerminationCause (Causa de la terminación)	Indica el motivo de la terminación de la llamada. Se aplica únicamente a llamadas terminadas.
usageInformation (Información sobre uso)	Conjunto de campos de información. Cada campo está representado por un *Campo de uso* (*UsageField*) que puede estar normalizado o no. Quedan en estudio los campos de uso normalizados (StandardUsageFields).

G.8.2.29 Confirmación de indicación de uso

El mensaje UsageIndicationConfirmation se envía en respuesta al mensaje UsageIndication; indica que el destinatario aceptó la indicación comunicada.

G.8.2.30 Rechazo de indicación de uso

El mensaje UsageIndicationRejection se envía en respuesta al mensaje UsageIndication; indica que el destinatario rechazó la indicación y la ignorará.

Campo	Descripción
Reason (motivo)	Indica el motivo por el cual el elemento de frontera rechazó el mensaje UsageIndication. Puede ser: • Llamada no válida (InvalidCall); • Seguridad (Security); • Relación de servicio inexistente (NoServiceRelationship); • Indefinido (Undefined).

G.8.2.31 Petición de validación

Un elemento de frontera que termina una llamada puede enviar un mensaje ValidationRequest a otro elemento de frontera para verificar la validez del origen de la llamada.

Campo	Descripción
DestinationInfo (Información del destino)	Indica detalles sobre el destino de la llamada.
SourceInfo (Información sobre el origen)	Información sobre el tipo de punto extremo que originó la llamada.
CallInfo (Información sobre la llamada)	Proporciona la identificación de la llamada cuya autorización de acceso se solicita.
UsageSpec (Especificación de uso)	Si está presente, indica el elemento de frontera que envía las peticiones de mensaje; solicita que se envíe la indicación de uso relativa a la llamada que ha sido validada.
AccessTokens (Testigos de acceso)	Testigos recibidos del origen de la llamada para confirmar la autorización de acceso de esa llamada.

G.8.2.32 Confirmación de validación

Este mensaje indica que se ha validado la llamada. El elemento de frontera que efectuó la validación puede indicar alias para terminar la llamada.

Campo	Descripción
DestinationInfo (Información de destino)	Parámetros alternativos para el destino que utilizará el elemento de frontera destinatario.
UsageSpec (Especificación de uso)	Si está presente, indica el elemento de frontera que envía las peticiones de confirmación solicita que se envíe la indicación de uso relativa a la llamada que ha sido validada.

G.8.2.33 Rechazo de validación

Indica que la llamada no es válida. El elemento de frontera solicitante puede no completar la llamada.

Campo	Descripción
Reason (motivo)	Indica el motivo del rechazo de la petición. Puede ser: • testigo no válido (tokenNotValid) – El testigo de acceso suministrado no es válido para la llamada. • Seguridad (Security) – El mensaje ValidationRequest no cumplió los requisitos de seguridad del destinatario. • Cómputo de saltos rebasado (HopCountExceeded) – El cómputo de saltos llegó a cero y no hay información disponible. • Información del origen insuficiente (MissingSourceInfo) – La información sobre el origen suministrada no era suficiente para validar la llamada. • Información sobre el destino insuficiente (MissingDestInfo) – La información sobre el destino suministrada no era suficiente para validar la llamada. • relación de servicio inexistente (noServiceRelationship) – El destinatario intercambiará esta información únicamente después del establecimiento de una relación de servicio. • Indefinido (Undefined) – El motivo del rechazo de ValidationRequest no corresponde a las otras posibilidades.

G.9 Ejemplos de señalización

Estos ejemplos de señalización tienen por objeto ilustrar el funcionamiento básico. En estos ejemplos se supone que los dominios administrativos tienen acuerdos entre sí, de modo que los elementos de frontera disponen de información mutua (por ejemplo, puertos TCP). En muchos de los ejemplos que figuran a continuación, un guardián de puerta y un elemento de frontera intercambian los mensajes RAS LRK/LCF en el mismo dominio administrativo. Esto tiene fines puramente ilustrativos ya que no se ha determinado el protocolo del punto de referencia B (véase G.1).

G.9.1 Red distribuida o malla completa

En la figura G.7 se muestra un ejemplo de red distribuida.

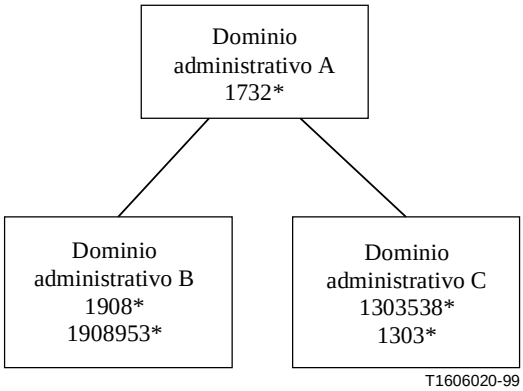


Figura G.7/H.225.0 – Red distribuida para ejemplos de señalización

Para este ejemplo, supóngase que cada dominio administrativo tiene un elemento de frontera, y que los elementos de frontera están configurados para resolver direcciones como sigue:

Dominio administrativo	Definición de plantilla	Comentarios
A	Descriptor "d1": Patrón = 1732* Dirección de transporte = Dirección de señal de llamada de BE_A Tipo de mensaje = sendSetup	La señalización para cualquier llamada al dominio administrativo A se hará a través del elemento de frontera de dicho dominio administrativo.
B	Descriptor "d1": Patrón = 1908* Dirección de transporte = Dirección de anexo g de BE_B Tipo de mensaje = sendAccessRequest Descriptor "d2": Patrón = 1908953* Dirección de transporte = Dirección de SEÑALIZACIÓN DE LLAMADA de GW_{B1} Tipo de mensaje = sendSetup	Para las llamadas a 1908*, se requiere un mensaje AccessRequest para obtener la dirección de señalización de llamada de destino (es decir, una pasarela). Para las llamadas a 1908953*, el mensaje Setup puede enviarse directamente a esta pasarela.
C	Descriptor "d1": Patrón = 1303538* Dirección de transporte = Dirección de señal de llamada de GK_{C1} Tipo de mensaje = sendSetup Descriptor "d2": Patrón = 1303* Dirección de transporte = Dirección de anexo g BE_C Tipo de mensaje = sendAccessRequest	Las llamadas a 1303538* se encaminarán a través de este guardián de puerta. Las llamadas a 1303* pueden señalizarse directamente a la pasarela de destino, pero debe enviarse un mensaje AccessRequest para obtener la dirección de señalización de llamada de la pasarela.

G.9.1.1 Intercambio de información de zona

En la organización distribuida (o malla completa) cada dominio administrativo conoce los demás dominios administrativos, probablemente a través de varios acuerdos contractuales bilaterales. En todo momento, un elemento de frontera de un dominio administrativo puede consultar a otro dominio administrativo para obtener información de direccionamiento. En la figura G.8 se ilustra un ejemplo de esta señalización.

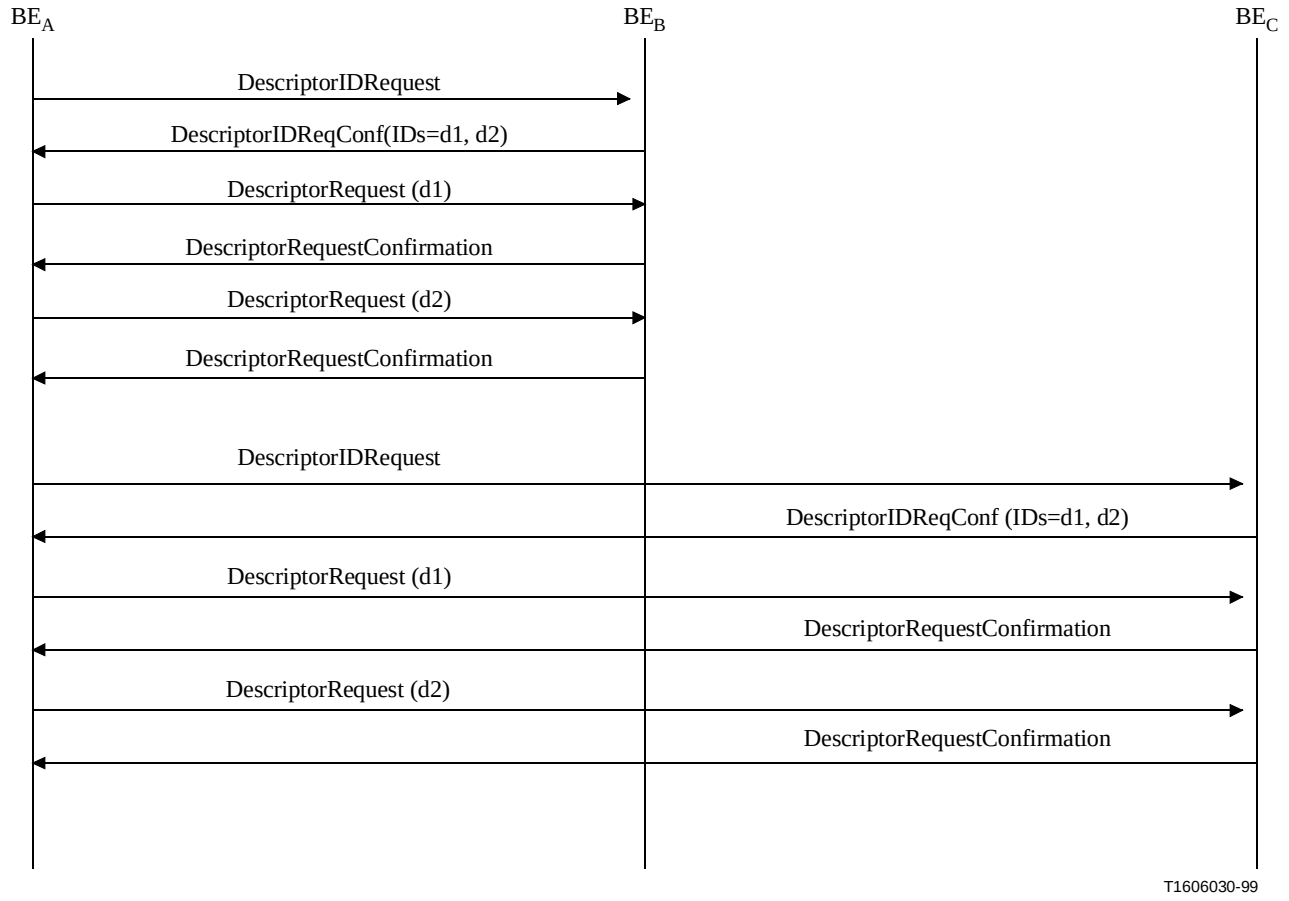


Figura G.8/H.225.0 – Ejemplo de intercambio de descriptor

De modo similar, BE_B consulta a BE_A y a BE_C, y BE_C consulta a BE_A y a BE_B.

G.9.1.2 Realización de una llamada

Supóngase que T1, en el dominio administrativo A, inicia una llamada a 19085551515 (T2). Al recibir la ARQ de T1, el guardián de puerta de T1 envía una LRQ. Un elemento de frontera del dominio administrativo A, BE_A, ha recibido previamente los descriptores de zona y sabe cómo cursar la petición. Como se muestra en la figura G.9, BE_A envía un mensaje AccessRequest a BE_B, tal como se especifica en el descriptor BE_A recibido de BE_B. BE_B responde con la dirección de señalización de llamada de T2 (en este ejemplo, T2 podría ser cualquier tipo de punto extremo). Enseguida, T1 envía el mensaje Setup H.225.0 a la dirección de señalización de llamada de T2 de conformidad con los procedimientos normales definidos en la Recomendación H.323 o en sus anexos.

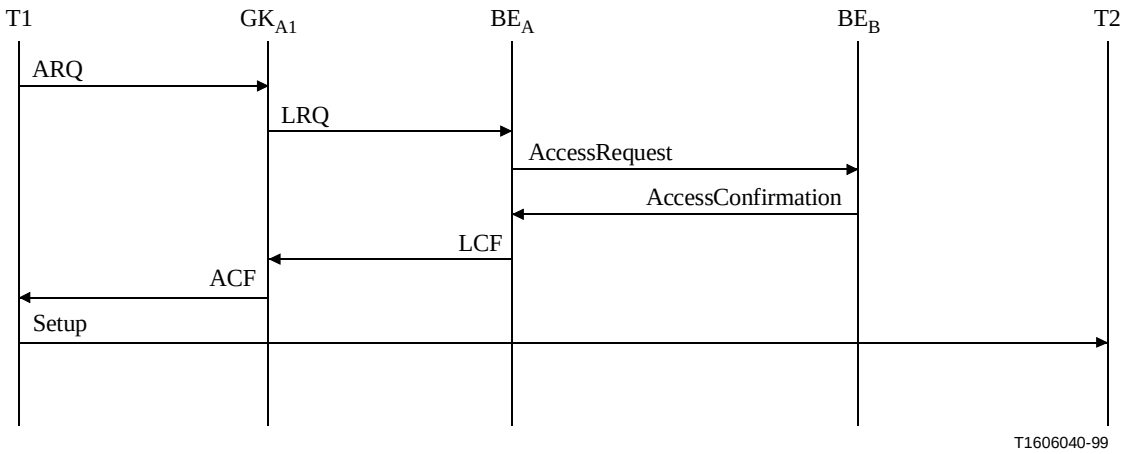


Figura G.9/H.225.0

Ahora, supóngase que T1 inicia una llamada a 19089532000. En este ejemplo, BE_A ha obtenido previamente la dirección de señalización de llamada de una pasarela del dominio administrativo que aceptará la llamada. Como se muestra en la figura G.10, BE_A puede responder a la LRQ sin ningún intercambio de mensajes hacia el dominio administrativo B, pudiendo T1 enviar el mensaje Setup directamente a la pasarela.

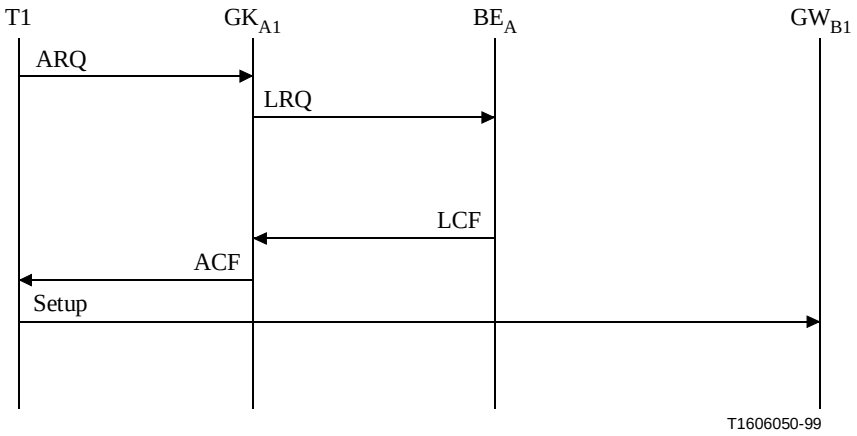


Figura G.10/H.225.0

En otro ejemplo, supóngase que T1 inicia una llamada a 13035382899. El dominio administrativo C ha señalado su posibilidad de aceptar una llamada a este número, y aceptará la señalización de llamada a través de su guardián de puerta implementando el modelo de encaminamiento por guardián de puerta. Como se muestra en la figura G.11, BE_A puede responder a la LRQ con una LCF que contiene la dirección de señalización de llamada de un guardián de puerta del dominio administrativo C sin ningún intercambio de mensajes hacia el dominio administrativo C.

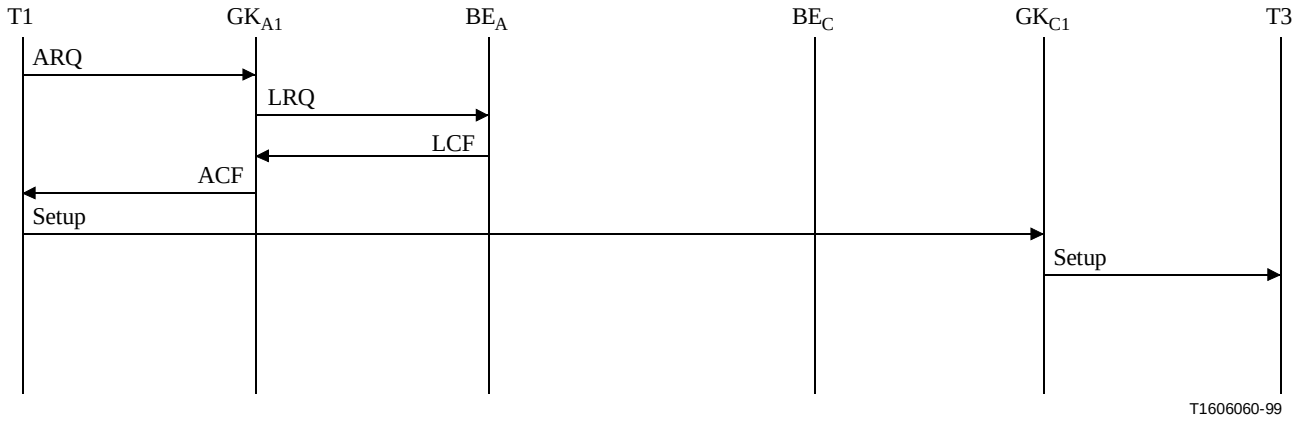


Figura G.11/H.225.0

Alternativamente, el guardián de puerta de T1 puede implementar el modelo de encaminamiento por guardián de puerta, tal como se muestra en la figura G.12.

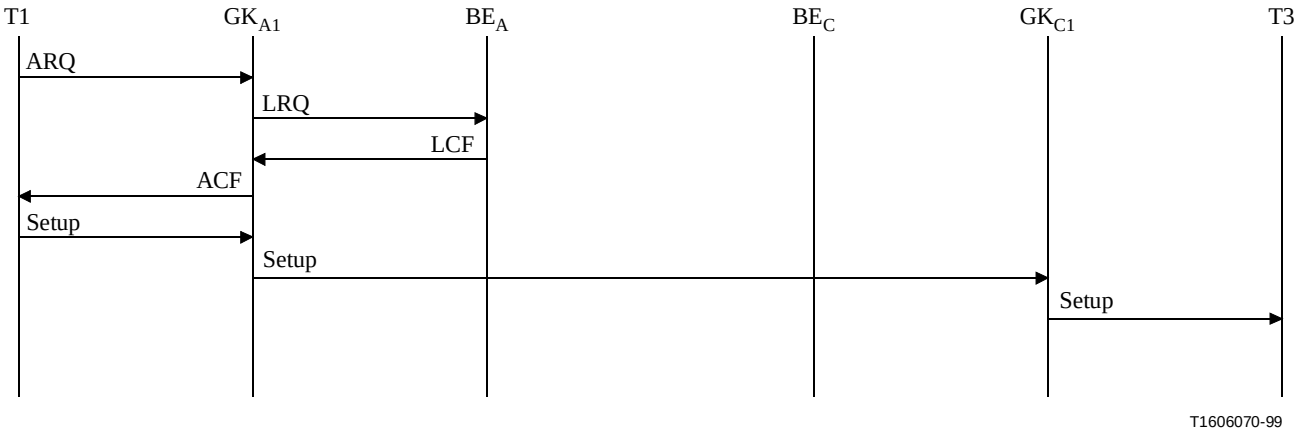


Figura G.12/H.225.0

G.9.2 Centro de resolución

En la figura G.13 se ilustra un ejemplo de configuración que utiliza un centro de resolución. Refiérase a esta figura para los ejemplos siguientes. En este ejemplo, el centro de resolución tiene la información de direccionamiento de todos los dominios administrativos para los cuales presta servicio.

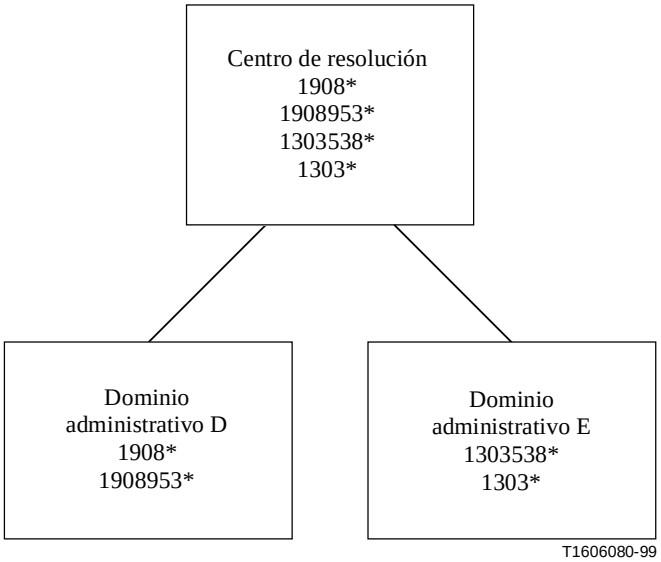


Figura G.13/H.225.0 – Ejemplo de configuración con centro de resolución

En este ejemplo, los elementos de frontera de los dominios administrativos D y E, así como el centro de resolución, contienen la información siguiente:

Dominio administrativo	**Definición de la plantilla**	**Comentarios**
D	Descriptor "d1": Patrón = 1908* Dirección de transporte = Dirección de anexo g de BE_D Tipo de mensaje = enviar AccessRequest Descriptor "d2": Patrón = 1908953* Dirección de transporte = Dirección de señalización de llamada de GW_{D1} Tipo de mensaje = enviar Setup	Para las llamadas a 1908* se necesita un mensaje AccessRequest para obtener la dirección de señalización de llamada de destino (es decir, una pasarela). Para las llamadas a 1908953*, el mensaje Setup puede enviarse directamente a esta pasarela.
E	Descriptor "d1": Patrón = 1303538* Dirección de transporte = Dirección de señalización de llamada de GK_{E1} Tipo de mensaje = enviar Setup Descriptor "d2": Patrón = 1303* Dirección de transporte = Dirección de anexo g de BE_E Tipo de mensaje = enviar AccessRequest	Las llamadas a 1303538* se encaminarán a través de este guardián de puerta. Las llamadas a 1303* pueden ser señalizadas directamente a la pasarela de destino, pero debe enviarse un mensaje AccessRequest para obtener la dirección de señalización de llamada de la pasarela.
CH (centro de resolución)	Descriptor "d1": Patrón = 1908* Dirección de transporte = Dirección anexo g de BE_D Tipo de mensaje = enviar AccessRequest Descriptor "d2": Patrón = 1908953* Dirección de transporte = Dirección de señalización de llamada de GW_{D1} Tipo de mensaje = enviar Setup	El centro de resolución obtiene descriptores de otros dominios administrativos y mantiene esta información para distribuirla durante el intercambio de descriptores.

Dominio administrativo	Definición de la plantilla	Comentarios
CH (centro de resolución) *(cont.)*	Descriptor "d3": Patrón = 1303538* Dirección de transporte = Dirección de señalización de llamada de GK_{E1} Tipo de mensaje = enviar Setup Descriptor "d4": Patrón = 1303* Dirección de transporte = Dirección anexo g de BE_E Tipo de mensaje = enviar AccessRequest	

G.9.2.1 Intercambio de zona de información

En este ejemplo, un centro de resolución intercambia información con dominios administrativos adscritos al servicio del centro de resolución. El centro de resolución conserva la información que recibe de cada dominio administrativo y la transmite a los otros dominios administrativos. En este ejemplo, el centro de resolución aparece como dominio administrativo E al dominio administrativo D, mientras que los dominios administrativos D y E no necesariamente tienen conocimiento el uno del otro. Véase la figura G.14.

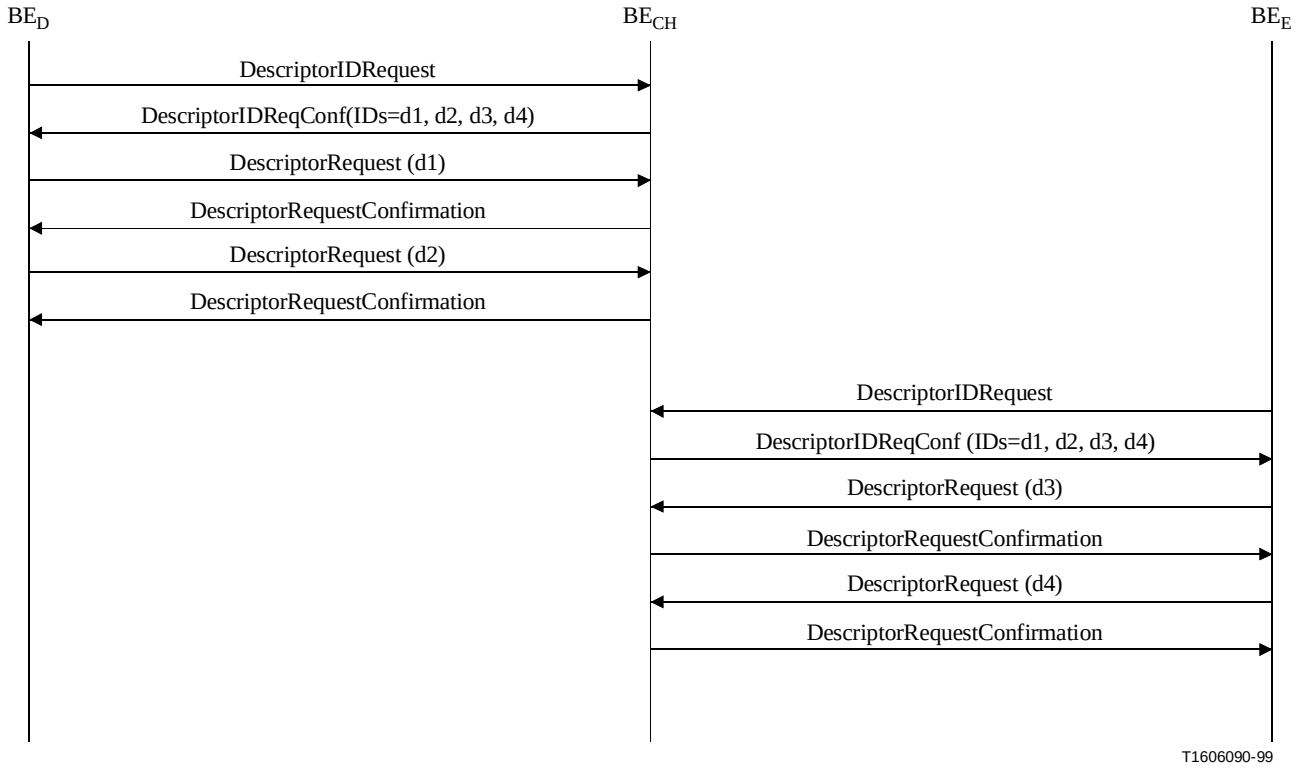


Figura G.14/H.225.0 – Ejemplo de intercambio de descriptores con centro de resolución

G.9.2.2 Realización de una llamada

Supóngase que T1, en el dominio administrativo E, inicia una llamada a 19085551515. El elemento de frontera del dominio administrativo E ha recibido del centro de resolución los descriptores que indican que debe consultarse al centro de resolución para dicha llamada. El elemento de frontera envía un mensaje AccessRequest al elemento de frontera centro de resolución. Basándose en los

descriptores que el elemento de frontera centro de resolución ha recibido del elemento de frontera del dominio administrativo D, el elemento de frontera centro de resolución envía un mensaje AccessRequest al elemento de frontera del dominio administrativo D. Cuando el elemento de frontera centro de resolución devuelve la confirmación al elemento de frontera del dominio administrativo E, la confirmación contiene la información enviada desde el elemento de frontera del dominio administrativo D. El guardián de puerta de T1 devuelve una ACF con la dirección de señalización de llamada de destino (destCallSignalAddress) de T2, pudiendo T1 enviar el mensaje Setup a T2. Véase la figura G.15.

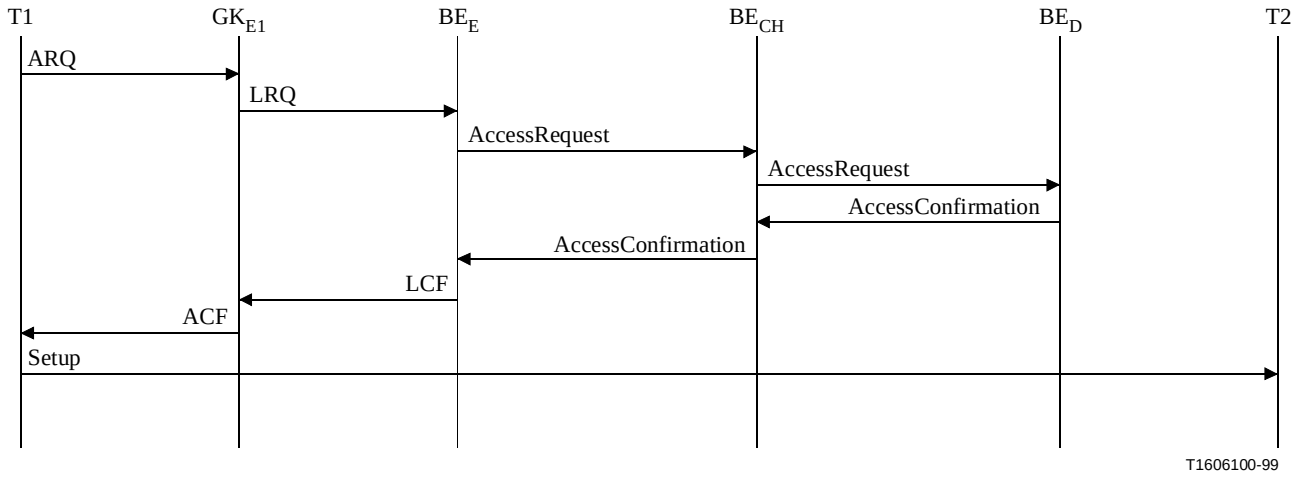


Figura G.15/H.225.0

Alternativamente, el guardián de puerta de T1 podría encaminar la señalización de llamada como se muestra en la figura G.16.

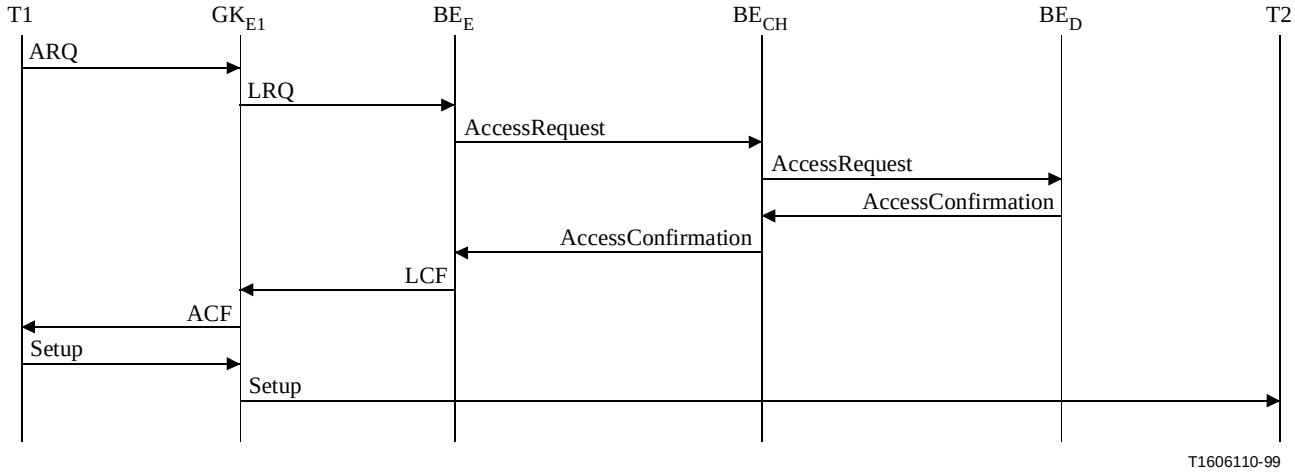


Figura G.16/H.225.0

Otra posibilidad consiste en que el centro de resolución responda al elemento de frontera del dominio administrativo E con la información de contacto del elemento de frontera del dominio administrativo D, como se muestra en la figura G.17.

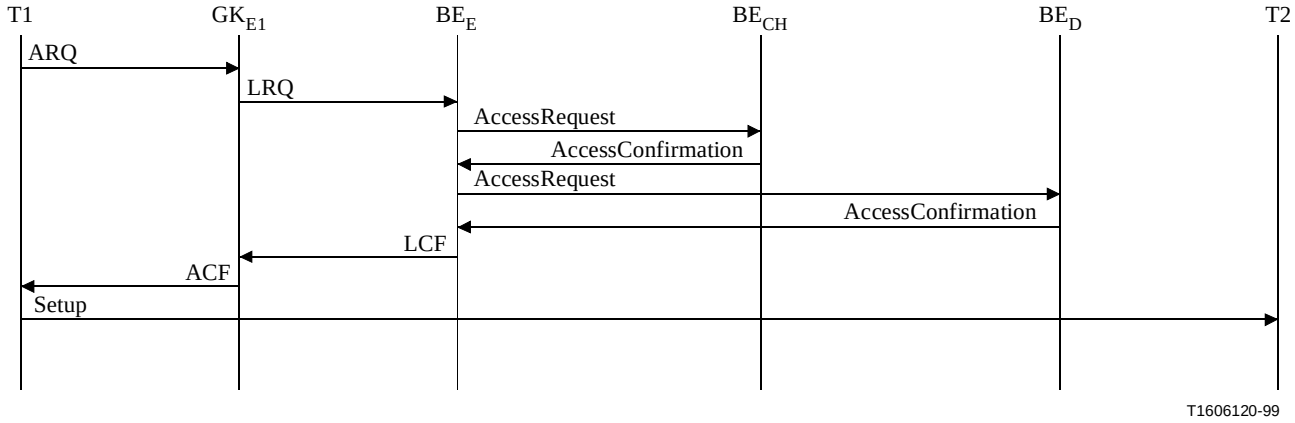


Figura G.17/H.225.0

Supóngase ahora que T1 inicia una llamada a 19089532000. Gracias a los descriptores intercambiados previamente, el elemento de frontera puede devolver la dirección de señalización de llamada a T1 sin consultar al centro de resolución, tal como se indica en la figura G.18.

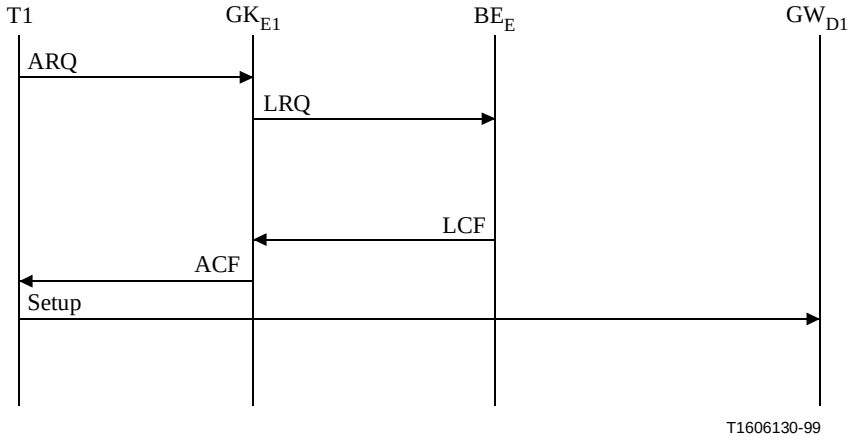


Figura G.18/H.225.0

A continuación, considérese una situación en la que T1 inicia una llamada a 13035382899. El elemento de frontera de un dominio administrativo E ha anunciado previamente que las llamadas a 1303538* pueden encaminarse directamente a un guardián de puerta del dominio administrativo E sin necesidad de un mensaje AccessRequest, tal como se indica en la figura G.19. (Este anuncio no indica que la entidad es un guardián de puerta, sino sólo que podría enviarse un mensaje Setup a una dirección especificada.) El elemento de frontera del dominio administrativo D ha recibido esta información del centro de resolución, suponiendo que el centro de resolución de este ejemplo no necesita proveer la resolución de dirección para estas llamadas.

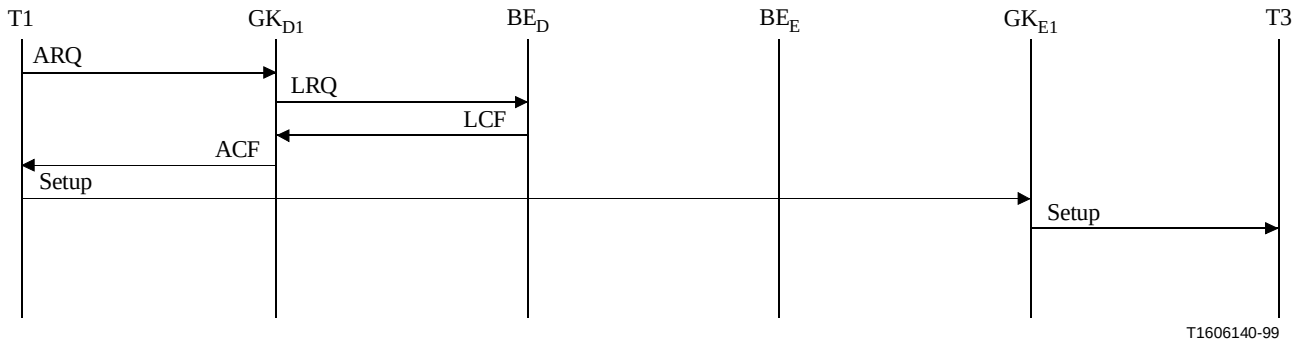


Figura G.19/H.225.0

Recuérdese que un elemento de frontera puede combinarse con un guardián de puerta, y puede también encaminar llamadas conforme al modelo de encaminamiento por guardián de puerta. En la figura G.20 se muestra un ejemplo de señalización alternativa. También es posible utilizar un elemento de frontera como guardián de puerta de encaminamiento hacia un dominio administrativo si los descriptores están configurados para ello.

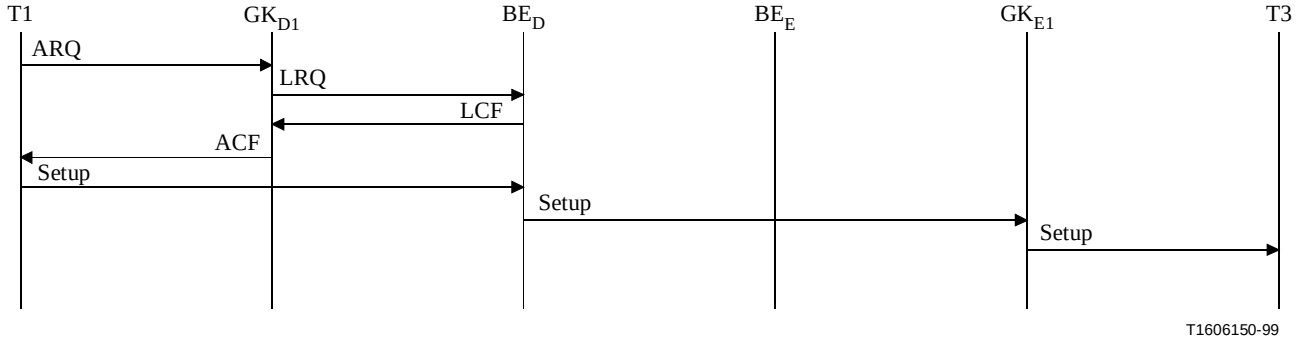


Figura G.20/H.225.0

En el ejemplo de la figura G.21, el centro de resolución valida la llamada para el dominio administrativo de terminación. Además, el centro de resolución solicita a los elementos de frontera de origen y de terminación que envíen indicaciones de uso para la llamada.

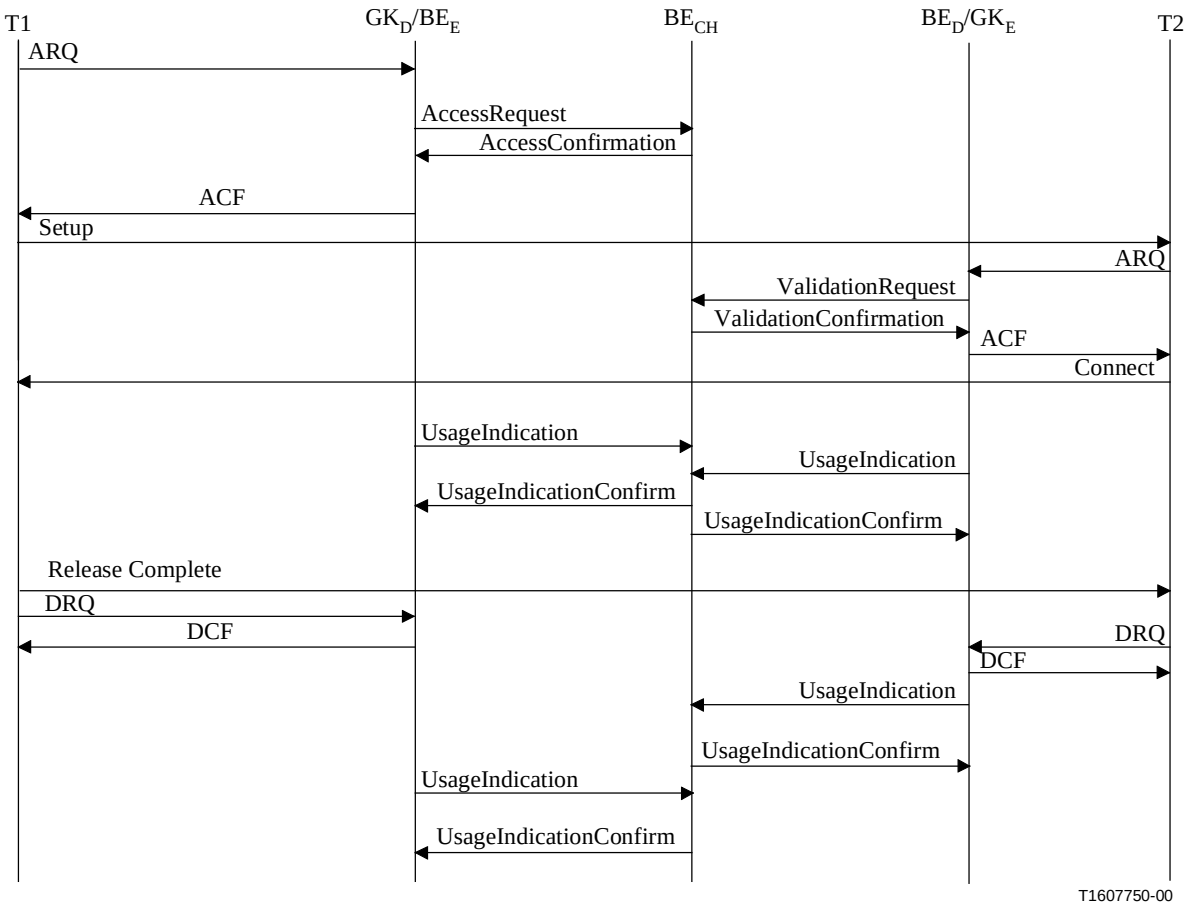


Figura G.21/H.225.0

Message Syntax

```
ANNEXG-MESSAGES DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
IMPORTS
      AuthenticationMechanism,
      TimeStamp,
      ClearToken
      FROM H235-SECURITY-MESSAGES

      AliasAddress,
      TransportAddress,
      ReleaseCompleteReason,
      ConferenceIdentifier,   CallIdentifier,   CryptoH323Token,  CryptoToken,

      EndpointType,
      GatekeeperIdentifier,
      GloballyUniqueID,
      NonStandardParameter,
      NumberDigits,
      PartyNumber,
      TransportQOS,
      VendorIdentifier,
      IntegrityMechanism,
      ICV
      FROM H323-MESSAGES;
```

```
Message ::= SEQUENCE
{
      body  AnnexGMessageBody,
      common      AnnexGCommonInfo,
      ...
}

AnnexGMessageBody ::= CHOICE
{
      serviceRequest                ServiceRequest,
      serviceConfirmation           ServiceConfirmation,
      serviceRejection              ServiceRejection,
      serviceRelease                ServiceRelease,
      descriptorRequest             DescriptorRequest,
      descriptorConfirmation        DescriptorConfirmation,
      descriptorRejection           DescriptorRejection,
      descriptorIDRequest           DescriptorIDRequest,
      descriptorIDConfirmation      DescriptorIDConfirmation,
      descriptorIDRejection         DescriptorIDRejection,
      descriptorUpdate              DescriptorUpdate,
      descriptorUpdateAck           DescriptorUpdateAck,
      accessRequest                 AccessRequest,
      accessConfirmation            AccessConfirmation,
      accessRejection               AccessRejection,
      requestInProgress             RequestInProgress,
      nonStandardRequest            NonStandardRequest,
      nonStandardConfirmation       NonStandardConfirmation,
      nonStandardRejection          NonStandardRejection,
      unknownMessageResponse        UnknownMessageResponse,
      usageRequest                  UsageRequest,
      usageConfirmation             UsageConfirmation,
      usageIndication               UsageIndication,
      usageIndicationConfirmation   UsageIndicationConfirmation,
      usageIndicationRejection      UsageIndicationRejection,
      usageRejection                UsageRejection,
      validationRequest             ValidationRequest,
      validationConfirmation        ValidationConfirmation,
      validationRejection           ValidationRejection,
      ...
}

AnnexGCommonInfo ::= SEQUENCE
{
      sequenceNumber          INTEGER (0..65535),
      version                 AnnexGVersion,
      hopCount                INTEGER (1..255),
      replyAddress            SEQUENCE OF TransportAddress OPTIONAL,
                              -- Must be present in request
      integrityCheckValue     ICV OPTIONAL,
      tokens                  SEQUENCE OF ClearToken OPTIONAL,
      cryptoTokens            SEQUENCE OF CryptoH323Token OPTIONAL,
      nonStandard             SEQUENCE OF NonStandardParameter OPTIONAL,
      ...
}



--
-- Annex G messages
--
```

```
ServiceRequest ::= SEQUENCE
{
      elementIdentifier       ElementIdentifier OPTIONAL,
      domainIdentifier  AliasAddress OPTIONAL,
      securityMode      SEQUENCE OF SecurityMode OPTIONAL,
      timeToLive        INTEGER (1..4294967295) OPTIONAL,
      ...
}

SecurityMode ::= SEQUENCE
{
      authentication    AuthenticationMechanism OPTIONAL,
      integrity         IntegrityMechanism OPTIONAL,
      algorithmOIDs     SEQUENCE OF OBJECT IDENTIFIER OPTIONAL,
      ...
}

ServiceConfirmation ::= SEQUENCE
{
      elementIdentifier ElementIdentifier,
      domainIdentifier  AliasAddress,
      alternates        AlternateBEInfo OPTIONAL,
      securityMode      SecurityMode OPTIONAL,
      timeToLive        INTEGER (1..4294967295) OPTIONAL,
      ...
}

ServiceRejection ::= SEQUENCE
{
      reason            ServiceRejectionReason,
      alternates        AlternateBEInfo OPTIONAL,
      ...
}

ServiceRejectionReason ::= CHOICE
{
      serviceUnavailable      NULL,
      serviceRedirected       NULL,
      security                NULL,
      continue                NULL,
      undefined               NULL,
      ...
}


ServiceRelease ::= SEQUENCE
{
      reason            ServiceReleaseReason,
      alternates        AlternateBEInfo OPTIONAL,
      ...
}

ServiceReleaseReason ::= CHOICE
{
      outOfService      NULL,
      maintenance       NULL,
      terminated        NULL,
      expired           NULL,
      ...
}
```

```
DescriptorRequest ::= SEQUENCE
{
      descriptorID      SEQUENCE OF DescriptorID,
      ...
}


DescriptorConfirmation ::= SEQUENCE
{
      descriptor        SEQUENCE OF Descriptor,
      ...
}


DescriptorRejection ::= SEQUENCE
{
      reason                  DescriptorRejectionReason,
      descriptorID            DescriptorID OPTIONAL,
      ...
}

DescriptorRejectionReason ::= CHOICE
{
      packetSizeExceeded      NULL, -- use other transport type
      illegalID               NULL, -- no descriptor for provided descriptorID
      security          NULL, -- request did not meet security requirements
      hopCountExceeded  NULL,
      noServiceRelationship NULL,
      undefined         NULL,
      ...
}




DescriptorIDRequest ::= SEQUENCE
{
      ...
}


DescriptorIDConfirmation ::= SEQUENCE
{
      descriptorInfo          SEQUENCE OF DescriptorInfo,
      ...
}


DescriptorIDRejection ::= SEQUENCE
{
      reason            DescriptorIDRejectionReason,
      ...
}

DescriptorIDRejectionReason ::= CHOICE
{
      noDescriptors           NULL, -- no descriptors to report
      security          NULL, -- request did not meet security requirements
      hopCountExceeded  NULL,
      noServiceRelationship   NULL,
      undefined         NULL,
      ...
}
```

```
DescriptorUpdate ::= SEQUENCE
{
      sender            AliasAddress,
      updateInfo        SEQUENCE OF UpdateInformation,
      ...
}

UpdateInformation ::=   SEQUENCE
{
      descriptorInfo    CHOICE      {
            descriptorID      DescriptorID,
            descriptor        Descriptor,
            ...
      },
      updateType  CHOICE
      {
            added             NULL,
            deleted           NULL,
            changed           NULL,
            ...
      },
      ...
}

DescriptorUpdateAck ::= SEQUENCE
{
      ...
}


AccessRequest ::= SEQUENCE
{
      destinationInfo   PartyInformation,
      sourceInfo        PartyInformation OPTIONAL,
      callInfo          CallInformation OPTIONAL,
      usageSpec         UsageSpecification OPTIONAL,  ...
}


AccessConfirmation ::= SEQUENCE
{
      templates         SEQUENCE OF AddressTemplate,
      partialResponse   BOOLEAN,
      ...
}


AccessRejection ::= SEQUENCE
{
      reason                  AccessRejectionReason,
      ...
}

AccessRejectionReason ::= CHOICE
{
      noMatch                 NULL, -- no template matched the destinationInfo
      packetSizeExceeded      NULL, -- use other transport type
      security          NULL, -- request did not meet security requirements
      hopCountExceeded  NULL,
      needCallInformation     NULL,       -- Call Information must be specified
      noServiceRelationship   NULL,
      undefined         NULL,
      ...
}
```

```
UsageRequest ::= SEQUENCE
{
      callInfo    CallInformation,
      usageSpec   UsageSpecification,
      ...
}


UsageConfirmation ::= SEQUENCE
{
      ...
}

UsageRejection ::= SEQUENCE
{
      reason                  UsageRejectReason,
      ...
}

UsageIndication ::= SEQUENCE
{
      callInfo                CallInformation,
      accessTokens            SEQUENCE OF AccessToken OPTIONAL,
      senderRole              Role,
      usageCallStatus         UsageCallStatus,
      srcInfo                 PartyInformation OPTIONAL,
      destAddress             PartyInformation,
      startTime               TimeStamp OPTIONAL,
      endTime                 TimeStamp OPTIONAL,
      terminationCause        TerminationCause OPTIONAL,
      usageFields             SEQUENCE OF UsageField,
      ...
}

UsageField ::= SEQUENCE
{
      id                      OBJECT IDENTIFIER,
      value                   OCTET STRING,
      ...
}



UsageRejectReason ::= CHOICE
{
      invalidCall             NULL,
      unavailable             NULL,
      security                NULL,
      noServiceRelationship   NULL,
      undefined               NULL,
      ...
}

UsageIndicationConfirmation ::= SEQUENCE
{
      ...
}

UsageIndicationRejection ::= SEQUENCE
{
      reason                  UsageIndicationRejectionReason,
      ...
}
```

```
UsageIndicationRejectionReason ::= CHOICE
{
      unknownCall       NULL,
      incomplete        NULL,
      security          NULL,
      noServiceRelationship   NULL,
      undefined         NULL,
      ...
}

ValidationRequest ::= SEQUENCE
{
      accessToken       SEQUENCE OF AccessToken OPTIONAL,
      destinationInfo   PartyInformation OPTIONAL,
      sourceInfo        PartyInformation OPTIONAL,
      callInfo          CallInformation,
      usageSpec         UsageSpecification OPTIONAL,
      ...
}


ValidationConfirmation ::= SEQUENCE
{
      destinationInfo   PartyInformation OPTIONAL,
      usageSpec         UsageSpecification OPTIONAL,
      ...
}


ValidationRejection ::= SEQUENCE
{
      reason                  ValidationRejectionReason,
      ...
}

ValidationRejectionReason ::= CHOICE
{
      tokenNotValid           NULL,
      security                NULL, -- request did not meet security requirements
      hopCountExceeded        NULL,
      missingSorceInfo        NULL,
      missingDestInfo         NULL,
      noServiceRelationship   NULL,
      undefined               NULL,
      ...
}

RequestInProgress ::= SEQUENCE
{
      delay       INTEGER (1..65535),
      ...
}



NonStandardRequest ::= SEQUENCE
{
      ...
}

NonStandardConfirmation ::= SEQUENCE
{
      ...
}
```

```
NonStandardRejection ::= SEQUENCE
{
      reason            NonStandardRejectionReason,
      ...
}

NonStandardRejectionReason ::= CHOICE
{
      notSupported            NULL,
      noServiceRelationship   NULL,
      undefined         NULL,
      ...
}



UnknownMessageResponse ::= SEQUENCE
{
      unknownMessage          OCTET STRING,
      reason                  UnknownMessageReason,
      ...
}

UnknownMessageReason ::= CHOICE
{
      notUnderstood           NULL,
      undefined         NULL,
      ...
}







--
-- structures common to multiple messages
--

AddressTemplate ::= SEQUENCE
{
      pattern           SEQUENCE OF Pattern,
      routeInfo         SEQUENCE OF RouteInformation,
      timeToLive        INTEGER (1..4294967295),
      ...
}

Pattern ::= CHOICE
{
      specific          AliasAddress,
      wildcard          AliasAddress,
      range       SEQUENCE          {
            startOfRange      PartyNumber,
            endOfRange  PartyNumber
      },
      ...
}
```

```
RouteInformation ::= SEQUENCE
{
      messageType CHOICE
      {
            sendAccessRequest NULL,
            sendSetup         NULL,
            nonExistent       NULL,
            ...
      },
      callSpecific      BOOLEAN,
      usageSpec   UsageSpecification OPTIONAL,
      priceInfo   SEQUENCE OF PriceInfoSpec OPTIONAL,
      contacts    SEQUENCE OF ContactInformation,
      type        EndpointType OPTIONAL,
                  -- must be present if messageType = sendSetup
      ...}
ContactInformation ::= SEQUENCE{    transportAddress  AliasAddress,     priority
            INTEGER (0..127), transportQoS      TransportQOS OPTIONAL,
      security          SEQUENCE OF SecurityMode OPTIONAL,
      accessTokens            SEQUENCE OF AccessToken OPTIONAL,
      ...
}


PriceInfoSpec ::= SEQUENCE
{
      currency                IA5String (SIZE(3)),          -- e.g. "USD"
      currencyScale                 INTEGER(-127..127),
      validFrom               GlobalTimeStamp OPTIONAL,
      validUntil              GlobalTimeStamp OPTIONAL,
      hoursFrom               IA5String (SIZE(6)) OPTIONAL, -- "HHMMSS" UTC
      hoursUntil              IA5String (SIZE(6)) OPTIONAL, -- "HHMMSS" UTC
      priceElement                  SEQUENCE OF PriceElement OPTIONAL,
      priceFormula                  IA5String (SIZE(1..2048)) OPTIONAL,
      ...
}

PriceElement ::= SEQUENCE
{
      amount                        INTEGER(0..4294967295), -- meter increment
      quantum                       INTEGER(0..4294967295), -- each or part
                                                            -- thereof
      units CHOICE
      {
            seconds                 NULL,
            packets                 NULL,
            bytes                   NULL,
            initial                 NULL,
            minimum                 NULL,
            maximum                 NULL,
            ...
      },
      ...
}



Descriptor ::= SEQUENCE
{
      descriptorInfo          DescriptorInfo,
      templates         SEQUENCE OF AddressTemplate,
      gatekeeperID            GatekeeperIdentifier OPTIONAL,
      ...
}
```

```
DescriptorInfo ::= SEQUENCE
{
      descriptorID            DescriptorID,
      lastChanged       GlobalTimeStamp,
      ...
}


AlternateBEInfo ::= SEQUENCE
{
      alternateBE       SEQUENCE OF AlternateBE,
      alternateIsPermanent    BOOLEAN,
      ...
}

AlternateBE ::= SEQUENCE
{
      contactAddress          AliasAddress,
      priority          INTEGER (1..127),
      elementIdentifier ElementIdentifier OPTIONAL,
      ...
}

AccessToken ::= CHOICE
{
      token       ClearToken,
      cryptoToken CryptoH323Token,
      ...
}

CallInformation ::= SEQUENCE
{
      callIdentifier          CallIdentifier,
      conferenceID            ConferenceIdentifier,
      ...
}


UsageCallStatus ::= CHOICE
{
      preConnect              NULL, -- Call has not started
      callInProgress                NULL, -- Call is in progress
      callEnded               NULL, -- Call ended
      ...
}

UserInformation ::= SEQUENCE
{
      userIdentifier          AliasAddress,
      userAuthenticator SEQUENCE OF CryptoH323Token OPTIONAL,
      ...
}


UsageSpecification ::= SEQUENCE
{
      sendTo                  ElementIdentifier,
      when SEQUENCE
      {
            never       NULL OPTIONAL,
            start       NULL OPTIONAL,
            end         NULL OPTIONAL,
```

```
            period            INTEGER(1..65535) OPTIONAL,   -- in seconds
            failures    NULL OPTIONAL,
            ...
      },
      required          SEQUENCE OF OBJECT IDENTIFIER,
      preferred         SEQUENCE OF OBJECT IDENTIFIER,
      ...
}


PartyInformation ::= SEQUENCE
{
      logicalAddresses  SEQUENCE OF AliasAddress,
      domainIdentifier  AliasAddress OPTIONAL,
      transportAddress  AliasAddress OPTIONAL,
      endpointType            EndpointType OPTIONAL,
      userInfo          UserInformation OPTIONAL,
      timeZone          TimeZone OPTIONAL,
      ...
}


Role ::= CHOICE
{
      originator        NULL,
      destination       NULL,
      nonStandardData   NonStandardParameter,
      ...
}

TimeZone ::=  INTEGER (-43200..43200)
                                        -- number of seconds relative to UTC
                                        -- including DST if appropriate


TerminationCause ::= SEQUENCE
{
      releaseCompleteReason         ReleaseCompleteReason,
      causeIE                       INTEGER (1..65535) OPTIONAL,
      nonStandardData               NonStandardParameter OPTIONAL,
      ...
}



AnnexGVersion           ::=   OBJECT IDENTIFIER
                        -- shall be set to
                        -- {itu-t (0) recommendation (0) h(8) h225.0(2250)
                        -- Annex (1) G (7) version (0) 1 (0)}
DescriptorID            ::=   GloballyUniqueID

ElementIdentifier ::=   BMPString (SIZE(1..128))

GlobalTimeStamp   ::=   IA5String (SIZE(14))    -- in the form YYYYMMDDHHmmSS
                                -- where YYYY = year, MM = month, DD = day,
                                -- HH = hour, mm = minute, SS = second
                                -- (for example, 19981219120000 for noon
                                -- 19 December 1998)




END   -- of ANNEXG-MESSAGES
```

ANEXO H

Sintaxis de mensajes H.225.0 (ASN.1)

Esta Recomendación define protocolos para RAS (básicamente, un protocolo de controlador de acceso) y señalización de llamada (básicamente, unidades de datos de protocolo que residen en un elemento de información de usuario a usuario). Estos protocolos se definen conjuntamente en el siguiente árbol ASN.1. Las definiciones semánticas para los mensajes y diversos elementos figuran en cláusulas anteriores.

```
H323-MESSAGES DEFINITIONS AUTOMATIC TAGS ::=
BEGIN


IMPORTS
               SIGNED{},
               ENCRYPTED{},
               HASHED{},
               ChallengeString,
               TimeStamp,
               RandomVal,
               Password,
               EncodedPwdCertToken,
               ClearToken,
               CryptoToken,
               AuthenticationMechanism
        FROM H235-SECURITY-MESSAGES
               DataProtocolCapability,
               T38FaxProfile
        FROM MULTIMEDIA-SYSTEM-CONTROL;

H323-UserInformation ::= SEQUENCE            -- root for all Q.931 related ASN.1
{
        h323-uu-pdu    H323-UU-PDU,
        user-data      SEQUENCE
        {
               protocol-discriminator        INTEGER(0..255),
               user-information              OCTET STRING (SIZE(1..131)),
               ...
        } OPTIONAL,
        ...
}

H323-UU-PDU ::= SEQUENCE
{
        h323-message-body   CHOICE
        {
               setup                 Setup-UUIE,
               callProceeding        CallProceeding-UUIE,
               connect               Connect-UUIE,
               alerting              Alerting-UUIE,
               information           Information-UUIE,
               releaseComplete       ReleaseComplete-UUIE,
               facility              Facility-UUIE,
               ...,
               progress              Progress-UUIE,
               empty                 NULL           -- used when a FACILITY message is sent,
                                                    -- but the Facility-UUIE is not to be invoked
                                                    -- (possible when transporting supplementary
                                                    -- services messages)
        },
        nonStandardData              NonStandardParameter OPTIONAL,
        ...,
        h4501SupplementaryService    SEQUENCE OF OCTET STRING OPTIONAL,
                                            -- each sequence of octet string is defined as one
                                            -- H4501SupplementaryService APDU as defined in
                                            -- Table 3/H.450.1

        h245Tunneling                BOOLEAN,
                                            -- if TRUE, tunneling of H.245 messages is enabled
        h245Control                  SEQUENCE OF OCTET STRING OPTIONAL,
                                            -- each octet string may contain exactly
                                            -- one H.245 PDU
        nonStandardControl           SEQUENCE OF NonStandardParameter OPTIONAL
}
```

```
Alerting-UUIE ::= SEQUENCE
{
        protocolIdentifier     ProtocolIdentifier,
        destinationInfo        EndpointType,
        h245Address            TransportAddress OPTIONAL,
        ...,
        callIdentifier         CallIdentifier,
        h245SecurityMode       H245Security OPTIONAL,
        tokens                 SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
        fastStart              SEQUENCE OF OCTET STRING OPTIONAL,
        multipleCalls          BOOLEAN,
        maintainConnection     BOOLEAN,
        alertingAddress        SEQUENCE OF AliasAddress OPTIONAL,
        presentationIndicator  PresentationIndicator,
        screeningIndicator     ScreeningIndicator
}

CallProceeding-UUIE ::= SEQUENCE
{
        protocolIdentifier     ProtocolIdentifier,
        destinationInfo        EndpointType,
        h245Address            TransportAddress OPTIONAL,
        ...,
        callIdentifier         CallIdentifier,
        h245SecurityMode       H245Security OPTIONAL,
        tokens                 SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
        fastStart              SEQUENCE OF OCTET STRING OPTIONAL,
        multipleCalls          BOOLEAN,
        maintainConnection     BOOLEAN
}

Connect-UUIE ::= SEQUENCE
{
        protocolIdentifier     ProtocolIdentifier,
        h245Address            TransportAddress OPTIONAL,
        destinationInfo        EndpointType,
        conferenceID           ConferenceIdentifier,
        ...,
        callIdentifier         CallIdentifier,
        h245SecurityMode       H245Security OPTIONAL,
        tokens                 SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
        fastStart              SEQUENCE OF OCTET STRING OPTIONAL,
        multipleCalls          BOOLEAN,
        maintainConnection     BOOLEAN,
        language               SEQUENCE OF IA5String(SIZE (1..32)) OPTIONAL,
                               -- RFC1766 language tag
        connectedAddress       SEQUENCE OF AliasAddress OPTIONAL,
        presentationIndicator  PresentationIndicator,
        screeningIndicator     ScreeningIndicator
}

Information-UUIE ::= SEQUENCE
{
        protocolIdentifier     ProtocolIdentifier,
        ...,
        callIdentifier         CallIdentifier,
        tokens                 SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
        fastStart              SEQUENCE OF OCTET STRING OPTIONAL
}

ReleaseComplete-UUIE ::= SEQUENCE
{
        protocolIdentifier     ProtocolIdentifier,
        reason                 ReleaseCompleteReason OPTIONAL,
        ...,
        callIdentifier         CallIdentifier,
        tokens                 SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
        busyAddress            SEQUENCE OF AliasAddress OPTIONAL,
        presentationIndicator  PresentationIndicator,
        screeningIndicator     ScreeningIndicator
}
```

```
ReleaseCompleteReason ::= CHOICE
{
        noBandwidth                     NULL,   -- bandwidth taken away or ARQ denied
        gatekeeperResources             NULL,   -- exhausted
        unreachableDestination          NULL,   -- no transport path to the destination
        destinationRejection            NULL,   -- rejected at destination
        invalidRevision                 NULL,
        noPermission                    NULL,   -- called party's gatekeeper rejects
        unreachableGatekeeper           NULL,   -- terminal cannot reach gatekeeper for ARQ
        gatewayResources                NULL,
        badFormatAddress                NULL,
        adaptiveBusy                    NULL,   -- call is dropping due to LAN crowding
        inConf                          NULL,   -- no address in AlternativeAddress
        undefinedReason                 NULL,
        ...,
        facilityCallDeflection          NULL,   -- call was deflected using a Facility message
        securityDenied                  NULL,   -- incompatible security settings
        calledPartyNotRegistered        NULL,   -- used by gatekeeper when endpoint has
                                                -- preGrantedARQ to bypass ARQ/ACF
        callerNotRegistered             NULL,   -- used by gatekeeper when endpoint has
                                                -- preGrantedARQ to bypass ARQ/ACF
        newConnectionNeeded             NULL,   -- indicates that the Setup was not accepted on this
                                                -- connection, but that the Setup may be accepted on
                                                -- a new connection
        nonStandardReason               NonStandardParameter,
        replaceWithConferenceInvite     ConferenceIdentifier    -- call dropped due to subsequent
                                                                -- invitation to a conference
                                                                -- (see 8.4.3.8/H.323)
}

Setup-UUIE ::= SEQUENCE
{
        protocolIdentifier              ProtocolIdentifier,
        h245Address                     TransportAddress OPTIONAL,
        sourceAddress                   SEQUENCE OF AliasAddress OPTIONAL,
        sourceInfo                      EndpointType,
        destinationAddress              SEQUENCE OF AliasAddress OPTIONAL,
        destCallSignalAddress           TransportAddress OPTIONAL,
        destExtraCallInfo               SEQUENCE OF AliasAddress OPTIONAL,          -- Note 1
        destExtraCRV                    SEQUENCE OF CallReferenceValue OPTIONAL,    -- Note 1
        activeMC                        BOOLEAN,
        conferenceID                    ConferenceIdentifier,
        conferenceGoal                  CHOICE
        {
                create                          NULL,
                join                            NULL,
                invite                          NULL,
                ...,
                capability-negotiation          NULL,
                callIndependentSupplementaryService          NULL
        },
        callServices                    QseriesOptions  OPTIONAL,
        callType                        CallType,
        ...,
        sourceCallSignalAddress         TransportAddress OPTIONAL,
        remoteExtensionAddress          AliasAddress OPTIONAL,
        callIdentifier                  CallIdentifier,
        h245SecurityCapability          SEQUENCE OF H245Security OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        fastStart                       SEQUENCE OF OCTET STRING OPTIONAL,
        mediaWaitForConnect             BOOLEAN,
        canOverlapSend                  BOOLEAN,
        endpointIdentifier              EndpointIdentifier OPTIONAL,
        multipleCalls                   BOOLEAN,
        maintainConnection              BOOLEAN,
        connectionParameters            SEQUENCE        -- additional gateway parameters
        {
                connectionType                  ScnConnectionType,
                numberOfScnConnections          INTEGER (0..65535),
                connectionAggregation           ScnConnectionAggregation,
                ...
        } OPTIONAL,
        language                        SEQUENCE OF IA5String(SIZE (1..32)) OPTIONAL,
                                        -- RFC1766 language tag
        presentationIndicator           PresentationIndicator,
        screeningIndicator              ScreeningIndicator
}
```

```
ScnConnectionType ::= CHOICE
{
        unknown         NULL,   -- should be seleceted when connection type is unknown
        bChannel        NULL,   -- each individual connection on the SCN is 64kbps.
                                        -- Note that where SCN delivers 56kbps usable data, the
                                        -- actual bandwidth allocated on SCN is still 64kbps.
        hybrid2x64      NULL,   -- each connection is a 128kbps hybrid call
        hybrid384       NULL,   -- each connection is an H0 (384kbps) hybrid call
        hybrid1536      NULL,   -- each connection is an H11 (1536kbps) hybrid call
        hybrid1920      NULL,   -- each connection is an H12 (1920kbps) hybrid call
        multirate       NULL,   -- bandwidth supplied by SCN using multirate.
                                        -- In this case, the information transfer rate octet in the
                                        -- bearer capability shall be set to multirate and the rate
                                        -- multiplier octet shall denote the number of B channels.
        ...
}

ScnConnectionAggregation ::= CHOICE
{
        auto            NULL,   -- aggregation mechanism is unknown
        none            NULL,   -- call produced using a single SCN connection
        h221            NULL,   -- use H.221 framing to aggregate the connections
        bonded-mode1    NULL,   -- use ISO/IEC 13871 bonding mode 1.
                                        -- Use bonded-mode1 to signal a bonded call if the precise
                                        -- bonding mode to be used is unknown.
        bonded-mode2    NULL,   -- use ISO/IEC 13871 bonding mode 2
        bonded-mode3    NULL,   -- use ISO/IEC 13871 bonding mode 3
        ...
}

PresentationIndicator ::= CHOICE
{
        presentationAllowed             NULL,
        presentationRestricted          NULL,
        addressNotAvailable             NULL,
        ...
}

ScreeningIndicator ::= ENUMERATED
{
        userProvidedNotScreened (0),
                -- number was provided by a remote user
                -- and has not been screened by a gatekeeper
        userProvidedVerifiedAndPassed (1),
                -- number was provided by a user
                -- equipment (or by a remote network), and has
                -- been screened by a gatekeeper
        userProvidedVerifiedAndFailed (2),
                -- not used, value reserved
        networkProvided (3),
                -- number was provided by a gatekeeper
        ...
}

Facility-UUIE ::= SEQUENCE
{
        protocolIdentifier              ProtocolIdentifier,
        alternativeAddress              TransportAddress OPTIONAL,
        alternativeAliasAddress         SEQUENCE OF AliasAddress OPTIONAL,
        conferenceID                    ConferenceIdentifier OPTIONAL,
        reason                          FacilityReason,
        ...,
        callIdentifier                  CallIdentifier,
        destExtraCallInfo               SEQUENCE OF AliasAddress OPTIONAL,
        remoteExtensionAddress          AliasAddress OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        conferences                     SEQUENCE OF ConferenceList OPTIONAL,
        h245Address                     TransportAddress OPTIONAL,
        fastStart                       SEQUENCE OF OCTET STRING OPTIONAL,
        multipleCalls                   BOOLEAN,
        maintainConnection              BOOLEAN
}
```

```
ConferenceList ::= SEQUENCE
{
        conferenceID                    ConferenceIdentifier OPTIONAL,
        conferenceAlias                 AliasAddress OPTIONAL,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...
}

FacilityReason ::= CHOICE
{
        routeCallToGatekeeper           NULL,           -- call must use gatekeeper model
                                                        -- gatekeeper is alternativeAddress
        callForwarded                   NULL,
        routeCallToMC                   NULL,
        undefinedReason                 NULL,
        ...,
        conferenceListChoice            NULL,
        startH245                       NULL,           -- recipient should connect to h245Address
        noH245                          NULL            -- endpoint does not support H.245
}

Progress-UUIE ::= SEQUENCE
{
        protocolIdentifier              ProtocolIdentifier,
        destinationInfo                 EndpointType,
        h245Address                     TransportAddress OPTIONAL,
        callIdentifier                  CallIdentifier,
        h245SecurityMode                H245Security OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        fastStart                       SEQUENCE OF OCTET STRING OPTIONAL,
        ...,
        multipleCalls                   BOOLEAN,
        maintainConnection              BOOLEAN
}

TransportAddress ::= CHOICE
{
        ipAddress       SEQUENCE
        {
                ip                      OCTET STRING (SIZE(4)),
                port                    INTEGER(0..65535)
        },
        ipSourceRoute           SEQUENCE
        {
                ip                      OCTET STRING (SIZE(4)),
                port                    INTEGER(0..65535),
                route                   SEQUENCE OF OCTET STRING(SIZE(4)),
                routing                 CHOICE
                {
                        strict  NULL,
                        loose   NULL,
                        ...
                },
                ...
        },
        ipxAddress      SEQUENCE
        {
                node            OCTET STRING (SIZE(6)),
                netnum          OCTET STRING (SIZE(4)),
                port            OCTET STRING (SIZE(2))
        },
        ip6Address      SEQUENCE
        {
                ip              OCTET STRING (SIZE(16)),
                port            INTEGER(0..65535),
                ...
        },
        netBios                 OCTET STRING (SIZE(16)),
        nsap                    OCTET STRING (SIZE(1..20)),
        nonStandardAddress      NonStandardParameter,
        ...
}
```

```
EndpointType ::= SEQUENCE
{
        nonStandardData         NonStandardParameter OPTIONAL,
        vendor                  VendorIdentifier OPTIONAL,
        gatekeeper              GatekeeperInfo OPTIONAL,
        gateway                 GatewayInfo OPTIONAL,
        mcu                     McuInfo OPTIONAL,       -- mc must be set as well
        terminal                TerminalInfo OPTIONAL,
        mc                      BOOLEAN,                -- shall not be set by itself
        undefinedNode           BOOLEAN,
        ...,
        set                     BIT STRING (SIZE(32)) OPTIONAL
                                        -- shall not be used with mc, gatekeeper
                                        -- code points for the various SET devices
                                        -- are defined in the respective SET Annexes
}

GatewayInfo ::= SEQUENCE
{
        protocol                        SEQUENCE OF SupportedProtocols OPTIONAL,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...
}

SupportedProtocols ::= CHOICE
{
        nonStandardData                 NonStandardParameter,
        h310                            H310Caps,
        h320                            H320Caps,
        h321                            H321Caps,
        h322                            H322Caps,
        h323                            H323Caps,
        h324                            H324Caps,
        voice                           VoiceCaps,
        t120-only                       T120OnlyCaps,
        ...,
        nonStandardProtocol             NonStandardProtocol,
        t38FaxAnnexbOnly                T38FaxAnnexbOnlyCaps
}

H310Caps ::= SEQUENCE
{
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        dataRatesSupported              SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes               SEQUENCE OF SupportedPrefix
}

H320Caps ::= SEQUENCE
{
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        dataRatesSupported              SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes               SEQUENCE OF SupportedPrefix
}

H321Caps ::= SEQUENCE
{
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        dataRatesSupported              SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes               SEQUENCE OF SupportedPrefix
}

H322Caps ::= SEQUENCE
{
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        dataRatesSupported              SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes               SEQUENCE OF SupportedPrefix
}

H323Caps ::= SEQUENCE
{
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        dataRatesSupported              SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes               SEQUENCE OF SupportedPrefix
}
```

```
H324Caps ::= SEQUENCE
{
        nonStandardData              NonStandardParameter OPTIONAL,
        ...,
        dataRatesSupported           SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes            SEQUENCE OF SupportedPrefix
}

VoiceCaps ::= SEQUENCE
{
        nonStandardData              NonStandardParameter OPTIONAL,
        ...,
        dataRatesSupported           SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes            SEQUENCE OF SupportedPrefix
}

T120OnlyCaps ::= SEQUENCE
{
        nonStandardData              NonStandardParameter OPTIONAL,
        ...,
        dataRatesSupported           SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes            SEQUENCE OF SupportedPrefix
}

NonStandardProtocol ::= SEQUENCE
{
        nonStandardData              NonStandardParameter OPTIONAL,
        dataRatesSupported           SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes            SEQUENCE OF SupportedPrefix,
        ...
}

T38FaxAnnexbOnlyCaps ::= SEQUENCE
{
        nonStandardData              NonStandardParameter OPTIONAL,
        dataRatesSupported           SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes            SEQUENCE OF SupportedPrefix,
        t38FaxProtocol               DataProtocolCapability,
        t38FaxProfile                T38FaxProfile,
        ...
}

McuInfo ::= SEQUENCE
{
        nonStandardData              NonStandardParameter OPTIONAL,
        ...
}

TerminalInfo ::= SEQUENCE
{
        nonStandardData              NonStandardParameter OPTIONAL,
        ...
}

GatekeeperInfo ::= SEQUENCE
{
        nonStandardData              NonStandardParameter OPTIONAL,
        ...
}

VendorIdentifier ::= SEQUENCE
{
        vendor                H221NonStandard,
        productId             OCTET STRING (SIZE(1..256)) OPTIONAL, -- per vendor
        versionId             OCTET STRING (SIZE(1..256)) OPTIONAL, -- per product
        ...
}

H221NonStandard ::= SEQUENCE
{       t35CountryCode        INTEGER(0..255),       -- country, as per T.35
        t35Extension          INTEGER(0..255),       -- assigned nationally
        manufacturerCode      INTEGER(0..65535),     -- assigned nationally
        ...
}

NonStandardParameter ::= SEQUENCE
{
        nonStandardIdentifier NonStandardIdentifier,
        data                  OCTET STRING
}
```

```
NonStandardIdentifier         ::= CHOICE
{
       object                 OBJECT IDENTIFIER,
       h221NonStandard        H221NonStandard,
       ...
}

AliasAddress ::= CHOICE
{
       e164                   IA5String (SIZE (1..128)) (FROM ("0123456789#*,")),
       h323-ID                BMPString (SIZE (1..256)),    -- Basic ISO/IEC 10646-1 (Unicode)
       ...,
       url-ID                 IA5String (SIZE(1..512)),     -- URL style address
       transportID            TransportAddress,
       email-ID               IA5String (SIZE(1..512)),     -- rfc822-compliant email address
       partyNumber            PartyNumber
}

PartyNumber ::= CHOICE
{
       publicNumber                  PublicPartyNumber,
                                            -- the numbering plan is according to
                                            -- Recommendations E.163 and E.164.
       dataPartyNumber               NumberDigits,
                                            -- not used, value reserved.
       telexPartyNumber              NumberDigits,
                                            -- not used, value reserved.
       privateNumber                 PrivatePartyNumber,
       nationalStandardPartyNumber   NumberDigits,
                                            -- not used, value reserved.
       ...
}

PublicPartyNumber ::= SEQUENCE
{
       publicTypeOfNumber     PublicTypeOfNumber,
       publicNumberDigits     NumberDigits
}

PrivatePartyNumber ::= SEQUENCE
{
       privateTypeOfNumber    PrivateTypeOfNumber,
       privateNumberDigits    NumberDigits
}

NumberDigits ::= IA5String (SIZE (1..128)) (FROM ("0123456789#*,"))

PublicTypeOfNumber ::= CHOICE
{
       unknown                NULL,
                                            -- if used number digits carry prefix indicating type
                                            -- of number according to national recommendations.
       internationalNumber    NULL,
       nationalNumber         NULL,
       networkSpecificNumber  NULL,
                                            -- not used, value reserved
       subscriberNumber       NULL,
       abbreviatedNumber      NULL,
                                            -- valid only for called party number at the outgoing
                                            -- access, network substitutes appropriate number.
...
}

PrivateTypeOfNumber ::= CHOICE
{
       unknown                NULL,
       level2RegionalNumber   NULL,
       level1RegionalNumber   NULL,
       pISNSpecificNumber     NULL,
       localNumber            NULL,
       abbreviatedNumber      NULL,
       ...
}
```

```
Endpoint ::= SEQUENCE
{
        nonStandardData               NonStandardParameter OPTIONAL,
        aliasAddress                  SEQUENCE OF AliasAddress OPTIONAL,
        callSignalAddress             SEQUENCE OF TransportAddress OPTIONAL,
        rasAddress                    SEQUENCE OF TransportAddress OPTIONAL,
        endpointType                  EndpointType OPTIONAL,
        tokens                        SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                  SEQUENCE OF CryptoH323Token OPTIONAL,
        priority                      INTEGER(0..127) OPTIONAL,
        remoteExtensionAddress        SEQUENCE OF AliasAddress OPTIONAL,
        destExtraCallInfo             SEQUENCE OF AliasAddress OPTIONAL,
        ...
}

AlternateGK ::= SEQUENCE
{
        rasAddress                    TransportAddress,
        gatekeeperIdentifier          GatekeeperIdentifier  OPTIONAL,
        needToRegister                BOOLEAN,
        priority                      INTEGER (0..127),
        ...
}

AltGKInfo ::= SEQUENCE
{
        alternateGatekeeper           SEQUENCE OF AlternateGK,
        altGKisPermanent              BOOLEAN,
        ...
}

SecurityServiceMode ::= CHOICE
{
        nonStandard           NonStandardParameter,
        none                  NULL,
        default               NULL,
        ...                           -- can be extended with other specific modes
}

SecurityCapabilities ::= SEQUENCE
{
        nonStandard                   NonStandardParameter OPTIONAL,
        encryption                    SecurityServiceMode,
        authenticaton                 SecurityServiceMode,
        integrity                     SecurityServiceMode,
        ...
}

H245Security ::= CHOICE
{
        nonStandard                   NonStandardParameter,
        noSecurity                    NULL,
        tls                           SecurityCapabilities,
        ipsec                         SecurityCapabilities,
        ...
}


QseriesOptions ::= SEQUENCE
{
        q932Full              BOOLEAN,       -- if true, indicates full support for Q.932
        q951Full              BOOLEAN,       -- if true, indicates full support for Q.951
        q952Full              BOOLEAN,       -- if true, indicates full support for Q.952
        q953Full              BOOLEAN,       -- if true, indicates full support for Q.953
        q955Full              BOOLEAN,       -- if true, indicates full support for Q.955
        q956Full              BOOLEAN,       -- if true, indicates full support for Q.956
        q957Full              BOOLEAN,       -- if true, indicates full support for Q.957
        q954Info              Q954Details,
        ...
}

Q954Details ::= SEQUENCE
{
        conferenceCalling             BOOLEAN,
        threePartyService             BOOLEAN,
        ...
}
```

```
GloballyUniqueID      ::=     OCTET STRING (SIZE(16))
ConferenceIdentifier  ::=     GloballyUniqueID
RequestSeqNum         ::=     INTEGER (1..65535)
GatekeeperIdentifier  ::=     BMPString (SIZE(1..128))
BandWidth             ::=     INTEGER (0.. 4294967295)      -- in 100s of bits
CallReferenceValue    ::=     INTEGER (0..65535)
EndpointIdentifier    ::=     BMPString (SIZE(1..128))
ProtocolIdentifier    ::=     OBJECT IDENTIFIER
                                      -- shall be set to
                                      -- {itu-t (0) recommendation (0) h (8) 2250 version (0) 3}
TimeToLive            ::=     INTEGER (1..4294967295)       --in seconds

CallIdentifier ::= SEQUENCE
{
       guid                   GloballyUniqueID,
       ...
}


EncryptIntAlg ::= CHOICE
{      -- core encryption algorithms for RAS message integrity
       nonStandard            NonStandardParameter,
       isoAlgorithm           OBJECT IDENTIFIER,     -- defined in ISO/IEC 9979
       ...
}
NonIsoIntegrityMechanism ::= CHOICE
{      -- HMAC mechanism used, no truncation, tagging may be necessary!
       hMAC-MD5               NULL,
       hMAC-iso10118-2-s      EncryptIntAlg, -- according to ISO/IEC 10118-2 using
                                             -- EncryptIntAlg as core block encryption algorithm
                                             -- (short MAC)
       hMAC-iso10118-2-l      EncryptIntAlg, -- according to ISO/IEC 10118-2 using
                                             -- EncryptIntAlg as core block encryption algorithm
                                             -- (long MAC)
       hMAC-iso10118-3        OBJECT IDENTIFIER,     -- according to ISO/IEC 10118-3 using
                                                     -- OID as hash function (OID is SHA-1,
                                                     -- RIPE-MD160,
                                                     -- RIPE-MD128)
       ...
}

IntegrityMechanism ::= CHOICE
{      -- for RAS message integrity
       nonStandard            NonStandardParameter,
       digSig                 NULL,                  -- indicates to apply a digital signature
       iso9797                OBJECT IDENTIFIER,     -- according to ISO/IEC 9797 using OID as
                                                     -- core encryption algorithm (X-CBC MAC)
       nonIsoIM               NonIsoIntegrityMechanism,
       ...
}

ICV ::= SEQUENCE
{
       algorithmOID   OBJECT IDENTIFIER,     -- the algorithm used to compute the signature
       icv            BIT STRING             -- the computed cryptographic integrity check value
              -- or signature
}

FastStartToken ::= ClearToken (WITH COMPONENTS {..., timeStamp PRESENT, dhkey PRESENT, generalID
PRESENT -- set to 'alias' -- })
EncodedFastStartToken ::= TYPE-IDENTIFIER.&Type (FastStartToken)
CryptoH323Token::= CHOICE
{
       cryptoEPPwdHash SEQUENCE
       {
              alias          AliasAddress,  -- alias of entity generating hash
              timeStamp      TimeStamp,             -- timestamp used in hash
              token          HASHED  { EncodedPwdCertToken -- generalID set to 'alias' -- }
       },
       cryptoGKPwdHash  SEQUENCE
       {
              gatekeeperId   GatekeeperIdentifier,  -- GatekeeperID of GK generating hash
              timeStamp      TimeStamp,             -- timestamp used in hash
              token          HASHED  { EncodedPwdCertToken  -- generalID set to Gatekeeperid -- }
       },
       cryptoEPPwdEncr        ENCRYPTED { EncodedPwdCertToken -- generalID set to Gatekeeperid --},
       cryptoGKPwdEncr        ENCRYPTED { EncodedPwdCertToken -- generalID set to Gatekeeperid --},
       cryptoEPCert           SIGNED { EncodedPwdCertToken -- generalID set to Gatekeeperid -- },
```

```
        cryptoGKCert            SIGNED { EncodedPwdCertToken -- generalID set to alias -- },
        cryptoFastStart         SIGNED { EncodedFastStartToken },
        nestedcryptoToken       CryptoToken,
        ...
}


DataRate ::= SEQUENCE
{
        nonStandardData                 NonStandardParameter OPTIONAL,
        channelRate                     BandWidth,
        channelMultiplier               INTEGER (1..256) OPTIONAL,
        ...
}

SupportedPrefix ::= SEQUENCE
{
        nonStandardData                 NonStandardParameter OPTIONAL,
        prefix                          AliasAddress,
        ...
}


RasMessage ::= CHOICE
{
        gatekeeperRequest               GatekeeperRequest,
        gatekeeperConfirm               GatekeeperConfirm,
        gatekeeperReject                GatekeeperReject,
        registrationRequest             RegistrationRequest,
        registrationConfirm             RegistrationConfirm,
        registrationReject              RegistrationReject,
        unregistrationRequest           UnregistrationRequest,
        unregistrationConfirm           UnregistrationConfirm,
        unregistrationReject            UnregistrationReject,
        admissionRequest                AdmissionRequest,
        admissionConfirm                AdmissionConfirm,
        admissionReject                 AdmissionReject,
        bandwidthRequest                BandwidthRequest,
        bandwidthConfirm                BandwidthConfirm,
        bandwidthReject                 BandwidthReject,
        disengageRequest                DisengageRequest,
        disengageConfirm                DisengageConfirm,
        disengageReject                 DisengageReject,
        locationRequest                 LocationRequest,
        locationConfirm                 LocationConfirm,
        locationReject                  LocationReject,
        infoRequest                     InfoRequest,
        infoRequestResponse             InfoRequestResponse,
        nonStandardMessage              NonStandardMessage,
        unknownMessageResponse          UnknownMessageResponse,
        ...,
        requestInProgress               RequestInProgress,
        resourcesAvailableIndicate      ResourcesAvailableIndicate,
        resourcesAvailableConfirm       ResourcesAvailableConfirm,
        infoRequestAck                  InfoRequestAck,
        infoRequestNak                  InfoRequestNak
}

GatekeeperRequest ::= SEQUENCE --(GRQ)
{

        requestSeqNum                   RequestSeqNum,
        protocolIdentifier              ProtocolIdentifier,
        nonStandardData                 NonStandardParameter OPTIONAL,
        rasAddress                      TransportAddress,
        endpointType                    EndpointType,
        gatekeeperIdentifier            GatekeeperIdentifier OPTIONAL,
        callServices                    QseriesOptions  OPTIONAL,
        endpointAlias                   SEQUENCE OF AliasAddress OPTIONAL,
        ...,
        alternateEndpoints              SEQUENCE OF Endpoint OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        authenticationCapability        SEQUENCE OF AuthenticationMechanism OPTIONAL,
        algorithmOIDs                   SEQUENCE OF OBJECT IDENTIFIER OPTIONAL,
        integrity                       SEQUENCE OF IntegrityMechanism OPTIONAL,
        integrityCheckValue             ICV OPTIONAL
}
```

```
GatekeeperConfirm ::= SEQUENCE --(GCF)
{
        requestSeqNum                   RequestSeqNum,
        protocolIdentifier              ProtocolIdentifier,
        nonStandardData                 NonStandardParameter OPTIONAL,
        gatekeeperIdentifier            GatekeeperIdentifier  OPTIONAL,
        rasAddress                      TransportAddress,
        ...,
        alternateGatekeeper             SEQUENCE OF AlternateGK OPTIONAL,
        authenticationMode              AuthenticationMechanism OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        algorithmOID                    OBJECT IDENTIFIER OPTIONAL,
        integrity                       SEQUENCE OF IntegrityMechanism OPTIONAL,
        integrityCheckValue             ICV OPTIONAL
}

GatekeeperReject ::= SEQUENCE --(GRJ)
{
        requestSeqNum                   RequestSeqNum,
        protocolIdentifier              ProtocolIdentifier,
        nonStandardData                 NonStandardParameter OPTIONAL,
        gatekeeperIdentifier            GatekeeperIdentifier  OPTIONAL,
        rejectReason                    GatekeeperRejectReason,
        ...,
        altGKInfo                       AltGKInfo OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL
}

GatekeeperRejectReason ::= CHOICE
{
        resourceUnavailable             NULL,
        terminalExcluded                NULL,   -- permission failure, not a resource failure
        invalidRevision                 NULL,
        undefinedReason                 NULL,
        ...,
        securityDenial                  NULL
}

RegistrationRequest ::= SEQUENCE --(RRQ)
{
        requestSeqNum                   RequestSeqNum,
        protocolIdentifier              ProtocolIdentifier,
        nonStandardData                 NonStandardParameter OPTIONAL,
        discoveryComplete               BOOLEAN,
        callSignalAddress               SEQUENCE OF TransportAddress,
        rasAddress                      SEQUENCE OF TransportAddress,
        terminalType                    EndpointType,
        terminalAlias                   SEQUENCE OF AliasAddress OPTIONAL,
        gatekeeperIdentifier            GatekeeperIdentifier  OPTIONAL,
        endpointVendor                  VendorIdentifier,
        ...,
        alternateEndpoints              SEQUENCE OF Endpoint OPTIONAL,
        timeToLive                      TimeToLive OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL,
        keepAlive                       BOOLEAN,
        endpointIdentifier              EndpointIdentifier OPTIONAL,
        willSupplyUUIEs                 BOOLEAN,
        maintainConnection              BOOLEAN,
        supportsAnnexECallSignalling    BOOLEAN
}

RegistrationConfirm ::= SEQUENCE --(RCF)
{
        requestSeqNum                   RequestSeqNum,
        protocolIdentifier              ProtocolIdentifier,
        nonStandardData                 NonStandardParameter OPTIONAL,
        callSignalAddress               SEQUENCE OF TransportAddress,
        terminalAlias                   SEQUENCE OF AliasAddress OPTIONAL,
        gatekeeperIdentifier            GatekeeperIdentifier  OPTIONAL,
        endpointIdentifier              EndpointIdentifier,
        ...,
        alternateGatekeeper             SEQUENCE OF AlternateGK OPTIONAL,
        timeToLive                      TimeToLive OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
```

```
        integrityCheckValue           ICV OPTIONAL,
        willRespondToIRR              BOOLEAN,
        preGrantedARQ                 SEQUENCE
        {
                makeCall                              BOOLEAN,
                useGKCallSignalAddressToMakeCall      BOOLEAN,
                answerCall                            BOOLEAN,
                useGKCallSignalAddressToAnswer        BOOLEAN,
                ...,
                irrFrequencyInCall                    INTEGER (1..65535) OPTIONAL,  -- in seconds; not
                                                                        -- present if GK
                                                                        -- does not want IRRs
                totalBandwidthRestriction             BandWidth OPTIONAL,    -- total limit for all
                                                                        -- concurrent calls
                useAnnexECallSignalling               BOOLEAN
        } OPTIONAL,
        maintainConnection            BOOLEAN
}

RegistrationReject ::= SEQUENCE --(RRJ)
{
        requestSeqNum                 RequestSeqNum,
        protocolIdentifier            ProtocolIdentifier,
        nonStandardData               NonStandardParameter OPTIONAL,
        rejectReason                  RegistrationRejectReason,
        gatekeeperIdentifier          GatekeeperIdentifier  OPTIONAL,
        ...,
        altGKInfo                     AltGKInfo OPTIONAL,
        tokens                        SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                  SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue           ICV OPTIONAL
}

RegistrationRejectReason ::= CHOICE
{
        discoveryRequired             NULL,
        invalidRevision               NULL,
        invalidCallSignalAddress      NULL,
        invalidRASAddress             NULL,   -- supplied address is invalid
        duplicateAlias                SEQUENCE OF AliasAddress,
                                              -- alias registered to another endpoint
        invalidTerminalType           NULL,
        undefinedReason               NULL,
        transportNotSupported         NULL,   -- one or more of the transports
        ...,
        transportQOSNotSupported      NULL,   -- endpoint QOS not supported
        resourceUnavailable           NULL,   -- gatekeeper resources exhausted
        invalidAlias                  NULL,   -- alias not consistent with gatekeeper rules
        securityDenial                NULL,
        fullRegistrationRequired      NULL    -- registration permission has expired
}

UnregistrationRequest ::= SEQUENCE --(URQ)
{
        requestSeqNum                 RequestSeqNum,
        callSignalAddress             SEQUENCE OF TransportAddress,
        endpointAlias                 SEQUENCE OF AliasAddress OPTIONAL,
        nonStandardData               NonStandardParameter OPTIONAL,
        endpointIdentifier            EndpointIdentifier OPTIONAL,
        ...,
        alternateEndpoints            SEQUENCE OF Endpoint OPTIONAL,
        gatekeeperIdentifier          GatekeeperIdentifier OPTIONAL,
        tokens                        SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                  SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue           ICV OPTIONAL,
        reason                        UnregRequestReason OPTIONAL
}

UnregRequestReason ::= CHOICE
{
        reregistrationRequired        NULL,
        ttlExpired                    NULL,
        securityDenial                NULL,
        undefinedReason               NULL,
        ...
}
```

```
UnregistrationConfirm ::= SEQUENCE --(UCF)
{
        requestSeqNum                RequestSeqNum,
        nonStandardData              NonStandardParameter OPTIONAL,
        ...,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL
}

UnregistrationReject ::= SEQUENCE --(URJ)
{

        requestSeqNum                RequestSeqNum,
        rejectReason                 UnregRejectReason,
        nonStandardData              NonStandardParameter OPTIONAL,
        ...,
        altGKInfo                    AltGKInfo OPTIONAL,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL
}

UnregRejectReason ::= CHOICE
{
        notCurrentlyRegistered       NULL,
        callInProgress               NULL,
        undefinedReason              NULL,
        ...,
        permissionDenied             NULL,   -- requesting user not allowed to unregister
                                             -- specified user
        securityDenial               NULL
}

AdmissionRequest ::= SEQUENCE --(ARQ)
{
        requestSeqNum                RequestSeqNum,
        callType                     CallType,
        callModel                    CallModel OPTIONAL,
        endpointIdentifier           EndpointIdentifier,
        destinationInfo              SEQUENCE OF AliasAddress OPTIONAL,    -- Note 1
        destCallSignalAddress        TransportAddress OPTIONAL,            -- Note 1
        destExtraCallInfo            SEQUENCE OF AliasAddress OPTIONAL,
        srcInfo                      SEQUENCE OF AliasAddress,
        srcCallSignalAddress         TransportAddress OPTIONAL,
        bandWidth                    BandWidth,
        callReferenceValue           CallReferenceValue,
        nonStandardData              NonStandardParameter OPTIONAL,
        callServices                 QseriesOptions  OPTIONAL,
        conferenceID                 ConferenceIdentifier,
        activeMC                     BOOLEAN,
        answerCall                   BOOLEAN,       -- answering a call
        ...,
        canMapAlias                  BOOLEAN,       -- can handle alias address
        callIdentifier               CallIdentifier,
        srcAlternatives              SEQUENCE OF Endpoint OPTIONAL,
        destAlternatives             SEQUENCE OF Endpoint OPTIONAL,
        gatekeeperIdentifier         GatekeeperIdentifier OPTIONAL,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL,
        transportQOS                 TransportQOS OPTIONAL,
        willSupplyUUIEs              BOOLEAN
}

CallType ::= CHOICE
{
        pointToPoint          NULL,          -- Point-to-point
        oneToN                NULL,          -- no interaction (FFS)
        nToOne                NULL,          -- no interaction (FFS)
        nToN                  NULL,          -- interactive (multipoint)
        ...
}

CallModel ::= CHOICE
{
        direct                NULL,
        gatekeeperRouted      NULL,
        ...
}
```

```
TransportQOS ::= CHOICE
{
        endpointControlled      NULL,
        gatekeeperControlled    NULL,
        noControl               NULL,
        ...
}

AdmissionConfirm ::= SEQUENCE --(ACF)
{
        requestSeqNum                   RequestSeqNum,
        bandWidth                       BandWidth,
        callModel                       CallModel,
        destCallSignalAddress           TransportAddress,
        irrFrequency                    INTEGER (1..65535) OPTIONAL,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        destinationInfo                 SEQUENCE OF AliasAddress OPTIONAL,
        destExtraCallInfo               SEQUENCE OF AliasAddress OPTIONAL,
        destinationType                 EndpointType OPTIONAL,
        remoteExtensionAddress          SEQUENCE OF AliasAddress OPTIONAL,
        alternateEndpoints              SEQUENCE OF Endpoint OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL,
        transportQOS                    TransportQOS OPTIONAL,
        willRespondToIRR                BOOLEAN,
        uuiesRequested                  UUIEsRequested,
        language                        SEQUENCE OF IA5String(SIZE (1..32)) OPTIONAL, -- RFC1766
language tag
        useAnnexECallSignalling         BOOLEAN
}

UUIEsRequested ::= SEQUENCE
{
        setup                           BOOLEAN,
        callProceeding                  BOOLEAN,
        connect                         BOOLEAN,
        alerting                        BOOLEAN,
        information                     BOOLEAN,
        releaseComplete                 BOOLEAN,
        facility                        BOOLEAN,
        progress                        BOOLEAN,
        empty                           BOOLEAN,
        ...
}

AdmissionReject ::= SEQUENCE --(ARJ)
{
        requestSeqNum           RequestSeqNum,
        rejectReason            AdmissionRejectReason,
        nonStandardData         NonStandardParameter OPTIONAL,
        ...,
        altGKInfo               AltGKInfo OPTIONAL,
        tokens                  SEQUENCE OF ClearToken OPTIONAL,
        callSignalAddress       SEQUENCE OF TransportAddress OPTIONAL,
        cryptoTokens            SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue     ICV OPTIONAL
}

AdmissionRejectReason ::= CHOICE
{
        calledPartyNotRegistered        NULL,   -- cannot translate address
        invalidPermission               NULL,   -- permission has expired
        requestDenied                   NULL,   -- no bandwidth available
        undefinedReason                 NULL,
        callerNotRegistered             NULL,
        routeCallToGatekeeper           NULL,
        invalidEndpointIdentifier       NULL,
        resourceUnavailable             NULL,
        ...,
        securityDenial                  NULL,
        qosControlNotSupported          NULL,
        incompleteAddress               NULL,
        routeCallToSCN                  SEQUENCE OF PartyNumber,
        aliasesInconsistent             NULL    -- multiple aliases in request identify distinct people
}
```

```
BandwidthRequest ::= SEQUENCE --(BRQ)
{
        requestSeqNum                   RequestSeqNum,
        endpointIdentifier              EndpointIdentifier,
        conferenceID                    ConferenceIdentifier,
        callReferenceValue              CallReferenceValue,
        callType                        CallType OPTIONAL,
        bandWidth                       BandWidth,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        callIdentifier                  CallIdentifier,
        gatekeeperIdentifier            GatekeeperIdentifier OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL,
        answeredCall                    BOOLEAN
}

BandwidthConfirm ::= SEQUENCE --(BCF)
{
        requestSeqNum                   RequestSeqNum,
        bandWidth                       BandWidth,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL
}

BandwidthReject ::= SEQUENCE --(BRJ)
{
        requestSeqNum                   RequestSeqNum,
        rejectReason                    BandRejectReason,
        allowedBandWidth                BandWidth,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        altGKInfo                       AltGKInfo OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL
}

BandRejectReason ::= CHOICE
{
        notBound                        NULL,   -- discovery permission has aged
        invalidConferenceID             NULL,   -- possible revision
        invalidPermission               NULL,   -- true permission violation
        insufficientResources           NULL,
        invalidRevision                 NULL,
        undefinedReason                 NULL,
        ...,
        securityDenial                  NULL
}

LocationRequest ::= SEQUENCE --(LRQ)
{
        requestSeqNum                   RequestSeqNum,
        endpointIdentifier              EndpointIdentifier OPTIONAL,
        destinationInfo                 SEQUENCE OF AliasAddress,
        nonStandardData                 NonStandardParameter OPTIONAL,
        replyAddress                    TransportAddress,
        ...,
        sourceInfo                      SEQUENCE OF AliasAddress OPTIONAL,
        canMapAlias                     BOOLEAN,        -- can handle alias address
        gatekeeperIdentifier            GatekeeperIdentifier OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL
}

LocationConfirm ::= SEQUENCE --(LCF)
{
        requestSeqNum                   RequestSeqNum,
        callSignalAddress               TransportAddress,
        rasAddress                      TransportAddress,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        destinationInfo                 SEQUENCE OF AliasAddress OPTIONAL,
        destExtraCallInfo               SEQUENCE OF AliasAddress OPTIONAL,
```

```
        destinationType                 EndpointType OPTIONAL,
        remoteExtensionAddress          SEQUENCE OF AliasAddress OPTIONAL,
        alternateEndpoints              SEQUENCE OF Endpoint OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL,
        supportsAnnexECallSignalling    BOOLEAN
}

LocationReject ::= SEQUENCE --(LRJ)
{
        requestSeqNum                   RequestSeqNum,
        rejectReason                    LocationRejectReason,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        altGKInfo                       AltGKInfo OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL
}

LocationRejectReason ::= CHOICE
{
        notRegistered           NULL,
        invalidPermission       NULL,   -- exclusion by administrator or feature
        requestDenied           NULL,   -- cannot find location
        undefinedReason         NULL,
        ...,
        securityDenial          NULL,
        routeCallToSCN          SEQUENCE OF PartyNumber,
        aliasesInconsistent     NULL    -- multiple aliases in request identify distinct people
}

DisengageRequest ::= SEQUENCE --(DRQ)
{
        requestSeqNum                   RequestSeqNum,
        endpointIdentifier              EndpointIdentifier,
        conferenceID                    ConferenceIdentifier,
        callReferenceValue              CallReferenceValue,
        disengageReason                 DisengageReason,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        callIdentifier                  CallIdentifier,
        gatekeeperIdentifier            GatekeeperIdentifier OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL,
        answeredCall                    BOOLEAN
}

DisengageReason ::= CHOICE
{
        forcedDrop                      NULL,   -- gatekeeper is forcing the drop
        normalDrop                      NULL,   -- associated with normal drop
        undefinedReason                 NULL,
        ...
}

DisengageConfirm ::= SEQUENCE --(DCF)
{
        requestSeqNum                   RequestSeqNum,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL
}

DisengageReject ::= SEQUENCE --(DRJ)
{
        requestSeqNum                   RequestSeqNum,
        rejectReason                    DisengageRejectReason,
        nonStandardData                 NonStandardParameter OPTIONAL,
        ...,
        altGKInfo                       AltGKInfo OPTIONAL,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL
}
```

```
DisengageRejectReason ::= CHOICE
{
        notRegistered           NULL,   -- not registered with gatekeeper
        requestToDropOther      NULL,   -- cannot request drop for others
        ...,
        securityDenial          NULL
}

InfoRequest ::= SEQUENCE --(IRQ)
{
        requestSeqNum                   RequestSeqNum,
        callReferenceValue              CallReferenceValue,
        nonStandardData                 NonStandardParameter OPTIONAL,
        replyAddress                    TransportAddress OPTIONAL,
        ...,
        callIdentifier                  CallIdentifier,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL,
        uuiesRequested                  UUIEsRequested OPTIONAL
}

InfoRequestResponse ::= SEQUENCE --(IRR)
{
        nonStandardData                 NonStandardParameter OPTIONAL,
        requestSeqNum                   RequestSeqNum,
        endpointType                    EndpointType,
        endpointIdentifier              EndpointIdentifier,
        rasAddress                      TransportAddress,
        callSignalAddress               SEQUENCE OF TransportAddress,
        endpointAlias                   SEQUENCE OF AliasAddress OPTIONAL,
        perCallInfo                     SEQUENCE OF SEQUENCE
        {
                nonStandardData                 NonStandardParameter OPTIONAL,
                callReferenceValue              CallReferenceValue,
                conferenceID                    ConferenceIdentifier,
                originator                      BOOLEAN OPTIONAL,
                audio                           SEQUENCE OF RTPSession OPTIONAL,
                video                           SEQUENCE OF RTPSession OPTIONAL,
                data                            SEQUENCE OF TransportChannelInfo OPTIONAL,
                h245                            TransportChannelInfo,
                callSignalling                  TransportChannelInfo,
                callType                        CallType,
                bandWidth                       BandWidth,
                callModel                       CallModel,
                ...,
                callIdentifier                  CallIdentifier,
                tokens                          SEQUENCE OF ClearToken OPTIONAL,
                cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
                substituteConfIDs               SEQUENCE OF ConferenceIdentifier,
                pdu                             SEQUENCE OF SEQUENCE
                {
                        h323pdu         H323-UU-PDU,
                        sent            BOOLEAN         -- TRUE is sent, FALSE is received
                } OPTIONAL
        } OPTIONAL,
        ...,
        tokens                          SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue             ICV OPTIONAL,
        needResponse                    BOOLEAN
}

TransportChannelInfo ::= SEQUENCE
{
        sendAddress                     TransportAddress OPTIONAL,
        recvAddress                     TransportAddress OPTIONAL,
        ...
}

RTPSession ::= SEQUENCE
{
        rtpAddress                      TransportChannelInfo,
        rtcpAddress                     TransportChannelInfo,
        cname                           PrintableString,
        ssrc                            INTEGER (1..4294967295),
        sessionId                       INTEGER (1..255),
        associatedSessionIds            SEQUENCE OF INTEGER (1..255),
        ...
}
```

```
InfoRequestAck ::= SEQUENCE --(IACK)
{
        requestSeqNum                RequestSeqNum,
        nonStandardData              NonStandardParameter OPTIONAL,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL,
        ...
}

InfoRequestNak ::= SEQUENCE --(INAK)
{
        requestSeqNum                RequestSeqNum,
        nonStandardData              NonStandardParameter OPTIONAL,
        nakReason                    InfoRequestNakReason,
        altGKInfo                    AltGKInfo OPTIONAL,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL,
        ...
}

InfoRequestNakReason ::= CHOICE
{
        notRegistered         NULL,   -- not registered with gatekeeper
        securityDenial        NULL,
        undefinedReason       NULL,
        ...
}

NonStandardMessage ::= SEQUENCE
{
        requestSeqNum                RequestSeqNum,
        nonStandardData              NonStandardParameter,
        ...,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL
}

UnknownMessageResponse ::= SEQUENCE   -- (XRS)
{
        requestSeqNum                RequestSeqNum,
        ...,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL
}

RequestInProgress ::= SEQUENCE              -- (RIP)
{
        requestSeqNum                RequestSeqNum,
        nonStandardData              NonStandardParameter  OPTIONAL,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL,
        delay                        INTEGER(1..65535),
        ...
}

ResourcesAvailableIndicate ::= SEQUENCE --(RAI)
{
        requestSeqNum                RequestSeqNum,
        protocolIdentifier           ProtocolIdentifier,
        nonStandardData              NonStandardParameter OPTIONAL,
        endpointIdentifier           EndpointIdentifier,
        protocols                    SEQUENCE OF SupportedProtocols,
        almostOutOfResources         BOOLEAN,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL,
        ...
}
```

```
ResourcesAvailableConfirm ::= SEQUENCE --(RAC)
{
        requestSeqNum                RequestSeqNum,
        protocolIdentifier           ProtocolIdentifier,
        nonStandardData              NonStandardParameter OPTIONAL,
        tokens                       SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
        integrityCheckValue          ICV OPTIONAL,
        ...
}

END             -- of ASN.1
```

ANEXO I

Paquetización de vídeo H.263+

En IETF RFC 2429 se especifica un formato de cabida útil del RTP para trenes de bits de vídeo H.263 que contienen las nuevas características "H.263+" adoptadas en la versión 2 (1998) de la Recomendación H.263 (que incluye las características que utilizan PLUSTYPE o los anexos I a T de H.263).

Es necesario que los trenes de bits H.263 que no utilizan las nuevas características de la versión 2 de H.263 tengan la capacidad de soportar el formato de cabida útil H.263 de la norma RFC 2190 como se especifica en el anexo E de la presente Recomendación, por razones de compatibilidad con implementaciones previas. Sin embargo, el nuevo formato de cabida útil especificado en RFC 2429 debiera utilizarse aún para trenes de bits que no contienen las nuevas características de la versión 2 siempre que el formato de cabida útil más nuevo posea las capacidades de los terminales de recepción.

APÉNDICE I

Algoritmos RTP/RTCP

El material informativo referenciado puede encontrarse en:

– SCHULZRINNE (H.), CASNER (S.), FREDERICK (R.), JACOBSON (V.): RTP: A Transport Protocol for Real-Time Applications, RFC 1889, *Internet Engineering Task Force*, 1996.

APÉNDICE II

Perfil RTP

El material informativo referenciado figura en:

– SCHULZRINNE (H.): RTP Profile for Audio and Video Conferences with Minimal Control, RFC 1890, *Internet Engineering Task Force*, 1996.

APÉNDICE III

Paquetización H.261

El material informativo referenciado figura en:

– TURLETTI (T.), HUITEMA (C.): RTP Payload Format for H.261 Video Stream, RFC 2032, *Internet Engineering Task Force*, 1996.

APÉNDICE IV

Funcionamiento de H.225.0 en distintas pilas de protocolos de la red de paquetes

En este apéndice figuran detalles adicionales relativos al funcionamiento de H.225.0 en distintas pilas reales de protocolos de la red de paquetes. Las redes de paquetes utilizadas en esta Recomendación proporcionarán modos de funcionamiento fiables y no fiables, incluido un medio para distinguir fronteras de paquetes.

IV.1 TCP/IP/UDP

Adviértase que UDP puede fragmentar y reensamblar grandes paquetes de vídeo, pero un fracaso en la ejecución de la paquetización de MB puede conducir a la pérdida de un GOB completo.

La multidifusión IP debe utilizarse para la distribución GRQ en oposición a la difusión de capa de acceso a medios.

<table>
<tr><th>Aplicaciones de entrega no fiable</th><th>Señalización de llamada y canal H.245</th></tr>
<tr><td rowspan="2">UDP</td><td>TPKT</td></tr>
<tr><td>TCP</td></tr>
<tr><td colspan="2">IP</td></tr>
<tr><td colspan="2">Capa de enlace</td></tr>
<tr><td colspan="2">Capa física</td></tr>
</table>

Un TPKT es un formato de paquete como se define en IETF RFC 1006. Se utiliza para delimitar mensajes particulares (PDU) dentro del tren TCP, que proporciona un tren continuo de octetos sin límites explícitos. Un TPKT consta de un campo de número de versión de un octeto seguido de un campo reservado de un octeto, seguido de un campo de longitud de dos octetos, seguido de los datos reales. El campo número de versión contendrá el valor "3", el campo reservado contendrá el valor "0". El campo de longitud contendrá la longitud de todo el paquete incluido los campos número de versión, reservado y de longitud con una palabra "big-endian" (los bytes de la izquierda son los más significativos) de 16 bits.

IV.1.1 Descubrimiento del controlador de acceso

IV.1.1.1 Descubrimiento utilizando dirección multidifusión o puerto conocido

Tras el descubrimiento del controlador de acceso y los procedimientos de registro descritos en la cláusula 7/H.323, los puntos extremos deben utilizar la siguiente dirección multidifusión o puerto conocido cuando intenten descubrir el controlador de acceso apropiado para su configuración de red:

– Dirección UDP para comunicación multidifusión con controladores de acceso: 224.0.1.41.

– Puerto UDP para comunicación multidifusión con controladores de acceso: 1718.

– Puerto UDP para comunicación RAS unidifusión donde no existe "otro acuerdo": 1719.

NOTA – Cabe señalar que "otro acuerdo" puede incluir el registro de un punto extremo con un controlador de acceso.

Se señala que las implementaciones deberán tener en cuenta el alcance de la multidifusión a fin de no sobrecargar Internet con mensajes de descubrimiento.

Suponiendo que un controlador de acceso tiene una dirección IP por ejemplo de 134.134.12.1, puede tener lugar la señalización siguiente:

– LRQ o GRQ llega a 134.134.12.1: puerto 1719.

– LRQ o GRQ llega a 134.134.12.1: puerto 1718 (se señala que esto puede ocurrir con los controladores de acceso de la versión 1).

– LRQ o GRQ llega a 224.0.1.41: puerto 1718.

El controlador de acceso puede transmitir un mensaje LRQ a las siguientes direcciones:

– 224.0.1.41: puerto 1718 (multidifusión a todos los controladores de acceso);

– X.X.X.X: puerto 1719 (a un determinado controlador de acceso).

El puerto 1719 sólo se debe utilizar cuando se envía una petición unidifusión. Esto permite al receptor conocer si debe enviar un rechazo (xRJ) al emisor (debe hacerlo en todos los casos).

El puerto 1718 sólo se debe utilizar cuando se envía una petición multidifusión. El receptor debe enviar la respuesta adecuada dependiendo del mensaje. Para LRQ que no requiere rechazo, el receptor no debe responder las peticiones multidifusión. Para GRQ se debe enviar un GRJ dirigido a la fuente de la GRQ.

IV.1.1.2 Descubrimiento utilizando DNS (informativo)

IV.1.1.2.1 Un URL para controladores de acceso

En primer lugar, obsérvese que un controlador de acceso se identifica mediante una dirección de transporte y un gatekeeperIdentifier (identificador de controlador de acceso), que es una cadena. Un controlador de acceso es un recurso particular en Internet, y por ello es lógico especificarlo en un localizador de recurso uniforme (URL, *uniform resource locator*). El protocolo formulado por el controlador de acceso es RAS, por lo que el URL para un controlador de acceso puede venir dado por:

ras://gkID@domainname

gkID (ID de controlador de acceso) es el gatekeeperIdentifier, y domainname (nombre de dominio) es un nombre del sistema de nombres de dominio (DNS, *domain name system*) que identifica el dominio del controlador de acceso. Se señala que no necesariamente es éste un nombre de dominio totalmente cualificado (FQDN, *fully qualified domain name*) con un registro A; no es preciso que este nombre de dominio tenga una interfaz de transporte física con un número IP registrado en el DNS. No obstante, si es un FQDN, es razonable insistir en que su número IP es el del controlador de acceso al que se refiere el URL. En este caso, se puede añadir un número de puerto opcional al URL:

ras://gkID@domainname:port_no.

Si no se proporciona ningún número de puerto, el valor conocido de 1719 se toma como un valor por defecto.

El caso más interesante es cuando no se trata de un FQDN, y el nombre de dominio no designa, por tanto, una dirección de transporte indicada en el DNS. El nombre de dominio puede referirse entonces a una pura "zona de autoridad del controlador de acceso". En la próxima subcláusula se explica cómo localizar el controlador de acceso en este caso.

IV.1.1.2.2 Localización del URL

El URL no resuelve el problema de localizar el controlador de acceso, sólo proporciona un formato normalizado de la información que hay que hallar. El problema consiste en cómo producir una dirección de transporte y un gatekeeperIdentifier para la señalización RAS dado el nombre de dominio de un controlador de acceso.

Si el controlador de acceso tiene un identificador conforme a IETF RFC 822, es fácil extraer un nombre de dominio a partir del identificador de un controlador de acceso conforme a RFC 822. De hecho, puede ser conveniente proporcionar a los puntos extremos identificadores conformes a RFC 822 y luego estipular que la parte nombre de dominio del identificador se refiere al dominio del controlador de acceso.

IV.1.1.2.2.1 Indagación de registro de recursos SRV

La primera solución tiene en cuenta el hecho de que el controlador de acceso es básicamente un servicio de sistema, y que la dirección de transporte de un servicio de sistema nominado se puede extraer del DNS mediante la indagación de un nuevo tipo de registro de recurso DNS, denominado registro de localización de servicio (SRV, *service location record*). Dado un nombre de dominio, se efectuará una indagación de registro SRV de la dirección de transporte del servicio RAS para ese dominio. El propio nombre de dominio, o el que se devuelve en la respuesta SRV, se utiliza como gatekeeperIdentifier.

Esta sencilla solución se normalizará muy pronto. El problema es que casi ninguna implementación actual de cliente o servidor DNS soporta todavía el registro de recursos SRV. A menos que el cliente DNS conozca el tipo de registro de recursos SRV, no es posible que pueda dar curso a indagaciones en este registro de recursos. Mientras dicho soporte no se generalice, existe la posibilidad razonable de que falle la indagación SRV.

IV.1.1.2.2.2 Indagación de registro TXT

Todas las implementaciones DNS actuales soportan el registro de recurso TXT. Se trata, básicamente, de algún texto libre que puede ser devuelto para cada nombre de dominio. Es posible almacenar muchos recursos TXT para un solo dominio. La norma estipula que se devuelvan todos los registros TXT cuando se efectúe una indagación sobre ellos.

Es posible utilizar indagaciones TXT si fallan las indagaciones SRV. Se supone que para extraer un nombre de dominio se utiliza el mismo convenio que se propuso anteriormente. Para gatekeeperIdentifiers pueden utilizarse cadenas conformes a RFC 822 (nombres de correo electrónico "-like") o cadenas conformes a IETF RFC 1768 (URL). En cualquiera de los dos casos se utiliza el nombre de dominio para efectuar una indagación TXT de DNS sobre el nombre de dominio. Los registros de recursos devueltos son líneas de texto libre, y el terminal buscará entonces líneas en la respuesta de la forma:

ras [< gk id>@]<domain name>[:<portno>] [<priority>]

El campo **<gk id>** es un ID de controlador de acceso opcional que está separado del nombre de dominio. Si este campo está ausente, se supone que el propio dominio es entonces el ID de controlador de acceso.

El campo **<domain name>** puede ser el nombre del registro A que contiene la dirección IP del controlador de acceso, o bien una dirección IP bruta en forma de puntos. No es necesario que el nombre de dominio esté totalmente cualificado; si no lo está, el subdominio en el que se localizó el registro TXT se añadirá a él para formar el nombre de registro A totalmente cualificado.

El campo opcional **[:<portno>]** puede utilizarse para especificar un número de puerto distinto del puerto RAS normalizado.

El campo opcional **[<priority>]** especifica el orden en que se debe acceder a los controladores de acceso enumerados para el descubrimiento o las indagaciones LRQ si hay más de un registro TXT ras. Los números más bajos tienen mayor prioridad.

Obsérvese que, si el campo <gk id> está ausente, este formato supone que los ID de controlador de acceso son en realidad nombres de dominio legales. No obstante, si es necesario que un solo sistema central soporte múltiples controladores de acceso lógicos, cada uno con un ID distinto, el formato lo soportará. Esto es así porque los registros A separados pueden contener la misma dirección IP.

Se utilizan espacios en blanco como delimitadores entre **ras** y **gk id**, si están presentes, o **domain name**, y entre **portno** y **priority**. Los espacios en blanco están formados por cualquier número de espacios o tabuladores.

Ejemplos de registros TXT de controlador de acceso válidos:

ras gk1

ras gk1.company.com

ras gk1:1500 3

ras 172.11.22.33:1500 2

El cliente analiza las líneas y a partir de ellas obtiene la dirección de transporte del controlador de acceso dentro de ese dominio a la que puede enviar mensajes RAS.

Puesto que el DNS requiere un servidor para devolver todos los registros TXT asociados con un nombre de dominio, el cliente puede seleccionar y procesar únicamente los registros que le son útiles. Permite también al DNS devolver una lista ordenada de controladores de acceso que pueden servir como alternativas y reservas, tal como se define en la presente Recomendación en el marco de la estructura AlternateGK ASN.1.

Se señala que el servidor devuelto en esa indagación podría ser una dirección de transporte real en notación decimal de puntos, o un FQDN que necesita una indagación de registro A en DNS para determinar la dirección de transporte. La ventaja de utilizar un FQDN es que habitualmente se ocultan los números IP reales. La ventaja de utilizar números IP consiste en que se evita una segunda indagación en DNS, acortando así el tiempo previo al establecimiento de la llamada.

IV.1.1.2.3 Procesamiento de los ID de correo electrónico por el controlador de acceso durante los mensajes ARQ y LRQ

Cuando el campo **destinationInfo (información de destino)** de un mensaje ARQ o LRQ contiene una dirección de alias **email-ID (identificador de correo electrónico)**, el controlador de acceso debe verificar primero la base de datos de su registro en relación con el alias. Si no puede resolverlo, el controlador de acceso debe analizar el alias para recuperar su porción de dominio. Si no se proporciona ningún dominio, el controlador de acceso puede generar un dominio por defecto. El dominio se utiliza entonces para localizar uno o más controladores de acceso, utilizando los procedimientos indicados en IV.1.1.2.2. El controlador de acceso puede indagar a todos los controladores de acceso así localizados con un intercambio de mensajes LRQ/LCF/LRJ.

Obsérvese que más de un controlador de acceso puede tener registros TXT correspondientes en un solo dominio DNS. En consecuencia, un solo dominio DNS puede "contener" más de una zona H.323. Por ello, incluso si un controlador de acceso no puede resolver un ID de correo electrónico cuya porción de dominio es uno de sus dominios por defecto, puede indagar todavía otras zonas en el mismo dominio DNS.

Si el controlador de acceso es presentado con un alias no registrado que es un **h323-id** y el ID puede ser interpretado como una porción de usuario legal de un nombre RFC 822, el controlador de acceso puede interpretar el alias como si fuera un ID de correo electrónico en su dominio por defecto e intentar localizar el alias en algún otro controlador de acceso. Del mismo modo, el controlador de acceso puede quitar el nombre de dominio de un ID de correo electrónico procedente de un mensaje LRQ entrante para que pueda ser localizado como un h323-ID.

IV.1.2 Comunicaciones de punto extremo a punto extremo

Los puntos extremos que deseen recibir llamadas de puntos extremos que caen fuera de la zona de su controlador de acceso deben utilizar el siguiente puerto por el canal de señalización de llamada:

– Puerto de señalización de llamada TCP de punto extremo 1720

Aunque está permitido utilizar valores dinámicos para que estos puertos permitan múltiples puntos extremos en un único dispositivo, debe entenderse que esto impedirá la interoperación con puntos extremos que quedan fuera de la zona del controlador de acceso, excepto vía de una pasarela en la zona.

IV.2 SPX/IPX

Adviértase que dado que no hay ningún reensamblamiento de red de paquetes grandes, es esencial el uso de fragmentación de MB.

Aplicaciones de entrega no fiable	**Canal H.245 canal de señalización de llamada**
PXP	SPX
IPX	
Capa de enlace	
Capa física	

IV.2.1 Descubrimiento del controlador de acceso

En terminología IPX, un "zócalo" es el equivalente de un "puerto" en IP y un "identificador TSAP" en esta Recomendación y en la Recomendación H.323.

En las redes basadas en IPX, los controladores de acceso deben anunciar el "tipo de servicio de controlador de acceso" definido más adelante para permitir a los puntos extremos localizarlos en una red. Análogamente, los puntos extremos deben solicitar al "tipo de servicio de controlador de acceso" que encuentre la ubicación del controlador de acceso más próximo.

– Tipo de servicio de controlador de acceso En estudio

NOTA – El tipo de servicio se denomina zócalo SAP (SAP socket) en alguna documentación IPX.

IV.2.2 Comunicación de punto extremo a punto extremo

Los puntos extremos que desean recibir llamadas de puntos extremos que caen fuera de la zona de su controlador de acceso, deben utilizar los siguientes zócalos para señalización de llamada.

– Puerto de señalización de llamada IPX de punto extremo En estudio

Aunque se permite utilizar valores dinámicos para que estos zócalos permitan múltiples puntos extremos en un solo dispositivo, debe entenderse que esto evitará la interoperación con puntos extremos que caen fuera de la zona del controlador de acceso, salvo vía una pasarela en la zona.

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 multimedios**
Serie I	Red digital de servicios integrados
Serie J	Transmisiones de señales radiofónicas, de televisión y de otras señales multimedios
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	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 protocolo Internet
Serie Z	Lenguajes y aspectos generales de soporte lógico para sistemas de telecomunicación