The ATM Forum

Technical Committee

Gateway for H.323 Media Transport Over ATM

AF-SAA-0124.000

July, 1999

Gateway for H.323 Media transport Over ATM

The ATM Forum
Worldwide Headquarters
2570 West El Camino Real, Suite 304
Mountain View, CA 94040-1313
Tel: +1-650-949-6700
Fax: +1-650-949-6705

A C K N O W L E D G M E N T S

Preparation of a Specification of this kind requires a concerted effort on the part of many individuals. The following individuals (names listed alphabetically), among others, provided written contributions and devoted valuable time towards the development of this Specification.

Marek Kotelba
Carlos Pazos
Joo Myoung Seok
Doug Young Suh
Andrew Malis
K.K. Ramakrishnan
Curtis Siller
Surya Varanasi

François Audet, Editor

Ameneh Zahir, Chair of ATM Forum SAA/RMOA Working Group

Bahman Mobassser, Chair of ATM Forum SAA Working Group

Contents

1 Introduction

1.1 Purpose and scope

This document specifies the ATM Forum's implementation agreement for Real-time Multimedia Over ATM service including telephony, video-conferencing and distance learning based on the exchange of audio, video, data. It describes the access to an ATM network using H.323. The H.323 terminal can be on a variety of network technologies, including non-native ATM IP-based (Ethernet, etc.), and native ATM.

Carrying H.323 media streams over ATM will ensure that the media streams take advantage of the inherent quality of service of ATM. However, to ensure a suitable end-to-end quality of service, it is necessary that appropriate quality of service mechanisms are applied outside the ATM network. It is not within the scope of this specification to describe these mechanisms.

The RTP compression mechanism described in section 3 may be applicable to protocols other than H.323.

1.2 References

1.2.1 Normative

[1] ATM Forum Technical Committee, AF-SIG-0061.000, *ATM User-Network Interface (UNI) Signalling Specification Version 4.0*, June 1996

[2] ITU-T G.711, *Pulse code modulation (PCM) of voice frequencies*

[3] ITU-T Recommendation H.246, *Interworking of H.Series multimedia terminals with H.Series multimedia terminals and voice terminals on GSTN and ISDN*

[4] ITU-T Recommendation H.225.0, *Media stream packetization and synchronization for visual telephone systems on non-guaranteed quality of service LANs*

[5] ITU-T Recommendation H.245, *Control protocol for multimedia communications*

[6] ITU-T Recommendation H.323, *Packet based multimedia communications systems*

[7] ITU-T I.363.56, *B-ISDN ATM Adaptation Layer (AAL) Specification, Type 5*

[8] ITU-T Q.2931, *Broadband Integrated Services Digital Network (B-ISDN); Digital Subscriber Signalling System No. 2 (DSS2); User-Network Interface (UNI) Layer 3 Specification for Basic Call/Connection Control*

[9] ITU-T Q.2941.1, *Generic Identifier Transport*

[10] ITU-T *Implementers Guide for the ITU-T H.323, H.225.0, H.245, H.246, H.235 and H.450 Series Recommendations - Packet-Based Multimedia Systems*

1.2.2 Informative

[11] ATM Forum Technical Committee, AF-VTOA-0083.001, *Voice and Telephony Over ATM to the Desktop,* March 1999

[12] ATM Forum Technical Committee, AF-SAA-0049.001, *Audiovisual Multimedia Services :Video on Demand Specification 1.1*, March 1997

[13] ATM Forum Technical Committee, AF-VTOA-0113.000, *ATM Trunking using AAL2 for Narrowband Services*, February 1999

[14] CCITT Recommendation G.722, *7 kHz audio-coding within 64 kbit/s*.

[15] ITU-T Recommendation G.723.1, *Speech coders: Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s*.

[16] CCITT Recommendation G.728, *Coding of speech at 16 kbit/s using low-delay code excited linear prediction*.

[17] ITU-T Recommendation G.729, *Coding of speech at 8 kbit/s using conjugate structure algebraic-code-excited linear-prediction (CS-ACELP)*.

[18] ITU-T Recommendation H.310, *Broad-band Audiovisual Communication Systems and terminals*

[19] ITU-T Recommendation H.320, *Narrowband Visual Telephone Systems and Terminal Equipment*

[20] ITU-T Recommendation H.321, *Adaptation of H.320 Visual Telephone Terminals to B-ISDN Environments*

[21] ITU-T Recommendation I.366.2, *AAL type 2 Service Specific Convergence Sublayer for Trunking*

[22] IETF RFC 768 (STD 6), *User Datagram Protocol*, ftp://ftp.isi.edu/in-notes/rfc768.txt

[23] IETF RFC 791 (STD 5), *Internet Protocol*, ftp://ftp.isi.edu/in-notes/rfc791.txt

[24] IETF RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*, ftp://ftp.isi.edu/in-notes/rfc1483.txt

[25] IETF RFC 1889, *RTP: A Transport Protocol for Real-Time Applications*, January 1996, ftp://ftp.isi.edu/in-notes/rfc1889.txt

[26] IETF RFC 1890, *RTP Profile for Audio and Video Conferences with Minimal Control*, January 1996, ftp://ftp.isi.edu/in-notes/rfc1890.txt

[27] IETF RFC 2508, *Compressing IP/UDP/RTP Headers for Low-Speed Serial Links*, ftp://ftp.isi.edu/in-notes/rfc2508.txt, Casner, Jacobson

[28] IETF RFC 2507, *IP Header Compression*, ftp://ftp.isi.edu/in-notes/rfc2507.txt, Degermark, Nordgren, Pink

[29] IETF RFC 2509, *IP Header Compression over PPP*, ftp://ftp.isi.edu/in-notes/rfc2509.txt, Engan, Casner, Bormann

1.3 Abbreviations and acronyms

AAL	ATM Adaptation Layer
ATM	Asynchronous Transfer Mode
APDU	Application Protocol Data Unit
C/D	Compressor/Decompressor
CLP	Cell Loss Priority
CPCS	Common Part Convergence Sublayer
CPCS-PDU	CPCS Protocol Data Unit
CPCS-SDU	CPCS Service Data Unit
CPCS-UU	CPCS User-to-User Indication
CRC	Cyclic Redundancy Check

CSRC	Contributing Source
IP	Internet Protocol
ISDN	Integrated Services Digital Network
ISO	International Organization for Standardization
ITU-T	International Telecommunication Union - Telecommunication Standardization Sector
LANE	LAN Emulation
MGC	Media Gateway Controller
MG	Media Gateway
MTU	Maximum Transmission Unit
OUI	Organizational Unique Identifier
PDU	Protocol Data Unit
PCM	Pulse Code Modulation
PNNI	Private Network-Network Interface
PPP	Point-to-Point Protocol
PT	Payload Type
RAS	Registration, Administration and Status
RTCP	Real Time Control Protocol
RTP	Real Time Protocol
SID	Silence Insertion Descriptor
SSCS	Service Specific Convergence Sublayer
SSRC	Synchronization Source
SVC	Switched Virtual Connection
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UNI	User-Network Interface
VC	Virtual Connection
VCI	Virtual Channel Identifier
VPI	Virtual Path Identifier
VTOA	Voice and Telephony Over ATM

1.4 Reference configuration

This specification makes use of H.323-to-H.323 gateways to interwork between ATM and non-ATM IP networks as shown in Figure 1. The term H.323-to-H.323 gateway implies that it is addressable as a gateway as defined in H.323, and it maps between H.323 on a non-ATM IP network and H.323 on an ATM network.

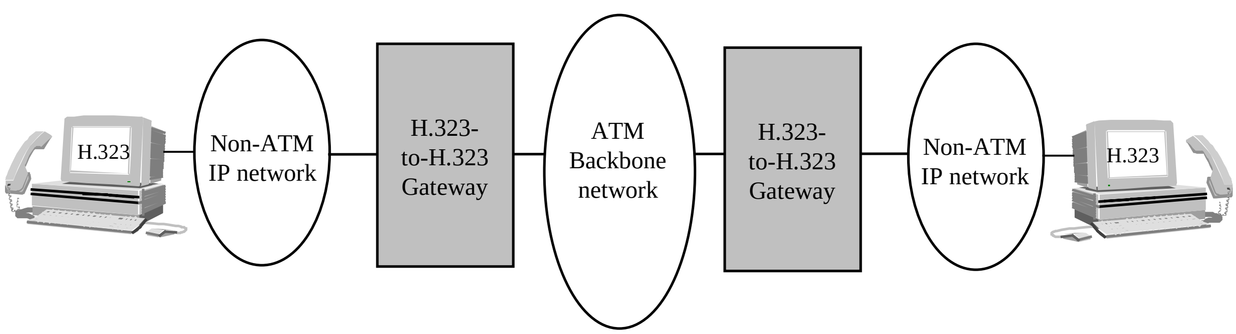

Figure 1 Location of the Gateway

A gateway can be functionally decomposed in 3 as illustrated in Figure 2.

1. Media Gateway

   The Media Gateway converts between the RTP/UDP/IP media stream on the non-ATM network and the compressed RTP media stream over an ATM SVC. The compressor/decompressor function of the media gateway is referred-to as C/D.

2. Media Gateway Controller

   The Media Gateway Controller includes H.323 control, i.e., H.225.0 and H.245. Since the gateway is between two H.323 systems using H.225.0/H.245, its functionality is quite simple.

3. ATM Signalling

   The ATM Signalling component is responsible for setting up SVCs to transport the RTP media stream on the ATM network.

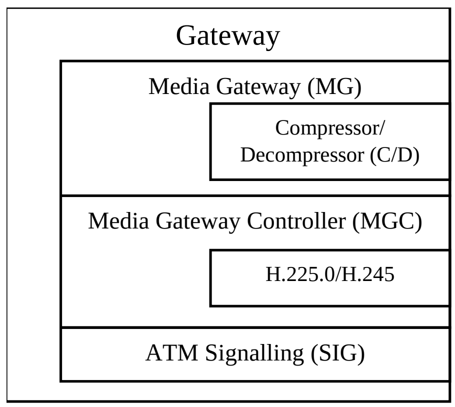

Figure 2 Functional decomposition of the Gateway

For the purpose of this specification, the Media Gateway, the Media Gateway Controller and the ATM Signalling component are co-located. Physical separation of the Media Gateway, the Media Gateway Controller and the ATM Signalling component is possible, but considered to be outside the scope of this specification and could require additional protocols between the decomposed components of the gateway. The information provided by the H.225.0/H.245 protocol to the Media Gateway Controller is used by the ATM Signalling component to set up the SVCs. This process is further explained in section 2.

Figure 3 illustrates how the different functions of the gateway relate to each other in the overall reference configuration. The H.323 control traffic (H.245/H.225.0) is terminated in the Media Gateway Controller, while the media streams are processed by the C/D function in the Media Gateway. The compression mechanism is described in section 3. The ATM Signalling component is responsible for the establishment and termination of a SVC that carries media traffic.

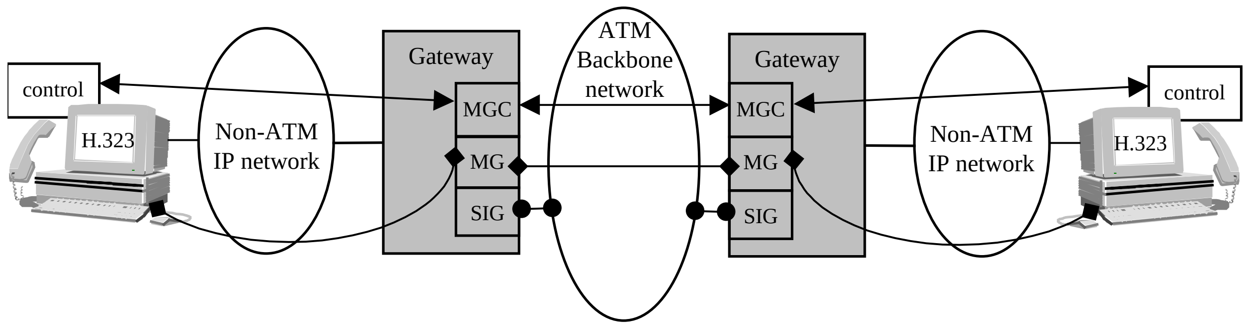

Figure 3 Overall reference configuration

1.5 Service requirements

A variety of devices can use H.323, under a variety of service configurations and network technologies.

The H.323 control information is transported using an IP over ATM method (e.g., RFC 1483, LANE).

The H.323 media stream information can be transported in an ATM network using three different mechanisms. Each of these mechanisms uses a different protocol stack for the media stream. The three mechanisms are:

- IP over ATM media stream transport (H.323),
- RTP over ATM media stream transport (H.323 /Annex C),
- Compressed RTP over ATM media stream transport.

This specification addresses compressed RTP over ATM media stream transport. Some ATM H.323 terminals may operate in more than one mode of operation. As specified in H.323/Annex C, an RTP over ATM media stream transport terminal has to also be able to operate as an IP over ATM media stream transport terminal.

Annex A provides guidelines on choosing voice packet sizes.

1.5.1 IP over ATM media stream transport (H.323)

H.323 in the basic mode of operation uses an IP over ATM method to carry the media stream over ATM instead of using a separate SVC. It therefore does not utilize the inherent Quality of Service capabilities of ATM. It is also very bandwidth inefficient because of the significant overhead introduced by using RTP (at least 12 octets), UDP (8 octets) and IP (at least 20 octets). The multiprotocol encapsulation header (RFC 1483) adds another 8 octets. Note that other methods such as LANE also add a header with its own overhead. This inefficiency is more crucial when the multimedia session is voice only because of the very small size of the voice packets.

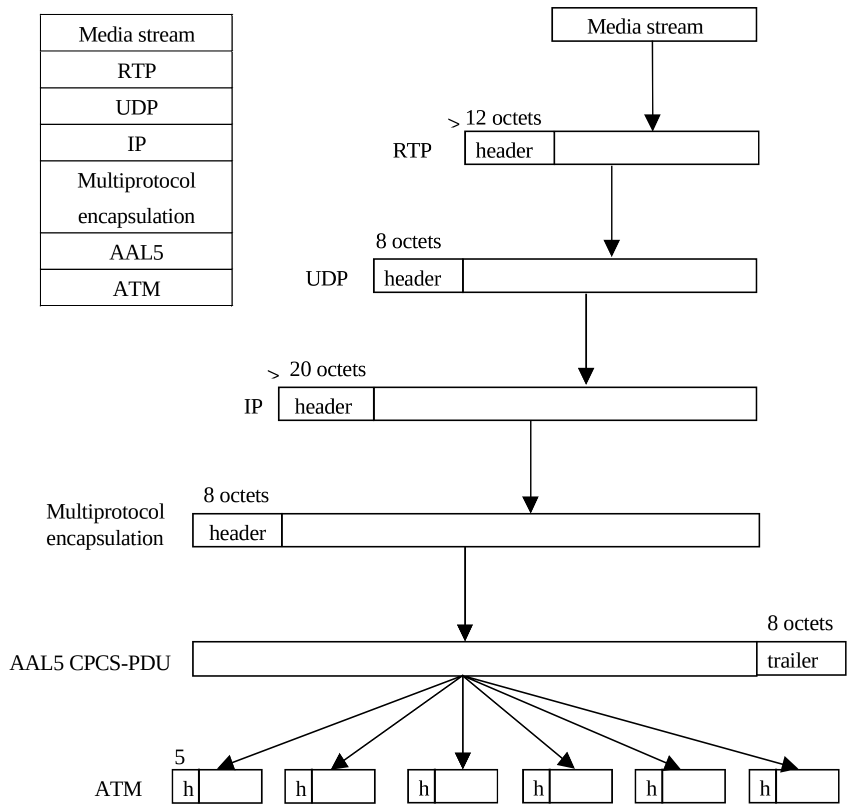

Figure 4 IP over ATM media stream transport

1.5.2 RTP over ATM media stream transport (H.323/Annex C)

H.323/Annex C uses AAL5 directly to transport media streams on RTP. It is an improvement over the IP over ATM method because it sets up an SVC for the media transport enabling the use of ATM Quality of Service. While better than for IP over ATM media stream transport, it is still quite bandwidth inefficient (especially for voice only multimedia sessions) because of the significant overhead of the RTP protocol (at least 12 octets). H.323/Annex C is intended for use by native ATM terminals. When a native ATM terminal interoperates with a non-ATM H.323 terminal, it will not operate in H.323/Annex C mode but in an IP over ATM method.

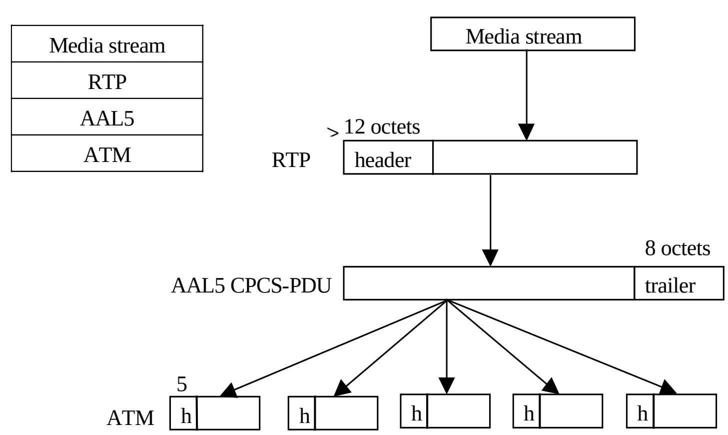

Figure 5 RTP over ATM media stream transport

1.5.3 Compressed RTP over ATM media stream transport

The method described in this specification terminates the UDP/IP protocols on the media gateways, removing the need to transport these headers end-to-end. As such, it is appropriate to suppress the IP and UDP headers, and carry the media stream with compressed or uncompressed RTP headers, as appropriate. The technique of header compression then becomes one of compressing just the RTP headers. The call and connection control for this H.323 Gateway to H.323 Gateway case that allows termination of media streams at the gateway is described in section 2. The compression of RTP headers is described in section 3.

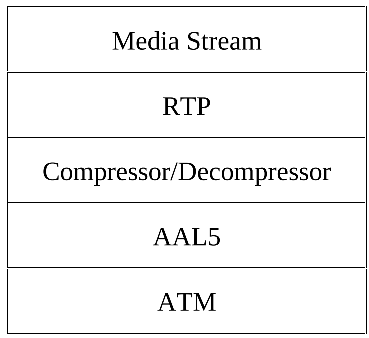

Figure 6 Compressed RTP over ATM media stream transport

2 Call and connection control

The reference architecture for establishment of the H.323 call using media stream transport over ATM is shown in Figure 7. For simplicity, only the case where both terminals are on non-ATM IP networks are shown. If one or more of the terminals were H.323/Annex C terminals, the gateway and terminal functionality would be merged: the two terminals can use the H.323/Annex C procedures and can additionally compress the RTP flow according to the compression method proposed in section 3.

- With respect to control traffic between the endpoints, each of the gateways to the ATM backbone network serves as an H.323-to-H.323 gateway.

- With respect to media streams between the endpoints, each of the gateways terminates the media stream for the near endpoint and acts as a compressor/decompressor for the RTP media stream.

Figure 7 is meant as an example of two endpoints communication through H.323-H.323 gateways. It is however necessary that the calls are terminated at Gateway X and Gateway Y at the edges of the ATM network.

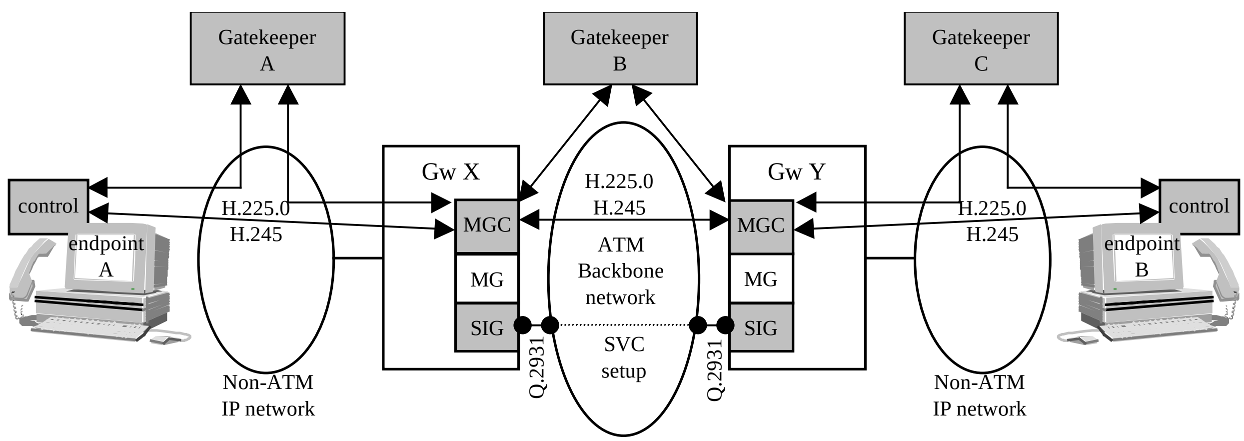

Figure 7 Architecture for call control

2.1 SVC Setup

SVCs are set up for the transport of RTP media streams based on H.245 or H.225.0 control messages exchanged by the endpoints and the gateway. In the ATM network, the corresponding control messages are sent between the gateways using an IP over ATM method.

SVCs for media streams are established at the edges of the ATM network, i.e., at the signalling gateway. H.323 has no concept of the reverse direction of a unidirectional logical channel. However, an important characteristic of point-to-point ATM VCs is that they are inherently bi-directional. The use of both directions of an ATM VC is therefore desirable. Otherwise, the audio and video streams will each need to be sent on two different VC's, one for each direction.

Outside of the ATM network, the logical channels are set-up uni-directionally. When possible, the gateway should establish a single SVC to transport two logical channels, one in each direction. This reduces the number of AAL5 SVCs to two in typical situations, one VC each for audio and for video, or to one for a voice-only call.

The following sections describe SVC setup based on control signalling, with and without the fast connect procedures.

2.1.1 SVC setup with fast connect

Fast connect is the preferred procedure for simple telephony calls. Fast connect provides faster call setup and solves some interoperability problems. It also simplifies the SVC setup process. For bi-directional calls, it is necessary to define a rule to determine which gateway is responsible for setting up the SVC, since the **fastStart** element in either direction may carry the ATM address needed to establish the SVC. It is logical that the side that initiates the H.323 call also initiates the SVC call set up. In the configuration of Figure 7, it will thus always be Gateway X that shall initiate the SVC setup when Endpoint A initiates the H.323 call.

Endpoint A initiates the call through an ARQ/ACF message exchange with the gatekeeper (Gatekeeper A) it is registered with, providing Endpoint B alias address. Gatekeeper A has Gateway X registered in its zone as the gateway serving the alias address of Endpoint B and returns that gateway's call signaling channel address in its ACF message to Endpoint A; such registration may have been manually configured. Endpoint A then sends a SETUP message to the call signaling address of Gateway X, providing Endpoint B's address. Therefore, from Endpoint A's perspective, the call proceeds as if the call was made to a local terminal on its non-ATM network. The SETUP message includes the **fastStart** element consisting of proposed options for the forward and reverse channels on the respective **OpenLogicalChannel** elements. See section 8.7.1/H.323. The respective **OpenLogicalChannel** elements also contain the transport addresses for the reverse RTCP and RTP channels directed to Gateway X.

On receiving the SETUP message, Gateway X initiates an ARQ/ACF exchange with Gatekeeper A to determine if it can accept the call. At this point in time, Gateway X may deny calls such as malicious calls. Gateway X may also send a CALL PROCEEDING message to Endpoint A so that Endpoint A does not timeout the call prematurely. Since the call goes over the ATM network, Gateway X also initiates an ARQ/ACF exchange with Gatekeeper B, providing Endpoint B's address. Gatekeeper B has all the H.323-H.323 gateways registered in its zone and therefore it knows each gateway's coverage area. For the ARQ from Gateway X containing the alias address of Endpoint B, Gatekeeper B returns the call signaling address for Gateway Y. Gateway X then sends a SETUP message (including **fastStart**) to Gateway Y.

On receiving the SETUP message, Gateway Y initiates an ARQ/ACF sequence with Gatekeeper B and optionally sends a CALL PROCEEDING message to Gateway X. Gateway Y also initiates an ARQ/ACF sequence with Gatekeeper C, providing Endpoint B's address. Then, Gateway Y sends a SETUP message (including **fastStart**) to endpoint B. After a successful ARQ/ACF exchange with Gatekeeper C, Endpoint B responds affirmatively to the fast connect procedure by including the **fastStart** element in a call control message to Gateway Y (CALL PROCEEDING, PROGRESS, ALERTING, or CONNECT) as per 8.7.1/H.323. Gateway Y shall then send the **fastStart** element to Gateway X.

Gateway X is responsible for establishing the ATM SVCs. If the logical channels indicated in the **fastStart** are bi-directional in nature (i.e., a pair of unidirectional channels), the gateway may set up a bi-directional SVC for the pair of logical channels to save on the number of SVCs used by the session. The SVC shall be set up upon receipt of the **fastStart** in the H.225.0 message from gateway Y. Gateway X shall establish as many SVCs as media streams opened. Since a Gateway may terminate many such SVCs to other gateways, the Q.2931 SETUP message carries the port number for the forward RTP media stream in the Generic Identifier Transport information element (see section 2.2). The RTP port number is equal to the RTCP Port number for the reverse direction minus one, and is sent in the **fastStart** element in an H.225.0 message. This information is used by Gateway Y to associate the incoming SVC call with the appropriate RTP channel. Gateway X shall only proceed with call set up procedures, including forwarding of the **fastStart** element to Endpoint A when the SVC setup is completed.

Note - CALL PROCEEDING has a local significance. When a called endpoint responds with ***fastStart*** *in a CALL PROCEEDING message, a gatekeeper or gateway that forwarded the SETUP message may already have responded to the calling endpoint with a CALL PROCEEDING message and the* ***fastStart*** *would thus be lost. The gatekeeper or gateway shall pass the* ***fastStart*** *element in a FACILITY message. This is in accordance with H.225.0 procedures. See the Implementers Guide [9].*

Figure 8 illustrates an example of setting-up a bi-directional communication for this scenario.

Endpoint B may send RTP packets before the SVC is setup, in which case the RTP packets will be delivered to Endpoint A using an RTP over ATM method with no QoS of the SVC.

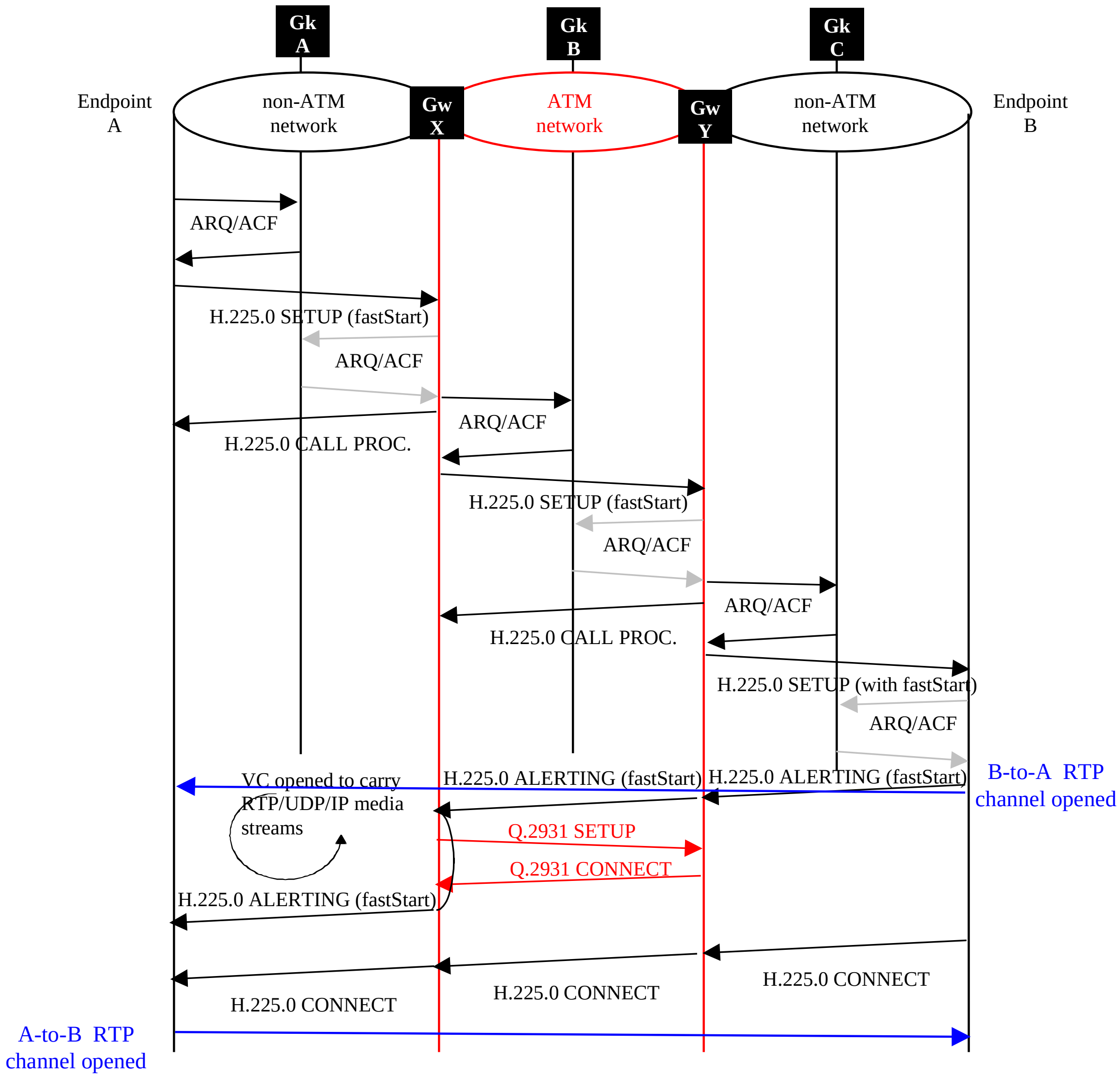

Figure 8 SVC setup with fast connect

2.1.2 SVC setup without Fast Connect

Figure 9 shows the control messages exchanged between the H.323 entities involved in the establishment of the H.245 control channel based on the architecture of Figure 7.

Endpoint A initiates the call through an ARQ/ACF message exchange with the gatekeeper (Gatekeeper A) it is registered with, providing Endpoint B alias address. Gatekeeper A has Gateway X registered in its zone as the gateway serving the alias address of Endpoint B and returns that gateway's call signalling channel address in its ACF message to Endpoint A; such registration may have been manually configured. Endpoint A then sends a SETUP message to the call signaling address of Gateway X, providing Endpoint B's address. Therefore, from Endpoint A's perspective, the call proceeds as if the call was made to a local terminal on its non-ATM network.

On receiving the SETUP message, Gateway X initiates an ARQ/ACF exchange with Gatekeeper A to determine if it can accept the call. At this point in time, gateway X may deny calls such as malicious calls. Gateway X may also send a CALL PROCEEDING message to Endpoint A so that Endpoint A does not timeout the call prematurely. Since the call goes over the ATM network, Gateway X also initiates an ARQ/ACF exchange with Gatekeeper B, providing Endpoint B's address. Gatekeeper B has all the H.323-H.323 gateways registered in its zone and therefore it knows each gateway's coverage area. For the ARQ from Gateway X containing the alias address of Endpoint B, Gatekeeper B returns the call signaling address for Gateway Y. Gateway X then sends a SETUP message to that address, providing Endpoint B's address.

On receiving the SETUP message, Gateway Y initiates an ARQ/ACF sequence with Gatekeeper B and optionally sends a CALL PROCEEDING message to Gateway X. Gateway Y also initiates an ARQ/ACF sequence with Gatekeeper C, providing Endpoint B's address. Then, Gateway Y sends a SETUP message to endpoint B. After a successful ARQ/ACF exchange with Gatekeeper C, Endpoint B responds with a CONNECT message to Gateway Y. Gateway Y then translates that message into a CONNECT message to Gateway X. Gateway X, in turn, sends a CONNECT message to Endpoint A.

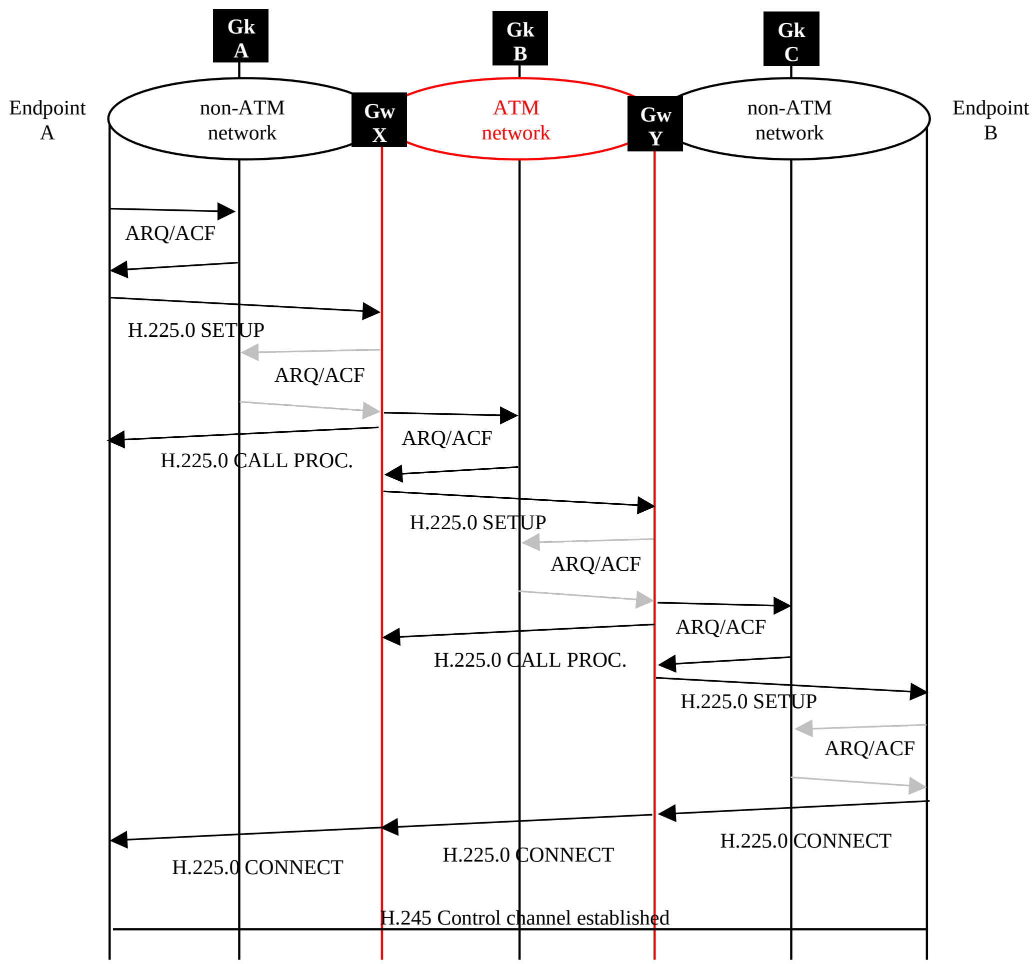

Figure 9 Control sequence to establish the H.245 control channel

At this point, the H.245 control channel is established and the control messages are explicitly addressed to the gateways. The gateways use the information transported in the **OpenLogicalChannel** and **OpenLogicalChannelAck** messages to establish a bi-directional SVC to carry the media stream.

The **OpenLogicalChannel** and **OpenLogicalChannelAck** are generated by the endpoints, exactly as if there were no intervening ATM network; the endpoints are not aware that ATM SVCs are set-up unless an endpoint itself is an ATM endpoint, in which case the function of the endpoint and the gateway are co-located. This approach requires no modifications to H.323 terminals attached to the non-ATM networks.

2.1.2.1 Bi-directional SVC set-up

Section 2.1.2 describes how the H.245 control channel is established through the gateways as intermediary processing points. Once this phase is complete, logical channels can be opened for the transport of media streams, following the Capability Exchange and the Master/Slave Exchange phases. Endpoint A sends an **OpenLogicalChannel** message, containing the transport address for the reverse RTCP channel to Gateway X. (Let us call the A-to-B channel the forward channel). Gateway X receives this message and sends an **OpenLogicalChannel** to Gateway Y that, in turn, sends an **OpenLogicalChannel** message to Endpoint B.

For bi-directional calls, it is necessary to define a rule to determine which gateway is responsible for setting up the SVC. The goal is to avoid a deadlock scenario that would arise if each gateway considers itself the initiating gateway for the connection. It is logical that the side that initiates the H.323 call also initiates the SVC call set up. In the configuration of Figure 7, it will thus always be Gateway X that shall initiate the SVC setup when Endpoint A initiates the H.323 call. The rest of this section illustrates the case when the **OpenLogicalChannel** message is only sent by Endpoint B after it receives an **OpenLogicalChannel** message from Endpoint A: this is merely an example since the **OpenLogicalChannel** messages can happen in any order.

Endpoint B then replies to Gateway Y with an **OpenLogicalChannelAck** message containing the transport addresses for the forward. RTP and RTCP channels. If the reverse channel is to be opened for the reverse direction, Endpoint B also sends an **OpenLogicalChannel** message to Gateway Y. That message contains the same RTCP transport address that was sent in the **OpenLogicalChannelAck** message. In this way, Gateway Y knows which transport addresses to use when forwarding the media and RTCP streams to Endpoint B. Gateway Y also forwards both messages to Gateway X while providing its ATM address in **OpenLogicalChannelAck** message.

On arrival of the **OpenLogicalChannelAck** message, Gateway X can set up a bi-directional SVC using the ATM address provided in this message. For the forward direction, it can request the necessary bandwidth to support the voice encoding standard selected by the **OpenLogicalChannelAck** message sent from Gateway Y. However, a possible associated **OpenLogicalChannel** message sent from Gateway Y may propose different voice encoders for the reverse direction and the establishment of an asymmetric SVC is appropriate. Therefore, Gateway X should delay the transmission of an **OpenLogicalChannelAck** message to Endpoint A until it receives the **OpenLogicalChannel** message from Gateway Y.

Since this **OpenLogicalChannel** message may propose different encoders, Gateway X needs to forward this message to Endpoint A and to further wait for an **OpenLogicalChannelAck** message from Endpoint A that selects an encoder from the proposed options. When this happens, Gateway X has all the information needed and it establishes the appropriate bi-directional SVC. However, since a gateway may terminate many such SVCs to other gateways, the Q.2931 SETUP message carries the port number for the forward RTP media stream in a Generic Identifier Transport information element. The forward RTP port number is derived from the RTCP port number for the forward direction sent in both the OpenLogicalChannel and the OpenLogicalChannelAck messages. The field in the OpenLogicalChannelAck messages that would carry the RTP port number actually carries an ATM address (refer to 2.2). The forward RTP port number is used by Gateway Y to associate the incoming SVC call with the appropriate RTP channel.

When the SVC setup is completed, Gateway X sends the delayed **OpenLogicalChannelAck** message from Gateway Y to Endpoint A, thus establishing the forward RTP media channel after the SVC is set up. Gateway X also forwards the **OpenLogicalChannelAck** message from Endpoint A, which contains the same transport address (for the RTCP stream) used for the forward direction. When this message reaches Gateway Y, it associates the RTP transport address with the pre-established SVC. Finally, when this message reaches Endpoint B, the reverse RTP media channel is established. Figure 10 illustrates the establishment of such bi-directional SVC.

The above scenario assumes that the logical channels will be opened in both the forward and the reverse direction, which is a typical case in the point-to-point voice call. However, to account for the logical channel not being opened in the reverse direction, the initiating gateway (Gateway X) needs to set a timer upon receiving the **OpenLogicalChannelAck**. The value of the timer is implementation specific. If this timer expires, the initiating gateway (Gateway X) shall proceed to set up a unidirectional SVC for the forward media stream (see 2.1.2.2).

There is a possibility that the **OpenLogicalChannel** in the reverse direction has been sufficiently delayed such that it is received by Gateway X after the timer expiry. In that case, Gateway Y becomes responsible for the SVC establishment in the reverse direction. Gateway Y will have to examine the reverse traffic descriptor in the Q.2931 SETUP message from Gateway X. If this traffic descriptor is not sufficient to accommodate the reverse media traffic, upon reception of the reverse **OpenLogicalChannelAck** from Gateway X, Gateway Y shall establish the reverse SVC. Note that this message must carry the ATM address of Gateway X.

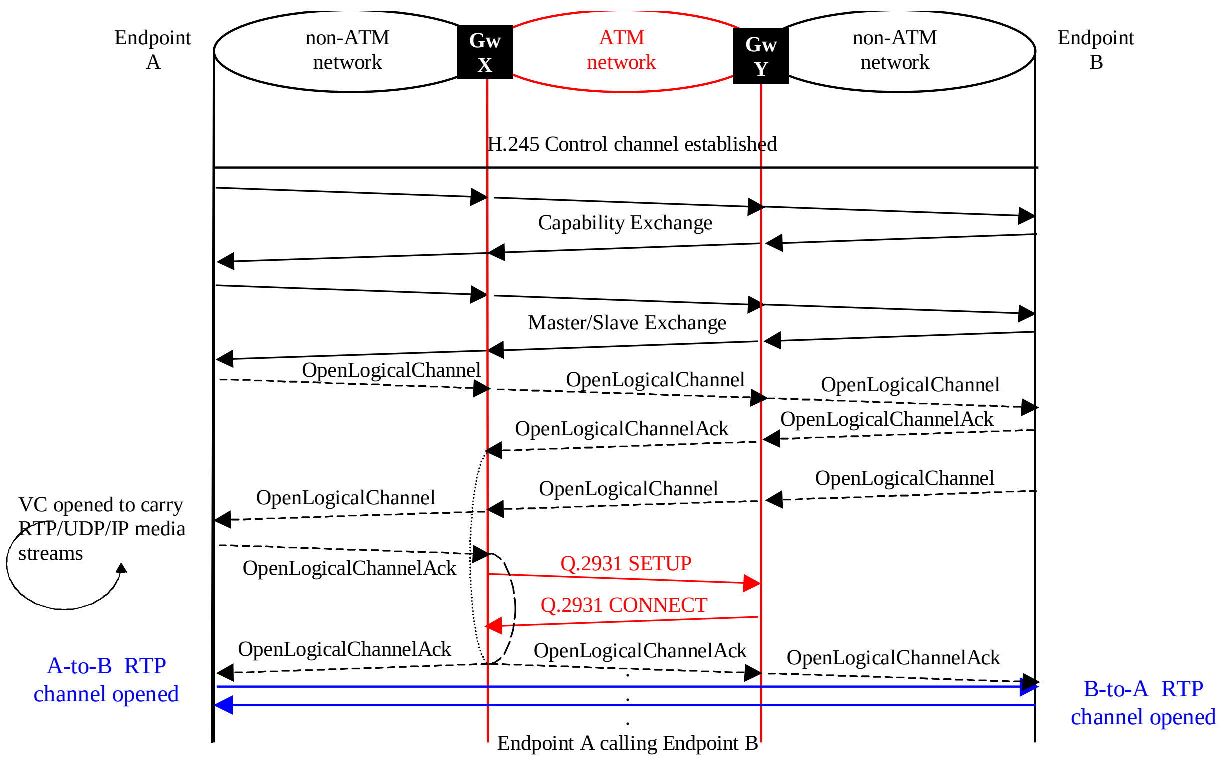

Figure 10 Bi-directional SVC set-up without fast connect

2.1.2.2 Unidirectional SVC set-up

Although a bi-directional SVC setup is preferable when possible, it is necessary to have procedures to setup unidirectional SVCs.

The gateway sets up a unidirectional SVC upon receipt of an **OpenLogicalChannelAck**. It shall then delay the forwarding of the **OpenLogicalChannelAck** to the other gateway until the SVC is set-up.

Figure 11 illustrates an example of setting-up a bi-directional communication for this scenario.

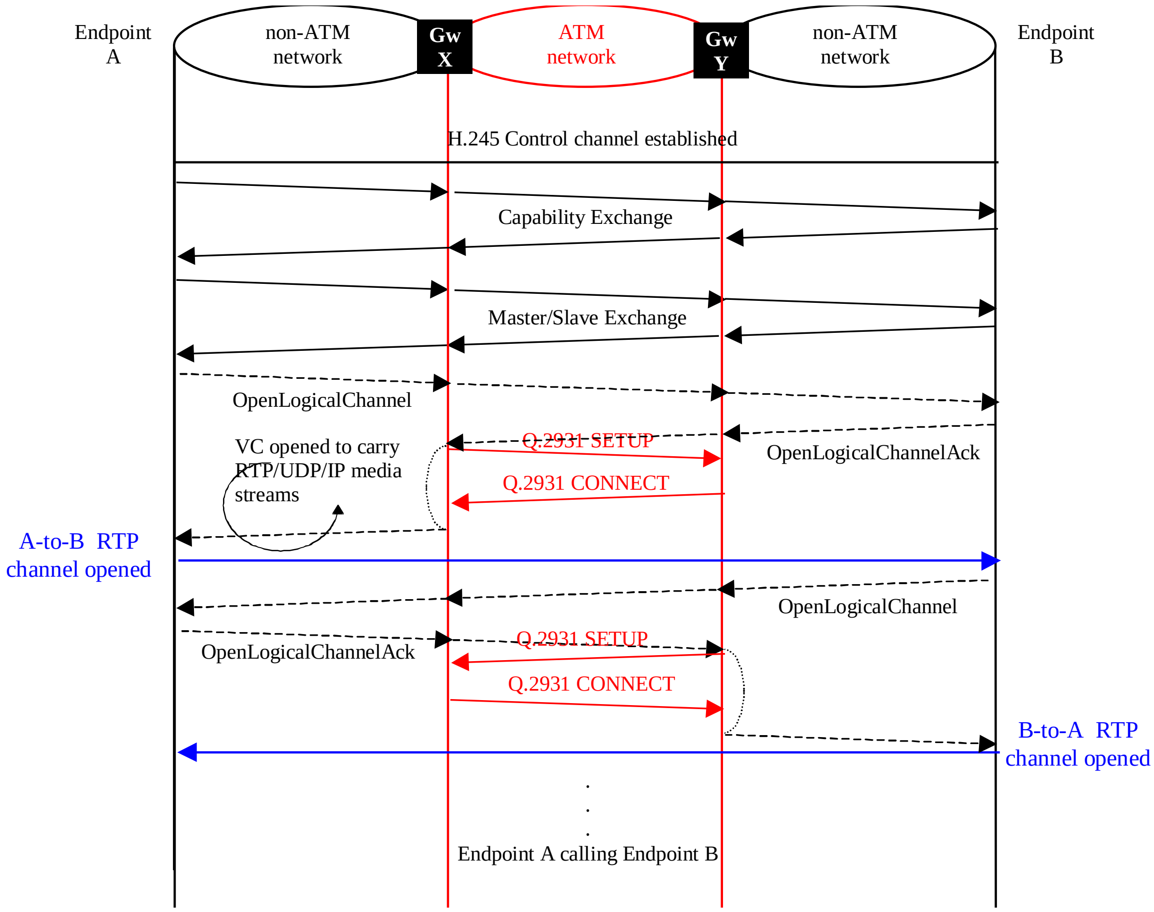

Figure 11 Unidirectional SVC set-up without fast connect

2.2 Termination of media streams at the gateways

The purpose of three gatekeepers in the reference scenario of Figure 7 is to implement control and media termination on the gateways. With control termination, the H.225.0 and the H.245 messages are intercepted by the gateways in order to establish an SVC for each media flow. With media termination, the RTP packets are directed to local interfaces on the gateways to simplify forwarding over the SVCs e.g., by preventing the gateways from having to inspect all IP traffic crossing them. Section 2.1 shows how SVCs are established based on the interception of control traffic.

Figure 12 illustrates the scenario in which RTP channels are established from Endpoint A to Endpoint B using the H.245 procedure. Similar is the case for the establishment of reverse channels also using H.245. Finally, a counterpart scenario can also be described using the **fastStart** element when the Fast Connect procedure is used. For simplicity, only one scenario is described.

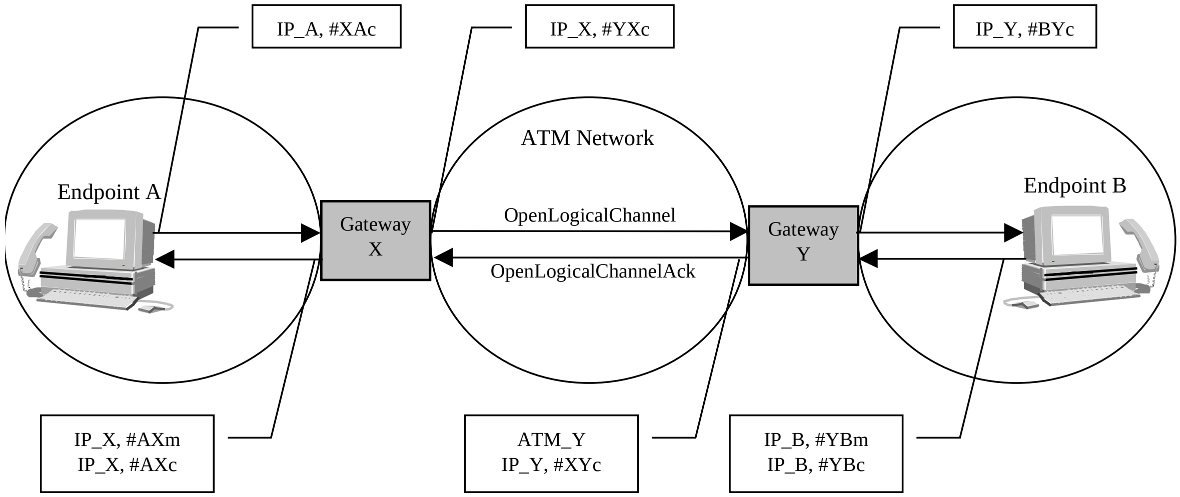

Figure 12 Media termination on gateways

Note: Within this specification, the terms “RTP port” and “UDP port” refer to the destination UDP port. Refer to RFC 768 for usage of the UDP source port.

The **OpenLogicalChannel** messages are terminated on each leg of the call. Similarly, the **OpenLogicalChannelAck** messages are also terminated on each leg. In common with both flows is that the respective transport addresses carried on these messages are replaced when moving across each leg boundary. The new transport addresses have local significance (note that the IP addresses refer to the boundary entities), and each entity on the end-to-end path chooses its own set of UDP port numbers. For further illustrative purposes, the port numbers in Figure 12 are designated by a three-letter sequence: the first two describe the direction of the flow and the latter the nature of the traffic using the port number, c for RTCP control and m for RTP media. For instance the destination port number #AXm is used to identify the forward RTP flow from A to X.

The consequence of this address translation on crossing each boundary is that the traffic sent from endpoint A uses different sets of transport addresses on each segment, effectively breaking the end-to-end call into three legs as illustrated in Figure 13. Furthermore, since it is the gateway that implements the change on the transport addresses, it can easily maintain the mapping for the addresses used in each side. Namely, Gateway X can inspect the transport addresses on incoming packets and replace them with the appropriate one for the SVC leg, and vice-versa. Since header compression is applied on the traffic sent over the SVC, there is no need for the IP and UDP headers and they are not sent on the SVC leg of the call. Nevertheless, the media flow received over an SVC needs to be mapped onto a set of transport addresses to be used on the last leg of the call, even if the set of addresses on the SVC leg is not available.

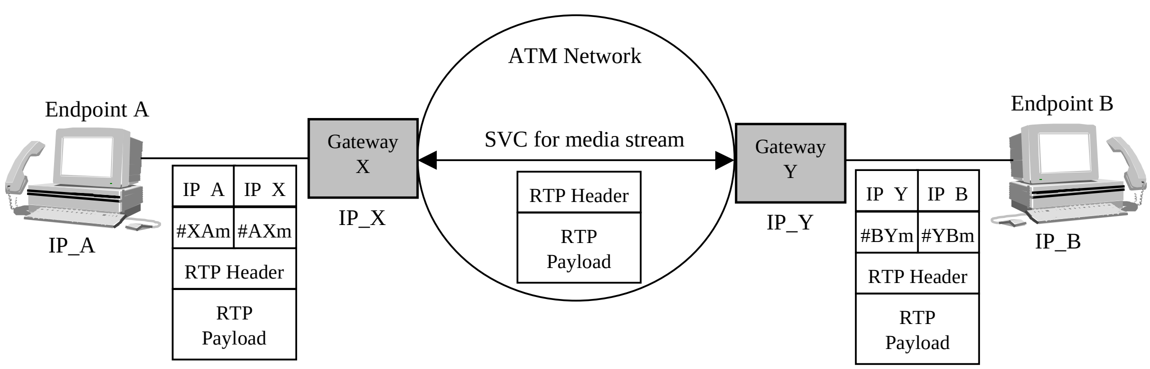

Figure 13 Efficient transport of media stream on ATM SVC

More generally, once the SVC is established, the RTP packets need to be forwarded over the SVC and the mapping of a transport address into an SVC is needed. Similarly, once an RTP packet is received from an SVC, a mapping is needed for the set of transport addresses to use on the following leg of the call. In addition, the gateways may be in the process of establishing many connections with many other gateways and it is necessary to determine which SVC set up request to associate with which outstanding RTP flow.

The figures above illustrate how the transport addresses are replaced at each leg boundary. In addition to the address mapping, an SVC needs to be established using the ATM address of Gateway Y provided in the **OpenLogicalChannelAck**. Gateway X will delay the **OpenLogicalChannelAck** from Gateway Y and wait for the reverse **OpenLogicalChannel** from Gateway Y. This message tries to open the reverse direction of the same call, and therefore it carries the same transport address sent on the **OpenLogicalChannelAck** used for the RTCP channel, i.e., #XYc. When this message comes, it is sent to endpoint A and GW X further waits for the **OpenLogicalChannelAck** from A. It is only then that the SVC can be established.

At this point, Gateway Y knows three transport addresses (see Figure 12): the two for the forward direction (A to B) and the one for the RTCP for the reverse direction, all of which are significant only to the leg over the SVC. The three transport addresses uniquely identify the bi-direction RTP flow for Gateway Y. Hence, it suffices that Gateway X sends one of them on the Q.2931 SETUP message for Gateway Y to associate the incoming SVC set up request with the bi-directional RTP flow to be carried over the SVC. For simplicity the transport address for the forward media stream is sent, since this would be the transport address used on the address mapping if it were sent over the SVC.

The gateways correlate the transport addresses on adjacent legs of the same RTP flow. When an SVC is setup between gateways, the port number for the forward RTP flow is used as a correlation identifier (H.245 **portNumber**) in the Generic Identifier Transport information element to allow the terminating gateway to map the RTP flows to the appropriate SVC.

Therefore, upon receiving this Q.2931 message, the newly provided VCI/VPI is associated by Gateway Y with the respective transport addresses for the last leg and vice-versa. When the Q.2931 CONNECT is received by Gateway X, it associates the newly assigned VCI/VPI with its local set of transport addresses for the first leg of the call.

In this way, an incoming RTP flow is mapped into a given interface #/VPI/VCI and an outgoing RTP flow in Gateway Y is mapped into an RTP flow with the proper source and destination transport addresses.

2.3 Selecting the Traffic Descriptors

It is important to set the proper cell rate parameters in the ATM Traffic Descriptor information element. The peak cell rate can be derived from H.245 and the RTP payload format packet size. For video, the **maxBitRate** field can be used from the **H261VideoCapability** or the **H263VideoCapability** to determine the ATM Cell rate. For audio, the audio capability chosen implies the bit rate to be used. For example, the use of **g711Ulaw64k** suggests the use of a 64 kbit/s audio channel, while the use of **g728** indicates the use of a 16 kbit/s channel. The RTP payload format indicates the packet size. For each packet, the subsequent AAL5 CPCS-PDU overhead and any needed padding to meet the packetization rules, must be added. This results in an overhead bit rate that is associated with the size of the packet and the way this packet is encapsulated in AAL5, and the frequency of this overhead from this encapsulation.

The bit rate of the data to be sent, and the packetization of the data according to the AAL packetization rules determine the cell rate. The packetization will determine the actual number of cells that must be sent for a given data stream at a given bit rate. See Annex A for guidelines on how to choose packet sizes for audio.

2.4 ATM Signalling

This specification uses UNI Signalling 4.0 or PNNI 1.0 to set-up SVCs for the transport of the RTP media stream. The following describes the coding of the necessary information elements.

2.4.1 Generic Identifier Transport

IE Parameter	Value	Notes
Identifier related standard /application (octet 5)	00001011	Recommendation H.323
Identifier Type (octet 6)	00001011	H.245 portNumber
Identifier length (octet 6.1)	0000 0010	2 octets
Identifier value (octets 6.2-6.3)	H.245 portNumber	16-bit binary coded forward H.245 portNumber

Table 1 Generic Identifier Transport

2.4.2 ATM Adaptation layer parameters

IE Parameter	Value	Notes
AAL type (octet 5)	'0000 0101'	AAL5
Forward maximum AAL-5 CPCS-SDU size (octets 6.1-6.2)	MTU size	This value reflects the combination of the maximum RTP payload plus the maximum RTP headers
Backward maximum AAL-5 CPCS-SDU size (octets 7.1-7.2)	MTU size	This value reflects the combination of the maximum RTP payload plus the maximum RTP headers
SSCS type (octet 8.1)	'0000 0000'	Null SSCS

Table 2 ATM Adaptation Layer Parameters

2.4.3 ATM Broadband bearer capability

a) using CBR

IE Parameter	**Value**	**Notes**
Bearer class	BCOB-A	
Susceptibility to clipping	Susceptible to clipping	
User-plane connection configuration	point-to-point	

Table 3 ATM Broadband bearer capability for CBR

b) Using real time VBR with CLR commitment on CLP=0+1

IE Parameter	Value	Notes
Bearer class	BCOB-X	
Broadband bearer capability	'0010011'	Real time VBR with CLR commitment on CLP=0+1
Susceptibility to clipping	Susceptible to clipping	
User-plane connection configuration	point-to-point	

Table 4 ATM Broadband bearer capability for rt-VBR with commitment on CLP = 0+1

2.5 H.245 transport capabilities

The **transportCapability.mediaChannelCapabilities.mediaTransport.mediaTransportType** field is used for H.245 capability negotiation in the **h2250Capability** field. It is then selected in the **h2250LogicalChannelParameters** of the

OpenLogicalChannel. With fast start, it is only included in the **h2250LogicalChannelParameters** of the **OpenLogicalChannel**.

For operation of H.323 over ATM, the **mediaTransportType** field shall be coded as **atm-AAL5-compressed** when header compression is used, unless a bilateral agreement on always using header compression exist between the gateways: this may be necessary if the H.245 version in use does not support the **atm-AAL-compressed** codepoint. See note below. As explained in section 3.2.2.3, an indication of which type of delta encoding is used (variable or fixed 2-octet) is required. The **variable-delta** field is set to true in the **atm-AAL5-compressed** field for the variable delta encoding, and set to false for the default fixed-length delta field.

*Note - It is expected that H.245 version 6 will support the **atm-AAL5-compressed** field in H.245, as described below. Earlier versions of H.245 do not support negotiation of header compression.*

```
MediaTransportType ::=CHOICE
{
     ip-UDP                 NULL,
     ip-TCP                 NULL,
     atm-AAL5-UNIDIR        NULL,
     atm-AAL5-BIDIR         NULL,
     atm-AAL5-compressed    SEQUENCE
     {
             variable-delta BOOLEAN ,
             ...
     },
     …
}
```

If header compression is not used, the **mediaTransportType** shall be coded as **atm-AAL5-UNIDIR** for uni-directional logical channels, or **atm-AAL5-BIDIR** for bi-directional logical channels such as in H.323/Annex C.

The ATM transfer capability shall be indicated in the **transportCapability.qOSCapabilities.atmParameters**.

3 Compressed RTP media stream transport over ATM

This section describes an AAL5 encapsulation for carrying real-time streams, such as audio and video, over ATM. The encapsulation function for AAL5 for real-time media streams sits just above the AAL5 adaptation layer, and thus uses existing ATM and AAL5 layers without modification. This allows AAL5 for real-time media streams to be deployed on existing hardware implementations without modification.

Real-time streams often need to communicate data other than the real-time samples. For example, end-end application information, such as inter-media synchronization and time stamps, must often be exchanged between end systems. AAL5 for real-time media streams provides a framework for such an information exchange through an extension mechanism in the UUI field allowing additional information to be embedded in the AAL5 frame along with the real-time samples. This provides the desired flexibility by allowing application specific information to be carried over the same virtual circuit as real-time samples.

Utilizing the framework provided by AAL5 for real-time media streams, this section describes a method for efficiently carrying RTP packets over an ATM network. This is done by treating an ATM virtual circuit as a point-to-point link and employing RTP header compression. An ATM compressed RTP header is combined with real-time samples into an AAL5 frame.

The goal of this method is to transport real-time data over an ATM network as efficiently as possible, recognizing that the real-time stream may be generated from an ATM end-system, or it may more generally be generated from a computer system using an RTP/UDP/IP protocol stack.

The big gain in this method comes from the observation that although several fields change in every packet, the difference from packet to packet is often constant and therefore the second order difference is zero. By maintaining both the uncompressed header and the first order differences in the session state shared between the compressor and the decompressor, all that must be communicated is an indication that the second order difference was zero. In that case, the decompressor can reconstruct the original header without any loss of information simply by adding the first order differences to the saved uncompressed header as each compressed packet is received.

Section 1.5.2 shows that for RTP over ATM, the typical (and minimum) overhead amounts to 12 octets. Compression of the headers occurs at the edges of the ATM network as shown below. The compressor/decompressor (C/D) function is thus part of the Media Gateway.

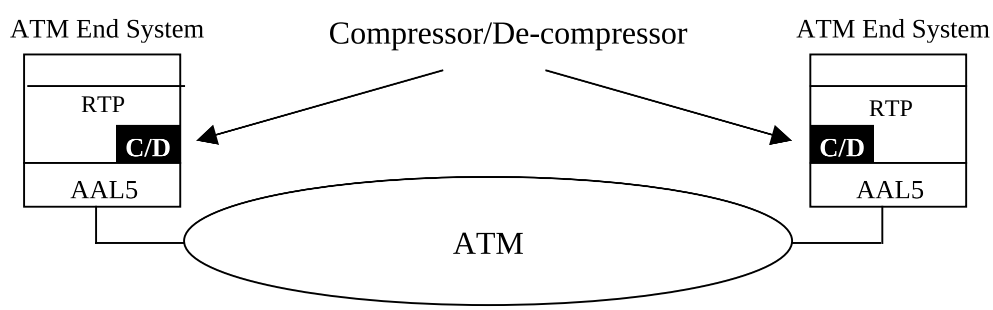

Figure 14 Compressor/De-compressor in an ATM End System

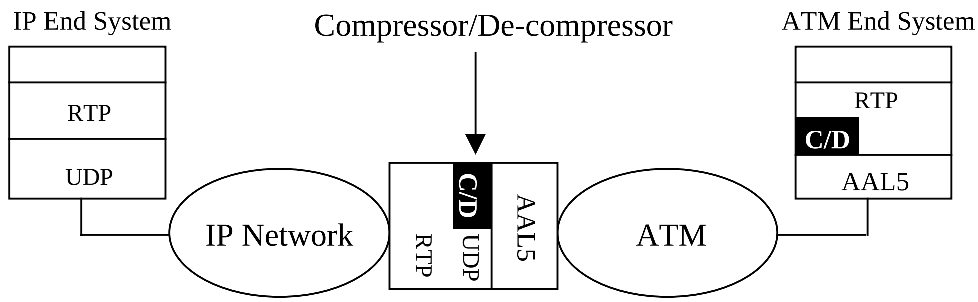

Figure 15 Compressor/De-compressor in an IP/ATM Router

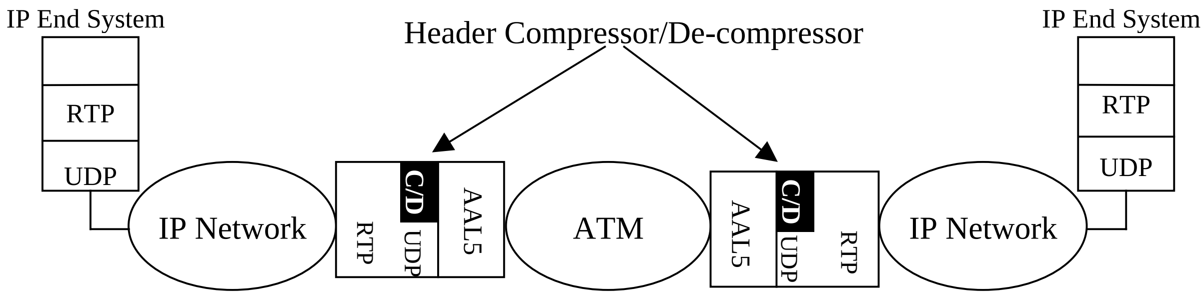

Figure 16 Carrying Compressed RTP/IP/UDP in an ATM backbone

Section 3.1 describes AAL5 for real-time media streams, and section 3.2 describes RTP header compression format which is an application of AAL5 for real-time media streams.

3.1 AAL5 CPCS-PDU for real-time media streams

The format for an AAL5 CPCS-PDU is shown in Figure 17. AAL5 CPCS-PDU makes use of the 8-bit user-to-user field (CPCS-UU) of I.363.5. The first bit is a “control bit”, the second bit is a "type bit" and the other 6 bits are the sequence number used to maintain compressor and decompressor synchronization. When applicable, the 6-bit sequence number is incremented with each AAL5 frame irrespective of the state of the type and control bits.

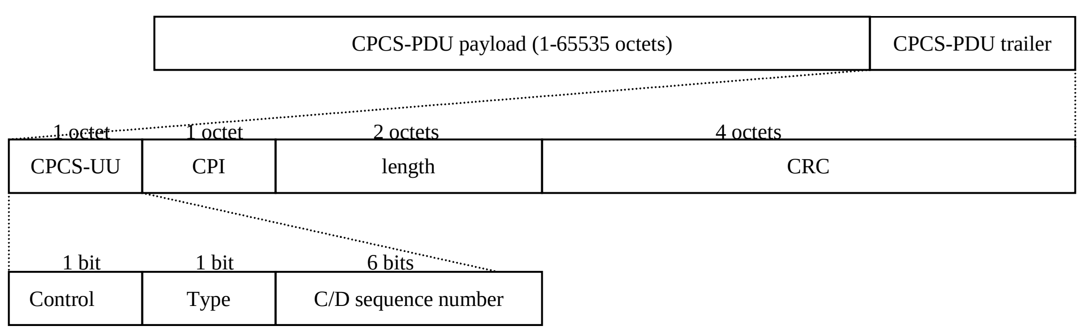

Figure 17 Format of AAL5 for real-time media streams CPCS-PDU

The control bit allows application specific information to be included along with real-time samples. As shown in Figure 18, when the control bit is set to 0, the AAL5 payload consists of real-time samples, possibly some compression information and necessary padding. The format of the compression information field is specified in section 3.2, while its presence is determined by the type bit (T). When the control bit is set to 1, a control octet field is added to the payload before the real-time samples, as shown in Figure 19. The payload consists of a real-time sample, and necessary padding.

Compression information	Real-time sample and padding	**0**	T	sequence #	CPI	length	CRC

Figure 18 AAL5 Encapsulation (Control bit= 0)

Control byte	Compression Information, Real-time sample and padding	**1**	X	sequence #	CPI	length	CRC

Figure 19 AAL5 Encapsulation (Control bit= 1)

Note - The 6-bit sequence number does not imply that the decoder is required to buffer 64 packets.

3.2 RTP header compression format on ATM

The Internet Engineering Task Force (IETF) has been working on header compression [28], in particular on compressing the RTP/UDP/IP headers [27] in the context of a point-to-point PPP link [29]. Since a virtual circuit in an ATM network looks similar to a point-to-point link, the same approach may be used in ATM networks. This has the advantage of not wasting ATM network bandwidth by carrying unnecessary information, and recovers a substantial part of the cell tax paid when carrying packets over an ATM network. This section describes how to carry RTP packets in an ATM network using AAL5 for real-time media streams.

For information, typical RTP, UDP and IP headers are shown in Figure 20.

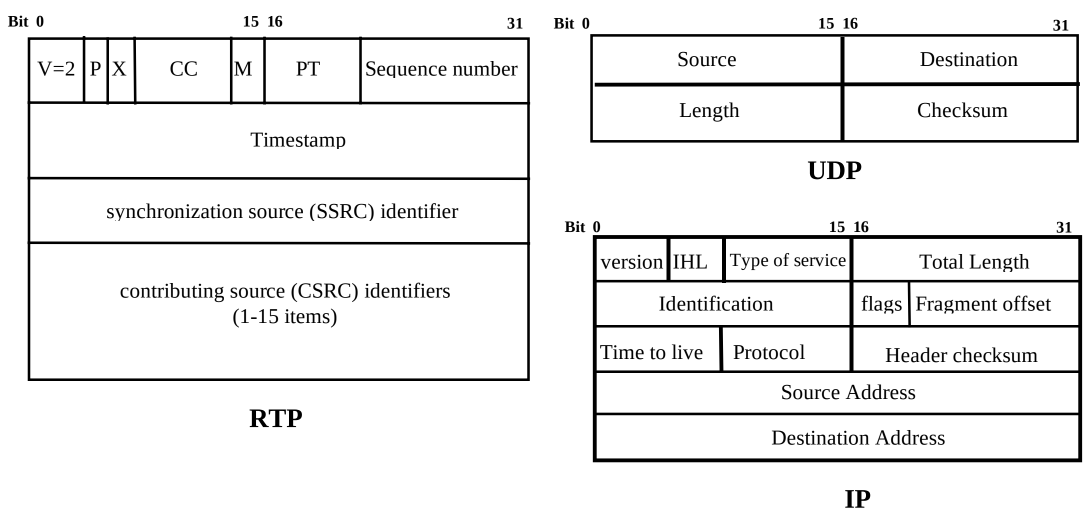

Figure 20 RTP, UDP and IP headers

The signalling architecture in section 2 allows an H.323-H.323 gateway to terminate H.323 media streams.

From the signalling procedures of section 2, an SVC is established between the H.323-H.323 Gateways X and Y for the transport of media stream between Endpoints A and B. A gateway then forwards the media packets received from an endpoint to the other gateway over this SVC. However, most of the IP and UDP information on packets (e.g., IP addresses and UDP port numbers) reaching a gateway has no meaning to the other Endpoint-Gateway communication and it need not be sent. Therefore, the only pertinent information to be transferred over the SVC is the RTP header and payload, as shown in Figure 1. Header compression can then be applied only to the RTP header. For this reason, RTP-only header compression shall be used when the media streams are terminated at an H.323-to-H.323 gateway in the core of an ATM network instead of RTP/UDP/IP header compression. Since the UDP and IP layers are terminated at the gateways, IP fragments shall also be re-assembled at the gateways. For applications that are delay sensitive such as voice, this should not be a problem since the packets are likely to be very small.

3.2.1 Definition of the call context

Even though header compression has been originally conceived for PPP links [27], it is also attractive when shipping RTP traffic over ATM SVCs. However, the requirements on header compression on H.323-H.323 gateways are more stringent because a gateway has to maintain contexts, implement the signaling functionality in section 2, and route each stream over the appropriate SVC. The use of RTP-only compression presented here reduces the memory requirements to maintain a call context. It would consist only of the last RTP header sent/received, the first order differences needed to reconstruct an RTP header from a compressed packet and a sequence number space to detect packet losses.

The fields in the RTP header that may require the 1st order difference in the call context are those that can change on a packet-by-packet basis. These are the M bit, the sequence number, the time stamp, the CC count and the CSRC list.

- The M bit marks special events for the real-time application using RTP and it may change frequently; therefore the M bit for a packet can always be communicated, and is not part of the call context.
- The RTP sequence number is expected to always increase by one and the 1st order difference is not included in the call context. The RTP sequence number is not correlated to the C/D sequence number in the AAL5-UUI field.
- Since the variations on the RTP time stamp can change over the duration of a call, the 1st order difference for this field is included in the call context.
- Changes to the CSRC list and CC count need to be communicated explicitly, hence 1st order differences need not be maintained. Figure 21 summarizes how the compressor and the de-compressor handle each of the RTP header fields.

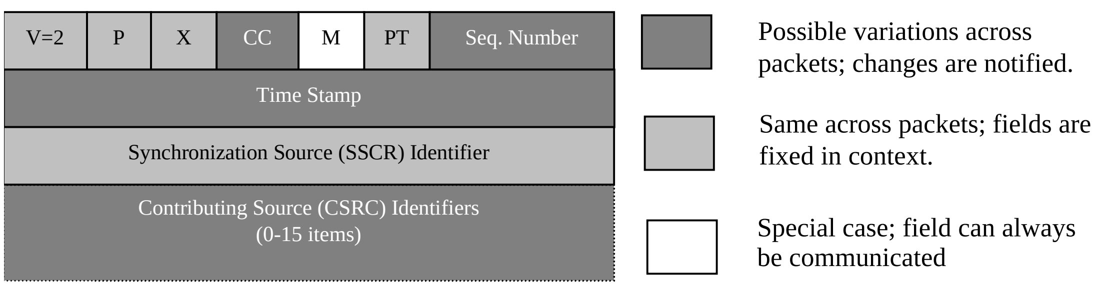

Figure 21 Compression effect on RTP header

The RTP header is the only header stored on the call context and memory requirements are further reduced by not including a table for the encoding of differences for each call, as suggested in [27]. Nevertheless, as part of a C/D negotiation phase, both C/D ends need to agree on whether to use the encoding table in [27] or an encoding restricted to only two octets per field. Since the IP and UDP layers are terminated at the gateways, the UDP checksum, when present, shall be set to zero by the decompressor.

With these simplifications, the call context for the RTP header compression is defined as illustrated in Figure 22. Note that the use of a generation field [27] is not considered, while the E flag is added. The E flag signals whether the encoding table in [27] is used to encode 1st order differences.

RTP Header
First order difference for the RTP time stamp field
C/D Sequence number value and E flag

Figure 22 Call context for RTP header compression

Similarly, to the IETF header compression approaches in [27], the RTP-only compression relies on the AAL5 layer being able to provide an indication of the different packet formats below, as the PPP layer does for the IETF approach. Whenever it is present in the original packet, the RTP header extension is preserved.

- FULL_RTP_HEADER - communicates a full RTP header to establish or synchronize the state in the de-compressor for a call context. Unlike the full IP/UDP/RTP header compression over PPP links [27], the RTP-only compression over SVCs does not require a context identifier (CID). Namely, in a PPP link, the C/Ds on both ends need to maintain context for many IP flows traversing the same link and the CIDs are used to determine the context in which a packet has to be considered. Since each direction (forward and backward) of the SVCs is dedicated to a single one-way RTP flow, the context is defined by the VCI/VPI of the SVC over which the packets are sent. A sequence number is still needed to establish the synchronization between the compressor and the de-compressor.

- CONTEXT_STATE – is a special packet sent from the de-compressor to the compressor indicating that the context associated with the SVC may have been invalidated. The compressor is expected to send the next packet as a FULL_RTP_HEADER packet.

- DELTA_RTP - communicates the new 1st order differences for the RTP header fields indicated since they incurred a non-zero 2nd order difference. The header for this type of packet contains a 4 bit flag field to indicate which fields in the RTP header changed, followed by the respective new 1st order differences. This 4 bit flag field is named the MSTC sequence. The M bit carries the original M bit of the RTP header. The S and the T bits signal the new 1st order differences for the sequence number and the time stamp, respectively. The C bit indicates changes to the RTP CSCR list. Note that the RTP sequence number is expected to always increase by one and therefore if the S bit is set, the de-compressor uses the new delta only once.

- ATM_COMPRESSED_HEADER – indicates that the RTP header has been fully compressed, i.e., all fields that change on a packet basis have actually had a 2nd order difference of zero. Therefore, only the RTP payload, in addition to a possible RTP header extension, is sent over the SVC. The use of a packet type to indicate a fully compressed packet is intended to improve the performance of the common case; it is expected that null 2nd order differences will be frequent.

3.2.2 RTP header compression transport on AAL5

The MSTC bits, the sequence number and the RTP header and payload are the relevant pieces of information needed for RTP-only header compression as illustrated in Figure 23. The sequence number, the RTP payload and, if applicable, the RTP header extension, are sent on all packet types described above, except on CONTEXT_STATE packets. The MSTC bits and possibly a partially compressed RTP header are sent on DELTA_RTP packets; no MSTC bits are sent, however, on a FULL_RTP_HEADER or on an ATM_COMPRESSED_HEADER packet.

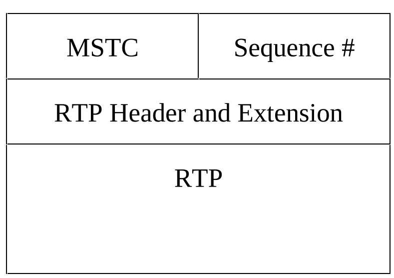

Figure 23 Information required for RTP header compression

Even though some of the information on Figure 23 need only be sent on certain packet types, the uniform real time AAL5 encapsulation of section 3.2.2 is used as illustrated in Figure 24. The C/D sequence number is carried as the six LSBs in the UUI octet of the AAL5 frame. The MSB in the same UUI octet is the control bit and the second MSB is the   type bit. The RTP header (in full or compressed form) and payload are carried along with the MSTC bits in the AAL5 CPCS-PDU.

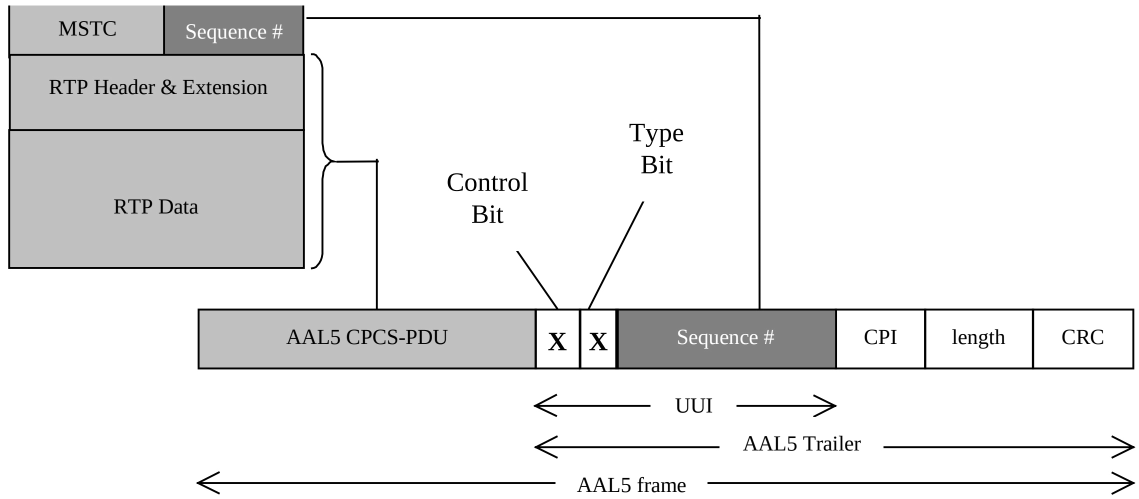

Figure 24 RTP header compression over AAL5

3.2.2.1 FULL_RTP_HEADER packet format

The definition of a FULL_RTP_HEADER packet given here is for the RTP-only compression over AAL5. The control bit is set to '1' indicating that a control octet is present. The control octet is set to '00000000' indicating a FULL_RTP_HEADER packet format. The value for the type bit is irrelevant (X). The AAL5 CPCS-PDU contains a full RTP header, a possible RTP header extension and the RTP payload. The MSTC bits are not sent since they do not apply to a FULL_RTP_HEADER packet.

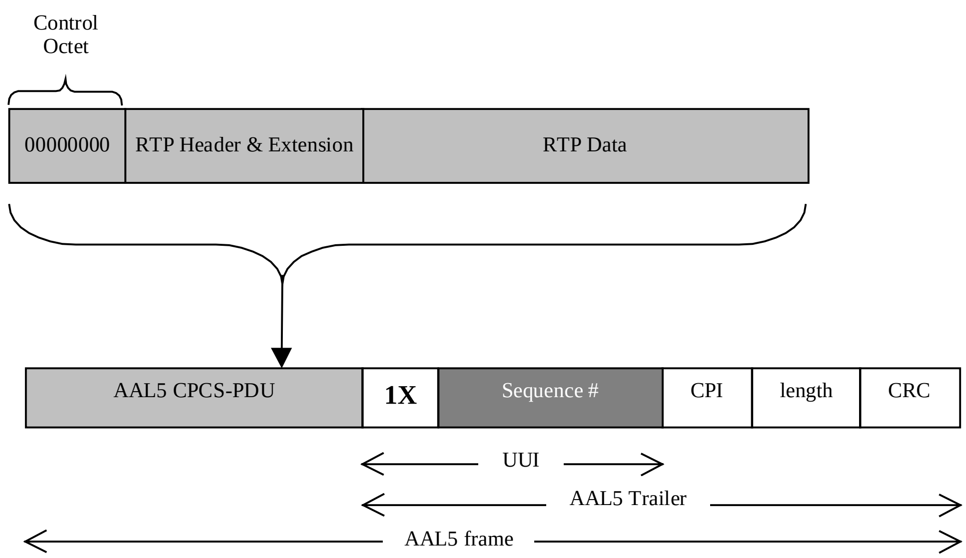

Figure 25 FULL_RTP_HEADER packet

3.2.2.2 CONTEXT_STATE packet format

When an AAL5 frame is discarded, e.g., when a cell from an AAL5 frame is lost, the AAL5 layer may provide an error indication. Since some time may elapse before another frame is received and dropped by the de-compressor (because the sequence number delta is different from 1), the de-compressor will send an advisory CONTEXT_STATE message carrying the last C/D sequence number received over the SVC, upon receiving the error indication. If the compressor C/D sequence number matches the sequence number from the advisory message, no actions are taken. Otherwise, the compressor invalidates its context and the next packet is sent as a FULL_RTP_HEADER packet.

In RTP-only compression, the compressor and the de-compressor communicate with many peers over dedicated SVCs. Therefore, each CONTEX_STATE message is specific to a single compression context indicated by the VPI/VCI of the SVC. As a result, a single octet suffices for a CONTEXT_STATE packet.

The bit labeled “I” is set to one to indicate that the context is invalid in the de-compressor and that a FULL_HEADER packet transmission is required. If the I bit is not set, it indicates an advisory CONTEX_STATE message that would be sent following an AAL5 error indication.

The control bit is set to '1' indicating that a a control octet is present. . The Control octet is set to '00000001' indicating a CONTEXT_STATE packet format The value for the type bit is irrelevant (X). Since this message type does not carry any payload and is sent in the reverse direction, the sequence number on the AAL5 UUI octet is not needed and is set to zero.

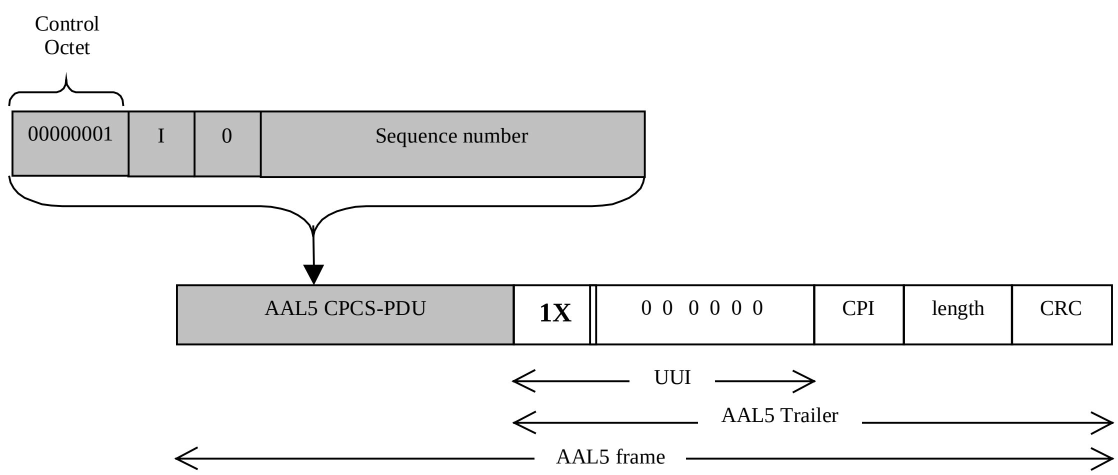

Figure 26 CONTEXT_STATE packet

Similarly to the approach described in [27], whenever the sequence number differs by more than 1, except when set by a FULL_RTP_HEADER packet, this is an indication that a packet was lost. Since synchronization is also lost, the de-compressor invalidates its context and sends a CONTEXT_STATE packet back to the compressor indicating that it has an invalid context. The de-compressor also discards all subsequent packets until a FULL_RTP_HEADER packet is received from the compressor.

Note that multiple DELTA_RTP and ATM_COMPRESSED_HEADER packets may arrive after a CONTEXT_STATE message is sent and before a FULL_RTP_HEADER packet is received. In this case, the de-compressor does not send a CONTEXT_STATE message for every packet discarded due to a sequence number mismatch to avoid flooding the reverse channel. Instead, the rate of CONTEXT_STATE message retransmission will be bounded by a configurable maximum rate.

Nevertheless, the CONTEXT_STATE message itself may be lost and this maximum rate may delay the compressor/de-compressor resynchronization, causing service degradation. Therefore, a configurable timer will also be set by the de-compressor upon transmission of a CONTEXT_STATE message and a new one will be sent when this timer expires if a FULL_HEADER packet was not received.

3.2.2.3 DELTA_RTP packet format

Either when the 2nd order differences from packet to packet are non-zero or when the M bit is set in the original RTP header, a DELTA_RTP packet is sent. (Note that unlike the STC bits that indicate a delta change in their respective fields, the M bit is explicitly carried in the DELTA-RTP packet when the original M bit in the RTP header is 1. The assumption here is that the M bit is 0 for most of the RTP packets. This is consistent with marking of the beginning of the talk spurt or the end of a video frame.) In the former case, this packet type transports the new delta values to be used by the de-compressor in generating the de-compressed packet. The control bit is set to ‘0’ indicating the absence of a control octet The type bit is set to ‘1’ indicating the presence of compression information and that the packet type is thus a DELTA_RTP packet. RTP extension header is also included along with the RTP payload provided the X bit is set in the context.

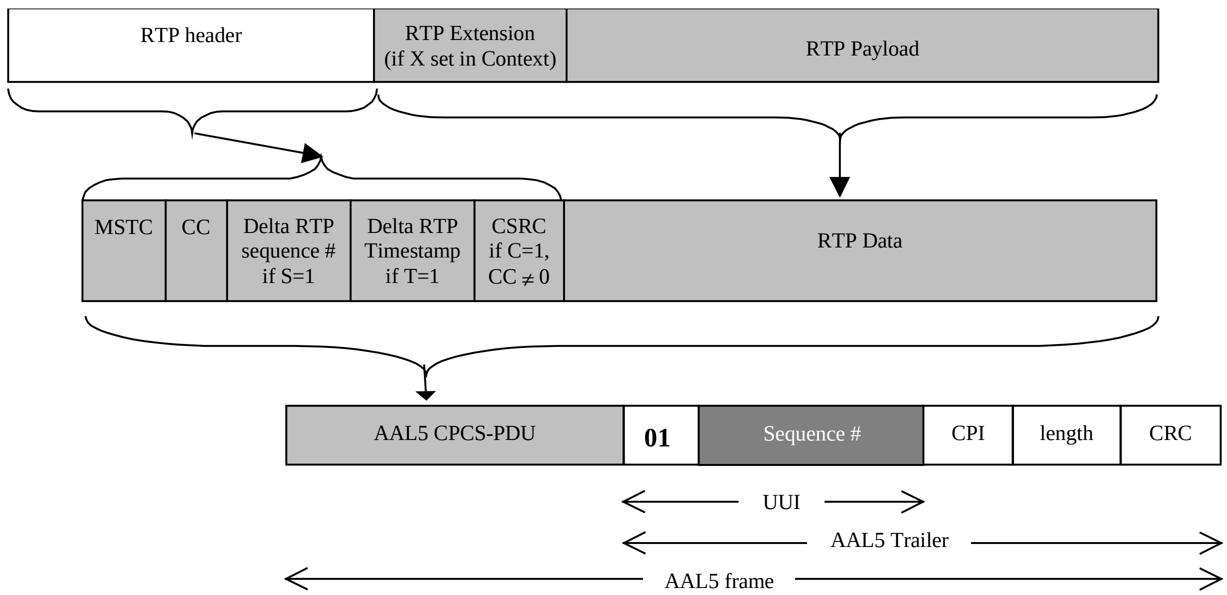

Figure 27 DELTA_RTP packet

The new 1st order differences for the changing fields in the RTP header are computed as the difference between the respective fields on the current RTP header and in the previous RTP header sent; this is stored on the call context. If these differences are encoded using the table presented in [27] (the E flag is set in the context), each 1st order difference is represented using from one up to three octets. Otherwise, the E flag is zero and a fixed two-octet field is used to convey each new difference. If the new 1st order difference cannot be accommodated in the number of octets available in either approach, a FULL_RTP_HEADER packet is sent.

Finally, when the C bit is set in the MSTC sequence, the CC count is transported in the four LSbs of the same octet carrying the MSTC bits.

3.2.2.4 ATM_COMPRESSED_HEADER packet format

An ATM_COMPRESSED_HEADER packet carries the same information that a DELTA_RTP packet with MSTC = 0000 would, i.e., just the RTP payload and a possible RTP header extension. Since having MSTC = 0000 is expected to be the common case, the use of a dedicated packet type is preferred because actions to treat this case can be taken immediately based on the packet type indication, saving one octet of overhead. Namely, rather than having to inspect the AAL5 payload to realize that no new delta values were sent when MSTC = 0000, an ATM_COMPRESSED_RTP packet is used instead of a DELTA_RTP packet.

The control bit is set to '0' indicating that no control octet is present. The type bit is set to '0', indicating the absence of compression information and the packet type to be ATM_COMPRESSED_HEADER. Figure 28 illustrates how an ATM_COMPRESSED_HEADER packet is encapsulated in the AAL5 payload. The RTP header extension is only present on an ATM_COMPRESSED_RTP packet if the previous FULL_RTP_HEADER packet had the X bit set on the RTP header and thus if the X bit is set in the context.

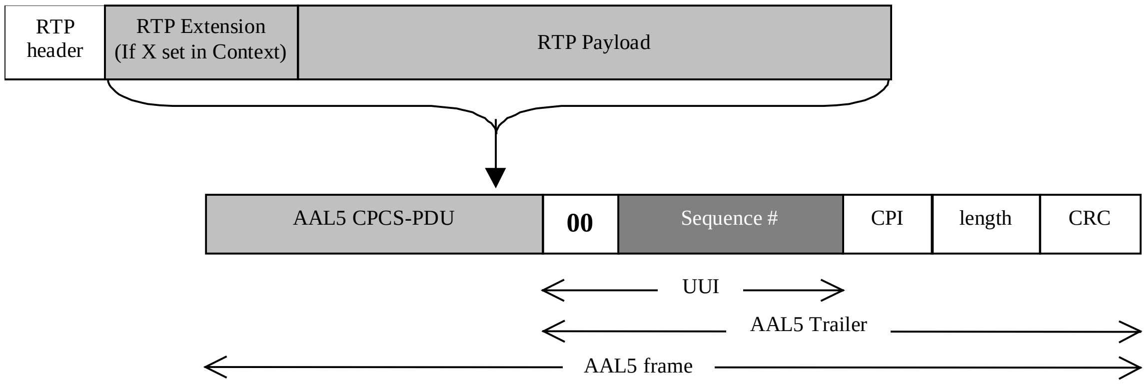

Figure 28 ATM_COMPRESSED_HEADER packet

4 Interworking

Interworking between various multimedia terminals on ATM or on alternative network technologies is of prime importance.

Interworking occurs at many levels:

- Call control (signalling protocol),
- System control,
- Media flow control,
- Addressing,
- Encoding of user plane information.

In addition, some parameters can be chosen to have better interworking. The following apply:

- Transcoding between voice encoding schemes should be avoided when possible because of voice quality degradation (i.e., a common voice or video encoding mechanism should be used).
- Packet size can be chosen in a way such as to facilitate efficient transport on ATM.

Interworking of H.Series multimedia terminals with H.Series multimedia terminals and voice/voiceband terminals on GSTN and ISDN is covered in H.246 [3].

4.1 AAL2 trunking (I.366.2 and AF-VTOA-0113)

Interworking AAL2 trunks with H.323 terminals requires the use of a gateway. The gateway shall remove the RTP headers when mapping from H.323 to AAL2 trunks for packets that are uncompressed RTP packets, and generate the RTP headers when mapping from AAL2 trunks. For the ATM compressed RTP packets, the extension bit and the sequence number in the AAL5 CPCS-UU octet shall be generated.

Interworking between H.323 and AAL2 Trunking should only apply to voice-only calls since video and data packetization utilizes large packets that are not suitable for AAL2. Using AAL2 would not preserve the integrity of the H.323 (H.225.0 and H.245) information end-to-end.

In order to avoid voice quality degradation, the same voice coders should be used on both sides of the gateway.

In order to avoid unnecessary re-packetization delays, the gateway should use the same voice payload size on H.323 and on AAL2 trunks.

4.2 VTOA to the Desktop (AF-VTOA-0083)

Interworking AF-VTOA-0083 terminals with H.323 terminals requires the use of a gateway. Choosing the default voice packet size to be 40 octets, if the AF-VTOA-0083 terminal uses AAL5, will minimize the packetization delay. It is preferable to use AAL5 for AF-VTOA-0083: however, if the af-vtoa-0083 terminal uses AAL1, choosing the default voice packet size to be 47 octet on H.323 will minimize the packetization delay. The gateway shall remove the RTP headers when mapping from H.323 to AF-VTOA-0083 for packets that are uncompressed RTP packets, and generate the RTP headers when mapping from AF-VTOA-0083. For the ATM compressed RTP packets, the extension bit and the sequence number in the AAL5 CPCS-UU octet shall be generated.

In order to avoid transcoding delays and voice quality degradation, it is strongly recommended that the H.323 terminal uses G.711 which is the only voice codec supported by AF-VTOA-0083.

Furthermore, an AF-VTOA-0083 terminal is equivalent to an H.321 terminal establishing a simple telephony connection.

4.3 Native ATM (H.310 and AF-SAA-0049)

Interworking H.310 and AF-SAA-0049 terminals with H.323 terminals requires the use of a gateway. The gateway shall remove the RTP headers when mapping from H.323 to H.310/AF-SAA-0049 for packets that are uncompressed RTP packets, and generate the RTP headers when mapping from H.310/AF-SAA-0049. For the ATM compressed RTP packets, the extension bit and the sequence number in the AAL5 CPCS-UU octet shall be generated.

4.4 H.321 (H.320 over ATM)

An H.321 terminal establishing a simple telephony connection is equivalent to an AF-VTOA-0083 terminal. Therefore, for simple telephony connections, section 4.2 is applicable.

4.5 H.323 Annex C

Since the H.225.0 and H.245 communications are on TCP/IP, the endpoint will be able to receive calls from any other endpoint regardless of if they support H.323/Annex C or compressed RTP over AAL5 (as per section 3).

The H.245 **transportCapabilities** is used by native terminals or H.323-H.323 gateways to indicate which method of transport of is supported: RTP over AAL5 (as per H.323/Annex C) or compressed RTP over AAL5 (as per section 3).

All gateways supporting this specification shall also be capable of supporting H.323/Annex C procedures. This will enable H.323/Annex C mode interworking between a gateway and a H.323/Annex C ATM terminals.

Annex A Guidelines for choosing voice packet sizes (Informative)

This section provides guidelines on choosing maximum packet sizes for the audio codecs supported by H.323: G.711 (PCM), G.722 (SB-ADPCM), G.723.1 (MP-MLQ/ACELP), G.728 (LD-CELP), and G.729 (CS-ACELP). Emphasis is given to the impact of the maximum codec packet sizes on the efficient transport over AAL5 with RTP-only header compression. The reader should keep in mind, however, that other important implications of this choice are end-to-end delay and possible voice quality degradation caused by packet losses. These aspects are not addressed in detail here.

H.323 through H.225.0 suggests that the default packetization interval for the voice traffic should be 20 ms for any voice codec, unless the codec itself cannot accommodate this value (e.g., for G.723.1 which has a voice frame size of 30 ms by definition). These default values mean that the default voice packet size (in octets) varies for every voice codec. However, H.323 allows for negotiation of the default voice packet size and this Annex investigates the improvements on bandwidth utilization if a packetization interval more appropriate for ATM transport is used. Table 5 shows the different packet sizes for the H.323-supported codecs for the default 20ms (or 30ms for G.723.1) packetization interval.

	G.711 PCM	**G.722** SB-ADPCM	**G.723.1** MP-MLQ/ACELP		**G.728** LD-CELP	**G.729** CS-ACELP
Rate (kbit/s)	64	64	6.3	5.3	16	8
Frame size (ms)	1	1	30	30	2.5	10
Frame size (octet)	8	8	24	20	5	10
Default packet size (ms)	20	20	30	30	20	20
Default Frames per packet	20	20	1	1	8	2
Default payload size (octet)	160	160	24	20	40	20

Table 5 H.323 default voice packet sizes

Each codec packet is then appended to an RTP header, incurring a minimum of 12 octets of overhead. When H.323-H.323 gateways communicate over an ATM network, these RTP packets are carried directly over AAL5, adding at least another 8 octets of overhead due to the AAL5 trailer. The AAL5 payload can be further padded with unused octets to make the AAL5 frame an integral multiple of 48 bytes for segmentation into ATM cells. Naturally, the H.323 default IP over ATM transport incurs in more protocol overhead but this case will not be addressed here since it is outside the scope of this specification.

Table 6 illustrates the resulting cell rates when the H.323 default packet sizes are used on RTP packets carried directly over AAL5. Note that these computations involve the minimum possible overhead since it refers to the minimum RTP header.

	G.711	**G.722**	**G.723.1**		**G.728**	**G.729**
Rate (kbit/s)	64	64	6.3	5.3	16	8
Frame size (ms)	1	1	30	30	2.5	10
Frame size (octet)	8	8	24	20	5	10
Default packet size (ms)	20	20	30	30	20	20
Default Frames per packet	20	20	1	1	8	2
Default payload size	160	160	24	20	40	20
Default packet size (+overhead)	172	172	36	32	52	32
Default packet size (cells)	4	4	1	1	2	1
Unused octets	12	12	4	8	36	8
Cell Rate (cell/s)	200	200	33.4	33.4	100	50

Table 6 Cell rates for RTP over ATM voice transport using H.323 default packetization.

The use of RTP-only compression implies that in most cases, the RTP overhead is eliminated, making the method just as efficient as Native ATM media stream transport. Table 7 shows the resulting cell rates in steady state when the RTP headers are fully compressed. These cell rates would be the minimum rates (excluding the CONTEXT_STATE effect) when RTP-only header compression is used. Comparing Table 6 and Table 7, it is clear that except for the G.728 codec, the use of RTP-only compression leads to no reduction in the required bandwidth to carry the RTP streams. Furthermore, for G.728 the DELTA_RTP packets would require the same number of cells as FULL_HEADER packets.

	G.711	G.722	G.723.1		G.728	G.729
Rate (kbit/s)	64	64	6.3	5.3	16	8
Frame size (ms)	1	1	30	30	2.5	10
Frame size (octet)	8	8	24	20	5	10
Default packet size (ms)	20	20	30	30	20	20
Default Frames per packet	20	20	1	1	8	2
Default payload size	160	160	24	20	40	20
Default packet size (cells)	4	4	1	1	1	1
Unused octets	24	24	16	20	0	20
Cell Rate (cell/s)	200	200	33.4	33.4	50	50

Table 7 Cell rates for Compressed RTP over ATM voice transport using H.323 default packetization.

Therefore, if the default packetization intervals are used, RTP-only header compression is only justifiable for G.728 streams. It is however possible to improve the efficiency by choosing packetization intervals that are better suited to ATM AAL5 CPCS-PDU encapsulation:

- G.711 and G.722

  These codecs are sample-based while H.323 specifies that they should be considered as frame-based with frame size of 1ms. This allows for a very thin granularity in selecting the maximum frame sizes. However, care must be taken to avoid too small packet sizes that decrease the protocol efficiency or too large packet sizes that may compromise interoperability. Different packet sizes using a coarser granularity (e.g., 10 ms) may be preferred for some codec implementations.

- G.723.1

  For G.723.1 at 6.3kbit/s the frame size is 24 bytes and any multiple of such frame will require the same number of cells per packet whether or not the RTP header is present. Nevertheless, a larger packet size could improve the bandwidth efficiency of G.723.1 at 5.3kbit/s. However, G.723 frames are 30ms by construction and larger packet sizes may compromise the delay performance for this codec.

- G.728

  Using a CPCS-PDU size of 40 octets matches perfectly the H.323 default packet sizes of 8 frames per packet. A different packetization interval will not improve bandwidth efficiency.

- G.729

  G.729 packetization is much more efficient if 4 frames are included per CPCS-PDU instead of the H.323 default of 2 since it corresponds exactly to one cell.

The above discussion suggests that the H.323-H.323 gateways need to notify the H.323 endpoints of the number of frames to include in an RTP packet in order to improve bandwidth efficiency. This can be done through the H.245 **AudioCapability** that specifies the maximum number of codec frames in a packet for each codec for receiving, transmitting or receiving and transmitting audio media. This may be one of the entries whose value the H.323-H.323 gateway would change upon relaying the capability message from one side of the gateway to the other.

H.323-H.323 gateways can only advertise higher supported numbers of frames per packet. They cannot *force* the endpoints to use these advertised maximum number values. The actual maximum number of frames selected by the endpoint opening forward logical channels is provided in the **OpenLogicalChannel**. The gateway could use this indication along with the above discussion to define whether RTP-only header compression needs to be applied to the

RTP streams requested by the **OpenLogicalChannel**. Namely, if the default maximum number of frames per packet is selected, a gateway implementation may apply RTP-only compression to G.728 streams only.