INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION STANDARDIZATION SECTOR

STUDY PERIOD 1993 - 1996

AVC-1009-E
Sept 1996
Original: English

Question 2 & 3/15 Rapporteur Meeting

STUDY GROUP 15 - CONTRIBUTION
Eibsee, 24-27 September 1996

SOURCE* : Mark Reid, Stephen Botzko

TITLE: Comments on H.323, H.225 and H.245

PURPOSE: Proposal and Discussion

---

* Contact: Mark Reid — Tel.: +1 508 292 5863
Stephen Botzko — Tel: +1 508 282 3739
PictureTel Corporation — Fax: +1 508 292 3309
USA — E-mail: reid@pictel.com

1.0 Introduction

The document contains various proposals and comments on H.323, H.225, and H.245 for the September Eibsee Meeting.

Specifications effected by proposals

Section	Title	H.323	H.225	H.245
2.0	RAS Proposal	X		
3.0	TPKT Proposal		X	
4.0	Coordinating H.323/T.120 Conferences	X		
5.0	CommunicationModeCommand Procedures	X		
6.0	Outbound Gateway calls		X	
7.0	G.723 Packetization		X	X
8.0	E.164 in LCF		X	
9.0	Multiple Media Stream Indications			X
10.0	MCU Hints			X

2.0 H.225 RAS Message Proposals

2.1 Add Timeout and Retry Parameters for RAS Messages

As identified as future work, add as a second paragraph of H.225.0 Section 7.6 RAS Message Common Parts:

"All request messages between endpoints and Gatekeepers should be retried at a minimum interval of once per second and a minimum of five times."

2.2 ARQ Name Resolution

The Gatekeeper may be required to seek outside assistance (e.g. DNS, ULS, etc.) for name resolution when presented with a name (e.g. reid@pictel.com) during the ARQ process. Given the timeout and retry mechanism above, it is desirable for the Gatekeeper to have a message that it can send periodically to an endpoint to indicate that it is working on the ARQ. This message would reset the timeout/retry timer proposed in Section 2.1 so that when external name services are required, the ARQ/ACF process can be extended.

Changes Required:

1. Add AdmissionInProgress to RasMessages

```
RasMessage                          ::= CHOICE
{
    ... data skipped ...
    unknownMessageResponse          UnknownMessageResponse,
```

```
    ...,
    admissioninprogress                AdmissionInProgress -- (AIP)
}
```

2. Add AIP to table. AIP would be optional for endpoint receive and Gatekeeper transmit.

RAS Message	*Endpoint (Tx)*	*Endpoint (Rx)*	*Gatekeeper (Tx)*	*Gatekeeper (Rx)*
AIP		O	O	

3. Define AIP

```
AdmissionInProgress                ::=SEQUENCE --(AIP)
{
    requestSeqNum                  RequestSeqNum,
    nonStandardData                NonStandardParameter OPTIONAL,
    ...
}
```

3.0 TPKT Changes

3.1 TPKT for H.245 Messages

In H.225 section 6.1 we state:

> "When messages are sent on the reliable H.245 control channel, more thanone message may be sent in a single packet as long as whole messages are sent; there shall be no fragmentation of H.245 messagesacross packets."

We should change the words "sent in a single packet" to "sent within a single TPKT header" and "across packets" to "across TPKT header boundaries" as "packet" is ambiguous.

3.2 TPKT for H.225 Messages

We should add the following paragraph to H.225 section 6.1 to mandate that each H.225 message should have its own TPKT header.

> "When messages are sent on the reliable H.225 control channel, only one whole message shall be sent within a single TPKT header and there shall be no fragmentation of H.225 messages across TPKT header boundaries."

4.0 Recommendation for Coordinating H.323/T.120 Conferences

In order to properly coordinate a T.120 conference with the H.323 media and control streams a number of issues arise which aren't clearly addressed in either standard.

In particular, the call setup procedures must ensure that the H.323 and T.120 connections are properly associated, otherwise inappropriate conference admission may result.

In addition, the T.120 top provider and the H.323 MC both have responsibilities for conference control.

Coordinating the T.120 top provider and the H.323 MC to be in the same device, and providing consistent connection setup procedures will product more robust conferences which have more predictable characteristics for users.

This submission includes suggestions which address these issues, and permit seamless conference control and interoperability between T.120/H.323 endpoints. Most of these recommendations apply to devices which support both T.120 and H.323.

These proposals do not address the relationship of cascading MC's which are for further study.

Recommendation 1:

The H.323 control channel should always be established before the T.120 connection is made. The motivation behind this principle is very simple - T.120 is an optional part of H.323. The H.323 capability exchangeneeds to be done so that the two endpoints know that a T.120 conference can be established. For reasons that are discussed later, the H.323 call setup should be used to establish the active MC (if any), before a T.120 connection is made.

Recommendation 2:

For an H.323/T.120 device, admission to an H.323 conference implies admission to the associated T.120 conference, and vice-versa. Admission to H.323 media streams and the T.120 applications should be deferred until both the T.120 control connection and the H.245 connections are established. A normal disconnect from one protocol should be followed by a normal disconnect on the other.

Stated more simply, from the user's perspective, there is only *one* conference, which uses both the H.323 and T.120 protocols. Obviously, if a device only supports H.323 or only supports T.120, then the is only one conference for that end-point by definition.

Recommendation 3

The H.323 callee always supplies a dynamic port to the caller for the opening of the T.120 logical channel. This allows the receiver of the call(s) (the "callee") to be able to associate the T.120 connection with the H.323 connection. This rule is executed by the following H.245 call setup procedure:

Before the T.120 connection is made, the H.323 endpoints open a T.120 logical channel, as specified in the H.323 standard. The caller never sends the T.120 transport address. The callee allocates a dynamic TCP/IP port for the T.120 connection, and sends that port to the caller as part of its T.120 transport address. If a dynamic port is not used, the callee could possibly incorrectly associate the two connections. This would occur if a T.120 only endpoint attempted to connect to the callee while the H.323/T.120 connection was being established.

Recommendation 4:

If T.120 is supported, T.120 should be used for admission to the conference. If the conference is password protected, the password should be queried via T.120 whenever possible.. If an end-point doesn't support T.120, or if the T.120 connect request times out, then the H.245 procedures should be used. Whichever protocol is used to request the password, the password itself and the password validation procedures should be identical.

Recommendation 5:

In a H.323/T.120 conference, the device which is the active MC should also be the T.120 top provider. Disconnecting either the activeMC device or the T.120 top provider has drastic effects on the conference. Keeping both conference control facilities in the same device creates more predictable behavior for the users,and allows the two control facilities to coordinate their activities (without requiring further standardization!). In order to implement this in the specific case of decentralized H.323 conference, the T.120 stack in the MC device should identify itself as a "T.120 MCU".

Recommendation 6:

A device which supports both an H.323 MC and the T.120 protocol must be capable of acting as a T.120 MCU. This follows from Recommendation 5.

Recommendation 7:

The following T.124 actions are recommended for the H.323 *create, join, and invite* requests:

(1) H.323 CREATE. Once the H.323 active MC is determined, the T.120 top provider will also be in the same device (as described above). If there is no MC, the T.120 top provider could be in either device. (In this case, the existing rules used for determining appropriate actions already defined in T.124 should be followed.) The T.120 top provider will create the T.120 conference. Since an H.323 Create action implies an ad-hoc H.323 conference with no password protection, the T.120 top provider should invite the other node. The new "wait for invitation" indication which has been added to the T.120 conference query should be used if the T.120 top provider is the callee.

(2) H.323 JOIN. An H.323 join request will only be made when an endpoint is calling the active MC. In this case the called device is also the T.120 top provider. If the H.323 conference is ad-hoc, or if there is no password protection on the conference, then the top provider should invite the new node to join the conference. As stated above, use of the new "wait for invitation" indication is recommended. If the conference does have password protection, then the new node should not be invited (since a T.120 invite doesn't permit password validation). In this case, the top provider should use the new T.120 default conference flag to indicate to the far end which conference should be joined.

(3) H.323 INVITE. The H.323 invite request can be sent to the active MC, or it can be sent by the active MC. When an active MC receives an H.323 invite request, the H.323 action is essentially the same as a T.120 "dial-through". If the active MC allows the invite, it will call the specified far end, and send the H.323 INVITE request to the new end-point. So if the H.323 invite is *sent to the active MC*, no action should be taken by the T.120 protocol. If the H.323 invite is *sent by the active MC*, the T.120 top provider (in the same device) will invite the far end to the appropriate T.120 conference.

Recommendation 8:

Before the T.120 top provider establishes a connection to a new node, the MC should attempt to establish an H.323 connection to that node. This presumes that the transport used to establish the T.120 connection supports H.323 (e.g. is a IP / IPX network), and the H.323 MC / T.120 top provider have no a-priori information regarding the new node's support of H.323. If the H.323 connection cannot be made, a T.120 connection may be made (adding the new node as a T.120-only device). T.120-only connections can be made from any node in the conference that supports T.120.

That is, there is no requirement for T.120 devices to limit their connections to the H.323 MC/T.120 Top Provider, as there is for H.323/T.120 nodes.

Recommendation 9:

A pre-existing T.120 (only) conference can not add H.323 connections. Unfortunately, this situation requires that the T.120 conference be torn down, and set up again. One exception: If the T.120 top provider also is an H.323 MCU, the T.120 conference can be extended easily to include new H.323/T.120 nodes.

Recommendation 10:

The following recommendations are made regarding H.323 Gateways. An H.323 gateway should determine the type of H.320/H.324 device it is connecting with before it responds to an H.323 setup message. If the H.320/H.324 device is an MCU, then the gateway should indicate that it is an H.323 MCU in its capset. In that situation, the T.120 top provider will either be in the H.320/H.324 MCU, or be in an MCU which is cascaded to it. If the H.320/H.324 device is *not* an MCU, then the gateway should signal that it has no MC. If the H.323 endpoint has an active MC, then the gateway should act as an MCU to its attached H.320/H.324 endpoint. The cascaded situation (where the H.323 endpoint has an active MC and the attached H.320/H.324 device is an MCU) is for further study. It is possible that the H.320/H.324 device is *not* an H.320/H.324 MCU, but *is* a T.120 MCU. This situation is also for further study.

5.0 Communication Mode Command Procedures

There are a number of procedures associated with the CommunicationModeCommand which need to be documented in H.323. This section proposes some of the procedural statements and provides a sample CommunicationModeCommand which should be considered for inclusion in H.323 to promote interoperability.

5.1 Statements

1. The CommunicationModeCommand conveys the transmit mode which an endpoint is to use for a conference. The command does not convey receive mode parameters, as they are specified by OpenLogicalChannel commands from the MC to the endpoint.
2. For media streams such as audio or video, the MC should open a logical channel for transmission to an endpoint after the endpoint sourcing the media has opened the channel to the MC.
3. In the CommunicationModeCommand, the terminalLabel is present in a specific table entry only if the table entry represents a media stream which is specific to a

single endpoint. For example, when audio from all endpoints in a conference is placed on one multicast address (one session), the table entry for the audio mode and address would not contain a terminalLabel. If the table entry is commanding an endpoint to send its video stream to a particular multicast address which is for that node only, the MC would include that endpoint's terminalLabel in the table entry.

4. When the MC commands an endpoint to open an RTCP channel for a video stream from another endpoint, the table entry in the CommunicationModeCommand will look exactly like an entry for video and video control with the exception that the mediaAddress will be omitted. (Note: When we collapsed the mediaAddress and mediaControlAddress into one OpenLogicalChannel/Ack sequence, we removed the VideoControl and AudioControl selections for opening a logical channel. In the OpenLogicalChannel and CommunicationModeCommand case, the open is for RTCP only where, the lack of a mediaAddress signifies a video control channel. The dataType for video control will have to be filled out as a video mode.

5.2 Sample MC to Terminal Communication Mode Command

Example: Endpoints A, B and C are in an audio and video distributed conference using multicast. The MC (which could be anyone of the nodes), has decided to place the media and media control channels on the following multicast addresses:

Stream	Multicast Address
Audio for all endpoints:	MCA1
Audio Control for all endpoints:	MCA2
Video from endpoint A:	MCA3
Video Control data about endpoint A:	MCA4
Video from endpoint B:	MCA5
Video Control data about endpoint B:	MCA6
Video from endpoint C:	MCA7
Video Control data about endpoint C:	MCA8

CommunicationMode Table to Endpoint A

All entries are commands for endpoint A to open a logical channel for transmission. TerminalLabel is only present when the entry is not specific to a single endpoint in the conference. Entry three and four's lack of a mediaChannel indicates that it is RTCP.

ENTRY ONE - AUDIO & AUDIO CONTROL FOR CONFERENCE

sessionID	1
sessionDescription	Audio
dataType	Audio Capability
mediaChannel	MCA1
mediaControlChannel	MCA2

ENTRY TWO - VIDEO FOR NODE A

sessionID	2
associatedSessionID	1
terminalLabel	M/T for A
sessionDescription	Video for Node A
dataType	Video Capability
mediaChannel	MCA3
mediaControlChannel	MCA4

ENTRY THREE - VIDEO CONTROL ABOUT NODE B

sessionID	3
terminalLabel	M/T for B
sessionDescription	Video Control for Node B
dataType	Video Capability
mediaControlChannel	MCA6

ENTRY FOUR - VIDEO CONTROL ABOUT NODE C

sessionID	4
terminalLabel	M/T for C
sessionDescription	Video Control for Node C
dataType	Video Capability
mediaControlChannel	MCA8

CommunicationMode Table to Endpoint B

All entries are commands for endpoint B to open a logical channel for transmission. TerminalLabel is only present when the entry is not specific to a single endpoint in the conference. Entry three and four's lack of a mediaChannel indicates that it is RTCP.

ENTRY ONE - AUDIO & AUDIO CONTROL FOR CONFERENCE

sessionID	1
sessionDescription	Audio
dataType	Audio Capability
mediaChannel	MCA1
mediaControlChannel	MCA2

ENTRY TWO - VIDEO FOR NODE B

sessionID	3
associatedSessionID	1
terminalLabel	M/T for B
sessionDescription	Video for Node B
dataType	Video Capability
mediaChannel	MCA5
mediaControlChannel	MCA6

ENTRY THREE - VIDEO CONTROL ABOUT NODE A

sessionID	2
terminalLabel	M/T for A
sessionDescription	Video Control for Node A
dataType	Video Capability
mediaControlChannel	MCA4

ENTRY FOUR - VIDEO CONTROL ABOUT NODE C

sessionID	4
terminalLabel	M/T for C
sessionDescription	Video Control for Node C
dataType	Video Capability

mediaControlChannel	MCA8

CommunicationMode Table to Endpoint C

All entries are commands for endpoint C to open a logical channel for transmission. TerminalLabel is only present when the entry is not specific to a single endpoint in the conference. Entry three and four's lack of a mediaChannel indicates that it is RTCP.

ENTRY ONE - AUDIO & AUDIO CONTROL FOR CONFERENCE

sessionID	1
sessionDescription	Audio
dataType	Audio Capability
mediaChannel	MCA1
mediaControlChannel	MCA2

ENTRY TWO - VIDEO FOR NODE C

sessionID	4
associatedSessionID	1
terminalLabel	M/T for C
sessionDescription	Video for Node C
dataType	Video Capability
mediaChannel	MCA7
mediaControlChannel	MCA8

ENTRY THREE - VIDEO CONTROL ABOUT NODE A

sessionID	2
terminalLabel	M/T for A
sessionDescription	Video Control for Node A
dataType	Video Capability
mediaControlChannel	MCA4

ENTRY FOUR - VIDEO CONTROL ABOUT NODE B

sessionID	3
terminalLabel	M/T for B
sessionDescription	Video Control for Node B
dataType	Video Capability
mediaControlChannel	MCA6

5.3 CommunicationModeCommand Procedures

After receiving the CommunicationModeCommand, each endpoint will close any previously opened unicast channels which are inconsistent with the MC's command. Each endpoint will then proceed to open logical channels to the MC for media streams it is sourcing and wait for the MC to open the media streams associated with the RTCP channels indicated in the CommunicationModeCommand.

The MC should wait until the source of a media stream has opened the appropriate logical channel before opening the logical channel of the media to the receiving endpoint.

For example: The sequence for video from endpoint A to endpoint B and C is as follows:

1. Endpoint A opens video and video control to the MC, the MC acknowledges. (The addresses passed are redundant with those in the CommunicationModeCommand)
2. The MC opens video and video control to Endpoint B and C, the endpoints acknowledge and open the return video control as specified in the CommunicationModeCommand. (Again, the addresses are redundant)

6.0 Outbound Dialing via Gateway

There are a number of dialing scenarios which we have considered in the H.323 specification:

Callingon the LAN

a) rcid@pictel.com	H.323 Alias
b) 5863	An E.164 extension
c) 140.12.12.12	An IP address

CallingISDN (H.320) via a Gateway

d) 9 7005601112 and 9 7005601113

where "9" is an arbitrary prefix for ISDN in this example

CallingGSTN (H.324) via a Gateway

e) 8 5082923500

where "8" is an arbitrary prefix for GSTN in this example

Callinga LAN endpoint via two Gateways

f) 9 7005601112 and 9 7005601113 and 5863 as the extension on the LAN.

Notes:

1. In case "b)", "c)" and a "1B case d)" call, the Gatekeeper has to have extensive programming to determine if the call is to the WAN via a gateway or to a local node with an extension. With the prefix strategy we have adopted, the first digit of local extensions will have to be restricted to the point where using your phone extension as your endpoint extension maybe precluded.

2. There have been proposals to use a delimiter such as "**" in the dial string to carry the 5863 in case "f)" and other proposals to place the 5863 in the subaddress portion of the SETUP message on the link from the LAN endpoint to the Gateway. Given that the far end gateway in case f) may require a sub-address to be reached itself, we should not adopt this strategy.

Based on Notes 1 and 2 and the fact the destinationAddress already exists in the SETUP message, we should always include the LAN based extension (E.164 field) in the destinationAddress if it is present. We should never place a LAN extension in the true Q.931 called party number fields.

This would make the ARQ and SETUP messages consistent, eliminate ambiguity for call routing in the Gatekeeper, and relieve Gateways of the processing required to parse the dial string looking for an extension. The "**" may conflict with a local dialing convention or be used by some device external to a gateway which requires it in the dial string.

Note: Gateways will pass extensions between Gateways (case f) using the TCS-4 command in H.230.

The following table proposes the location of fields for dialing the various scenarios described above:

Message	Field Name	Local LAN Call with E.164 extension "3230"	Local LAN call with Alias "h323node"	2B Gateway call to H.320 node	2B Gateway to Gateway call to H.323 node with E.164 address "3230"
ARQ	destinationAddress	3230	h323node		3230
	destExtraCallInfo			7005601101 7005601102	7005601101 7005601102
Setup	Called Party Num			7005601101	7005601101
	destinationAddress	3230	h323node		3230
	destExtraCallInfo			7005601102	7005601102

7.0 Comments on G.723.1 Packetization Proposal for H.225

In the "RTP Payload Format for G.723 Audio Streams" proposal of May 17, 1996 contributed to the IETF and ITU by Intel Corporation, three modes of encapsulating G.723 into RTP are proposed. The first, "Sequential G.723 audio data" matches the text of H.225.0 in that the whole number of G.723 frames would be placed into RTP packets. The other two options, "interleaving frames" and "duplicating frames", requires special marker bit processing and a payload header for operation.

Although the optional payload header formats are useful for Internet radio and other playback applications, these options should not be required for a purely conversational endpoint. To allow for this, we need to add a capability to the existing H.245 MediaPacketizationCapability to signal whether or not you support the proposed packetization.

```
MediaPacketizationCapability            ::=SEQUENCE
{
    h261aVideoPacketization             BOOLEAN,
    ...,
    g723aAudioPacketization             BOOLEAN

}
```

g723aAudioPacketization indicates that the endpoint supports both the interleaving and duplicated frame option of the G.723 RTP payload format described in H.225.

In the logical channel open, notification would be given as to the use of the packetization scheme:

```
H2250LogicalChannelParameters           ::=SEQUENCE
{
    ... parameters skipped ....
    dynamicRTPPayloadType               INTEGER(96..127) OPTIONAL,
    mediaPacketization                  CHOICE
    {
        h261aVideoPacketization         NULL,
        ...
        g723aAudioPacketization         NULL,
    } OPTIONAL,
    ...
}
```

8.0 E.164 numbers in LCF

The Gatekeeper should have the capability to return addresses which require a Gateway as the result of an LRQ request. (e.g. LRQ for "help desk" returns 7005601112/1113 as proposed Pete Cordell's July 10th email).

```
LocationConfirm                         ::=SEQUENCE --(LCF)
{
    requestSeqNum                       RequestSeqNum,
    callSignalAddress                   TransportAddress,
    rasAddress                          TransportAddress,
    nonStandardData                     NonStandardParameter OPTIONAL,
```

```
    ...,
    destinationInfo                 SEQUENCE of AliasAddress OPTIONAL,
    destExtraCallInfo               SEQUENCE of AliasAddress OPTIONAL
}
```

9.0 Multiple Audio Streams on one Logical channel (Multicast)

When multiple audio or video streams are multicast on a single session, (e.g. one multicast address) there will be multiple streams on a single logical channel. In these cases, miscellaneous indications such as logicalChannelActive/InActive need to be further identified with a TerminalLabel as LogicalChannelNumberis insufficient. We propose adding an optional TerminalLabel to the MiscellaneousIndication to qualify which terminal is being referenced in the multicast conference.

```
MiscellaneousIndication             ::=SEQUENCE
{
    logicalChannelNumber            LogicalChannelNumber,
    type                            CHOICE
    {
        logicalChannelActive        NULL,
        ... data skipped ...
      }

    },
    ...,
    terminalLabel                   TerminalLabel           OPTIONAL
}
```

10.0 MCU Hint for Endpoints

In cases where an MCU is running a hybrid conference, it is desirable forthe MCU to provide multicast based endpoints information about the MCU's selected speaker. Although endpoints would not be mandated to switch to the speaker, the indication could save valuable endpoint processing time.

```
MiscellaneousIndication             ::=SEQUENCE
{
    logicalChannelNumber            LogicalChannelNumber,
    type                            CHOICE
    {
        logicalChannelActive        NULL,
        ... data skipped ...
        ...,

        videoNotDecodedMBs          SEQUENCE
        {
          firstMB                   INTEGER (1..6336),
          numberOfMBs               INTEGER (1..6336),
          temporalReference         INTEGER (0..255),
            ...
        },
        channelInMCUMix             NULL

    },
    ...,
```

terminalLabel	TerminalLabel	OPTIONAL

}

channelInMCUMix indicates that the logicalchannel is currently being processedby an MCU as an active audio or video channel in a H.323 conference. This indication is intended to provide endpoints in the distributed portion of a hybrid H.323 multipoint call with information about MCU decisions in the centralized portion of the same call. As an example, if an MCU/Gateway is selecting an endpoint as the speaker of a conference in the centralized portion of the call, this indication would tell the multicast participants on the LAN of the MCU's decision.