C.C.I.T.T. WP XV/1
Specialists' Group on Coding for Visual Telephony

Document # 135
June 1986

Question 4/XV

English Version

Source : France

Title :

SOME CONSIDERATIONS ON MULTIPOINT VIDEOCONFERENCING

1. INTRODUCTION

"In a multipoint videoconference, more than two terminals are involved and interconnection of these terminals is performed through one or several Multipoint Control Units (MCUs). An MCU may be considered as a piece of equipment, say a server, located at a node of a digital network. It receives several videoconference channels from network access ports. The purpose of the MCU is to permit the transmission of signals including coded audio and video, and control/indication information among a number of separated terminals". This definition is inspired from [1].

An implementation of this system already exists in Europe [2], [3].

Figure 1 gives the basic reference network configuration for such a system. The multipoint connection may then be considered as n point-to-point connections from the terminals to the MCU, each being established separately.

For 384 kbit/s codecs (2nd generation), the same configuration may be used with the following changes :

- In the primary rate connection from the terminal to the network, only an HO channel is used.

- The n primary rate channels used to connect the MCU to the network may be replaced by only one by multiplexing the HO channels.

This paper considers the influence of the MCU definition and functions on the codec design.

2. MCU FUNCTIONS

Although reference [1] foresees the possibility to simultaneously display all video signals, alternating display seems the most advantageous solution. This implies switching between different video signals, whether it is automatic according to audio level or manual according to conferee/ chairman control. From a service point of view it is necessary that the switching between two video signals would not produce any annoying degradation (false blocks, picture break, etc) for the viewer, on the decoded picture.

Four levels of synchronisation must be considered :

- network clock : the primary rate clook must be the same for all streams to avoid the loss of information and of PCM synchronisation,
- picture continuity : a careless switching could lead to a discontinuity in group-of-block numbering in the GBSC and errors being visible in the picture,
- picture memory updating : interframe and MC coding would produce erroneous blocks after switching,
- buffer synchronism : the encoder and decoder buffers are supposed to work in opposite phase. After a switch this is no more true and could lead to a buffer shift and the corresponding degradation.

Other MCU functions can raise problems : data broadcast through the data time slot, encryption.

3. CLOCK SYNCHRONISM

This is easily obtained if the whole network is synchronous. If this is not the case (eg. one subnetwork is not synchronous), the MCU must act as a synchronisation source and impose the same clock (its own one) to all codecs. If the MCU is connected to a synchronous network, then the MCU can have its own clock or take its clock from any incoming port.

To impose its clock to all codecs, the MCU may use the "Synchronous operation bit" defined in document #77. As said during the last Tokyo meeting, the best place for such a bit is in the Service Channel of the frame structure for the following reasons :

- TSO is not end-to-end transparently transmitted.
- For SGXVIII, ISDN is a synchronous network and there is no need to modify G.704 or I.431.
- Even if the network would be capable of handling this bit in TS 0, it will be difficult and expensive to have this function implemented in all digital exchanges at the same time in all countries.

4. VIDEO SWITCHING

Two solutions are possible. The first one, similar to the one used in the first generation codecs, limits the exchange of information by using one-way only protocols, ie an action is done while a signal describing this action is sent without any command/reponse protocol. The second one, uses a message-type protocol where every action is preceded by a request and an acknowledge, which introduces unacceptable delay in video switching. The first solution is preferred.

The absence of degradation during the switching may be obtained by the following procedure :

1) The MCU when deciding to switch (automatically or on chairman/ conferee request) asks all decoders to freeze their received video ("Freeze Picture Request").

2) It then executes the switching from video signal A to video signal B. Degradations occuring in all decoders (buffer resync, video multiplex resync) are not visible since the picture is frozen. Decoders then regain all synchronisations.

3) It then asks the switched-in encoder A to empty its transmit buffer and encode its next picture in intraframe mode with such coding parameters as to avoid buffer overflow ("Fast Update Request").

4) When receiving a picture attribute indicating intraframe mode, the decoders resume normal video decoding.

Points 2) or 3) may occur simultaneously or be inverted.

It is proposed that those "Freeze Picture Request" (FPR) and "Fast Update Request" (FUR) may be coded with one bit each in the Application Channel of the Audio Time Slot.

5. DATA CONTINUITY

For the moment, two kinds of data transmission are envisaged : one at 64 kbit/s using the data time slot, and the other at 320 or 256 kbit/s replacing the moving video. Figure 2 shows the different possibilities.

A solution which does not involve any message channel or command/response protocol, eg by using BAS monitoring, is not viable since it is delicately implementable and seriously restricts the service. The following procedure is proposed :

i) Codec A sends a request to transmit data to the MCU and clears its data channel.

ii) The MCU asks all codecs to clear their data channel in turn.

iii) All codecs clear their data channel

iv) The MCU, having received all acknowledgements, inserts the A data channel in all outgoing ports (except A) and sends A an authorisation to transmit.

v) A transmits data. When it is over, A sends a request to end data transmission to the MCU.

vi) The MCU withdraws the A data channel from outgoing ports and sends an acknowledge to A. It then sends to all other codecs a request to resume video transmission in the data channel.

Note 1 : This protocol does not impose A to be seen by everybody before the data transmission begins.

Note 2 : This protocol only considers one-way data transmission. For two-way transmission, the protocol must be completed to identify which codec is answering and to insert its data time slot in the stream going to A.

6. ENCRYPTION

This is by far the most complex problem since audio signals cannot anymore be decoded, "(n-1)*summed" and recoded. Continuous presence for audio must then be lost and audio must be switched separately. It does not seem possible to switch audio and video together since audio requires fast switching (some ms) while video switching is inherently slow (some tenths of a second). Also the video switching criterion may be different from the audio one. Anyway some basic assumptions may be made:

- Audio, data and video must be ciphered separately to allow independent switching.
- Unciphered audio power must be available to the MCU.
- Only a memoryless cipherment algorithm can be used to allow switching.
- Those considerations do not prevent the use of regular initialisation vectors.

REFERENCES

[1] Annex 8 to the preliminary reply to Question 4/XV. Framework for multipoint videoconferencing system study. CCITT WP XV/1 - Geneva 24-27 Feb 86. TD 21 (XV/1).

[2] J.P. TEMIME. Influence of network aspects on the COST 211 codec design. Globecom 82. Miami 29 Nov - 2 Dec 1982. Section 4 of the paper.

[3] A. NAGRA, J.P. TEMIME, E. VIALE. Multipoint videoconferencing using conditional replenishment codecs. International Teleconference Symposium. London, 3-5 April 1984.

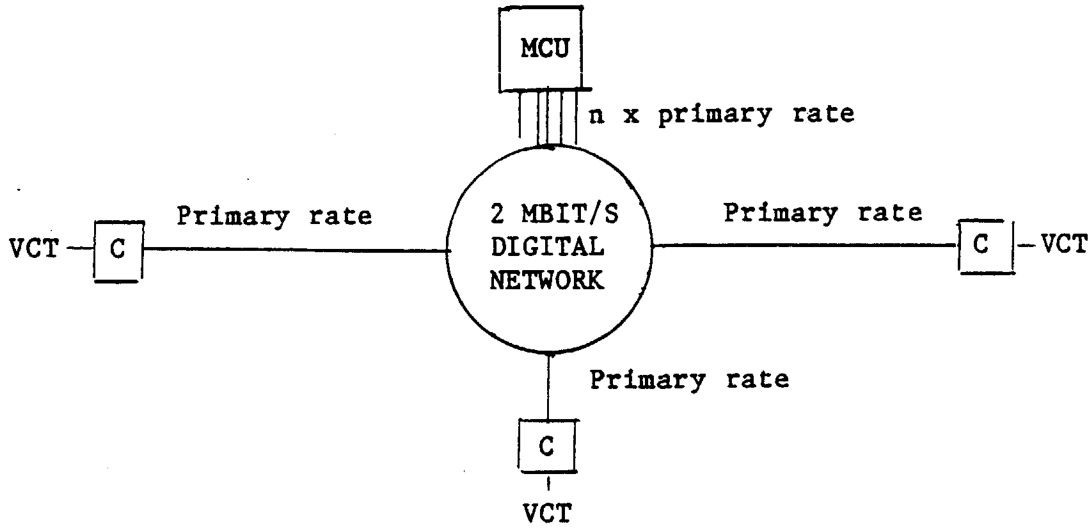

Figure 1. Reference configuration for multipoint videoconferencing

MCU = Multipoint Control Unit
C = Codec
VCT = VideoConference Terminal

Time Slot #	1	2	3	4	5	6
	Audio	Video				
a) Data 1	Audio	Video		Data1	Video	
b) Data 2	Audio	Data2				
c) Data 1 & 2	Audio	Data2		Data1	Data2	

Figure 2. Different kinds of data transmission.