

**ITU-T Telecommunications Standardization Sect
Study Group 15
Experts Group for Video Coding and Systems
in ATM and Other Network Environments**

**AVC 775
15 May 95**

Source: AT&T (Reha Civanlar, Glenn Cash, Barry Haskell)

Title: **Proposals for Error Resilience**

Purpose: Information and Proposal

Pre-transmission of High Priority Information

The impact of cell losses on decoded video quality is most severe if they cause loss of "high priority (HP) information", which, for example, may be defined as the information contained in the HP data partition of MPEG-2 corresponding to a break point value of 1. This information can be protected by transmitting it over a high priority connection with a very low cell loss rate. However, many networks, including existing packet networks, some ATM networks, and all mixes of the two, do not facilitate transmission with different priorities.

We suggest that on such networks, particularly when previously encoded video is being transmitted, the HP information can be pre-transmitted using a reliable transport protocol, eg, TCP/IP, ARQ or any other ACK/NACK, and stored at the decoder to be used during the real-time decoding when cell losses occur. In most cases the pre-transmission can be accomplished without extremely long delays since the HP information defined above constitutes a very small percentage of the entire video data. Also for the same reason, the decoder buffer required to store the HP information is not very large.

Additionally, the following functionalities may be needed:

1. For very long sequences for which the HP information is too large to be pre-transmitted in its entirety, it may be transmitted in parts during the play-back.
2. In cases where the same material needs to be transmitted several times, after the first transmission, the pre-transmission step may be skipped.
3. In addition to the HP information as defined above, some other information such as certain key I frames may be included in the pre-transmitted information.
4. The method, can be applied to an interactive real-time system if the encoder knows beforehand what the header information will be for future frames. If delay is not important, the scheme in item 1 can be used with real-time encoders buffering a portion of the encoded data.

There are a number of syntax choices that could implement these features.:

1. The video information could be encoded according to the data partitioning (DP) specification of MPEG-2 with breakpoint = 1. However, there is no profile currently defined for DP
2. The video information could be encoded according to the data partitioning (DP) specification of MPEG-2 with breakpoint = 127 and all EOB's in the base layer. Then the enhancement layer contains only Sequence, GOP, Picture and Slice headers, which is exactly the HP information described above that would be transmitted on the low loss channel. The

base layer could be decoded, in principle, by non DP decoders. However, the presence of the `sequence_scalable_extension()` and priority break points may break some standard profile decoders.

3. Same as above, except the base layer would not contain the priority break points or `sequence_scalable_extension()`. The Hierarchy descriptor (13818-1, Sec. 2.6.6) would define the DP enhancement layer, which would only contain Seq, GOP, Pix and Slice headers. Normal decoders should ignore the Hierarchy descriptor.

4. Same as above, except the enhancement layer contains some additional redundantly copied video data such as I-frames. Decoders must be smart enough to skip over unneeded redundant data. Error recovery is also enhanced if the slice headers have some variability from frame to frame. (However, see additional proposal below).

Need for including slice_picture_id in slice header

Prior to the Lausanne MPEG meeting, slice headers did not carry information about the frame or field number. Under moderately large packet losses, which may occur on conventional non-guaranteed bandwidth networks, this may cause a decoder to decode data for one picture using the decoding syntax and picture level data for another picture. This causes errors even if the decoder synchronizes using the slice start codes. Additionally, not knowing the frame number may cause errors in the display timing.

The Lausanne MPEG meeting provided a method of including frame or field number information in the slice header using the parameter `slice_picture_id`. This feature combined with reduced error rate transmission of the "high priority data" (see above) makes it much easier for decoders to recover after such packet losses.

Video tapes of VBR MPEG2 under 1e-2 packet loss rate will be shown.

Need for H.222.1 wording on Data Partitioning

We propose the addition of wording in H.222.1 to enable the above error resilience proposals. Specifically, in Section 12 add the following text:

In order to improve error resilience, especially during severe error bursts, the video information may be encoded according to the data partitioning (DP) specification of H.262 with breakpoint = 127 and all End-of-Block (EOB's) in the base layer. Then the enhancement layer contains only Sequence, GOP, Picture and Slice headers, and may be transmitted on a low-loss virtual channel. The base layer could be decoded, in principle, by both DP as well as non DP decoders. However, the presence of the `sequence_scalable_extension()` and priority break points may disrupt some standard-profile non DP decoders.

In order to improve interoperability, the above encoding may be implemented with a base layer that does not contain priority break points or `sequence_scalable_extension()`, in which case a **default_break_point** value of 127 is used. The Hierarchy descriptor (H.222.0, Sec. 2.6.6) defines the DP enhancement layer, which in this case contains only Sequence, GOP, Picture and Slice headers.

In order to further improve error resilience, the enhancement layer may also contain some additional redundantly copied video data such as I-frames or I-slices. Non-error-resilient decoders will ignore the Hierarchy Descriptor and the enhancement layer.

not
include^t