

CHAPTER 3

Integration of Packet Video Into a Network Architecture Model

3.1 The application layer

3.2 The presentation layer

3.3 The session layer

3.3.1 Session types

3.3.2 Session quality

3.4 The transport layer

3.5 The network layer

3.6 The physical link and data link control layers

3.7 Congestion control

3.8 Summary

In the previous chapter, we saw that in the case of variable-rate transmission of real-time information, the separation of source, channel and receiver can no longer be assumed as a valid paradigm. The strong dependencies between the entities involved therefore require consideration of the global packet video system. A structured way of considering the entire system of signal processing and transmission is to place the required functions in their proper context, a network architecture model. Thereby the interactions between functions can be determined [CHI84a, LAZ87, KAR89a], and the model may help to design a system which minimizes a function's dependency on the format of the video signal. The general model is illustrated in Fig. 3.1. By using the Open Systems Interconnection (OSI) model [ZIM84, BER87, SCH87] of the International Standards Organization as the starting point for discussion, we are provided with an adequate terminology that is not contingent upon any specific

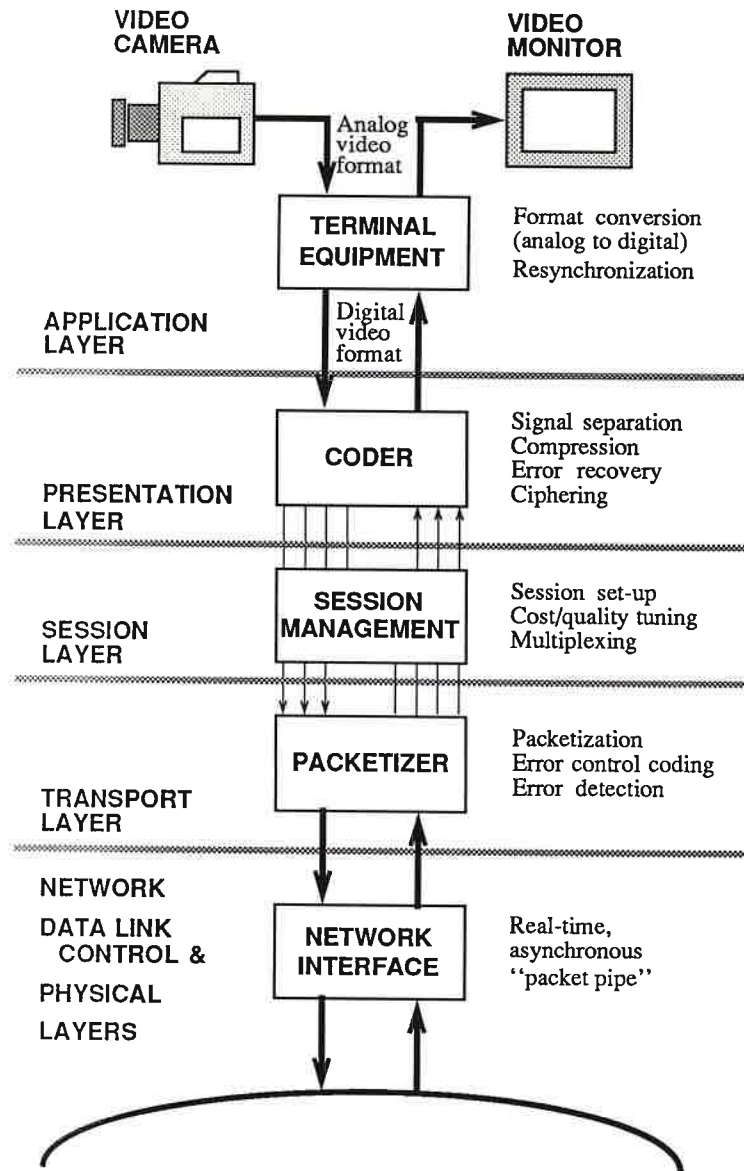


Figure 3.1 The major functions of the layers of the packet video network architecture model. The arrows indicate signal flows; control flows are not considered.

network implementation. Note that the OSI model was not designed to support real-time transmission and the discussion in this chapter is not intended as an extension of the OSI concept. The network architecture model will be considered mainly from a video processing point of view. The control flows will not be covered.

The three lowest layers of the model are the physical link layer, the data link control

layer, and the network layer (see Fig. 3.1). We shall refer to them collectively as the network layers. If the division of functionalities between the layers is kept as suggested in this chapter, then all of the network layers may be designed without consideration to the signal type. The only requirement is that the layers have to support real-time services; it is irrelevant whether it is video or voice (except in need for capacity). Generally, this requires the network layers to be as simple as possible without any processing of the information; sophisticated signal handling is better performed at the upper layers where the signal's format and importance is known. Thus, the network may be described as a real-time, asynchronous "packet pipe," where packets inserted at one end come out orderly at the other end, although some will contain errors and others might have been lost.

While the network layers often constitute a network node, the upper four are at the customer's premises. In that sense, a packet video coder is seen as including the set of functions associated with the upper four layers, and not only the source coding. Each of the network layers shields the layer above from some aspects of the network implementation. Similarly, we would like each of the video-specific, upper layers to shield its service provider from some of the peculiarities of the format of the video signal. Thus, if the network nodes provide real-time transmission, without regard to the information type, then the user's choice of video format, compression method, and encryption can be made independently of the network. In other words, the network does not restrict the introduction of new signal formats or more advanced compression methods. (Still remaining, however, is the dependency caused by the variable rate of the source and the varying capacity of the channel as discussed in Section 2.1.2. But, both the cause, variable rate coding and resource allocation, and the counter measures, resynchronization and error recovery, are functions in the upper four layers.)

The functions of packet video, as described in the previous chapter, are part of the four upper layers of the network architecture model (see Fig. 3.1). The model is presented top-down to illustrate how the dependence on the video signal's format is incrementally reduced. At the end we discuss the issue of congestion control. It is done separately since it pertains to more than one of the layers in the model.

3.1 The application layer

The application layer forms the boundary between the user and the network. For the signal, it provides analog inputs and outputs which adhere to the standard of the users choice and the analog-to-digital and digital-to-analog conversions. This layer is thus dependent on both the analog and digital video formats. To possibly obtain compatibility between different analog video formats, and to limit the variety of digital formats (see Fig. 3.2 (a)), a highly reduced set of digital video formats could be used, as shown in Fig. 3.2 (b). The allowed set of formats have to be such that any possible analog video format can be obtained from its members. Sensible cross conversion between digital formats should be allowed, like a downward compatibility from HDTV to regular television format. The idea originates from the format independent image-communication protocol of Bellcore [FIV88].

Video resynchronization due to unequal clock frequencies can be done by frame rate alteration (see Section 2.4.2). The resynchronization is placed at this layer since it is done according to an external clock (the receiving party's) which is available only at this layer. Note that resynchronization based on an estimation of the sender's clock is performed at the input buffer at the transport layer (see Section 3.4).

The application layer will also have to handle synchronization of video with related voice (sound) sessions. If the end-to-end delay is kept within the prescribed limit

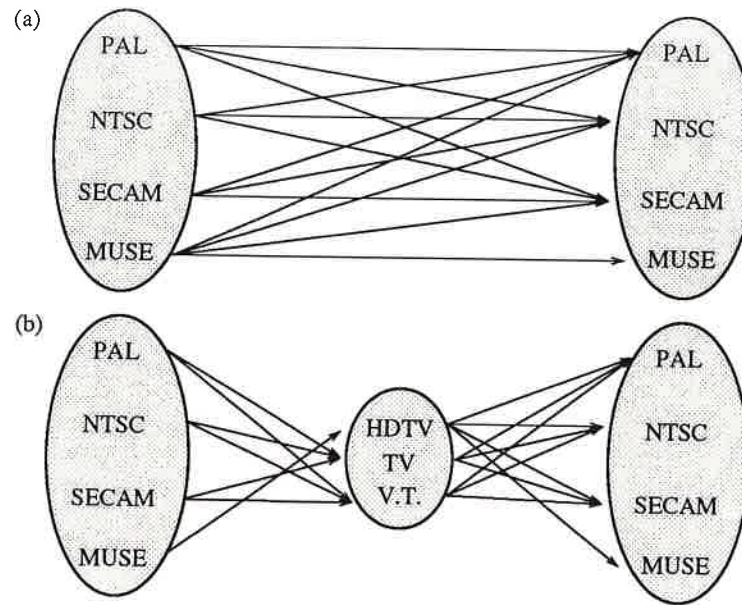


Figure 3.2 Cross-compatibility between some common video formats: (a) direct format to format conversion, and (b) the use of a limited set of intermediate formats for HDTV, regular television, and video telephony (V.T.). (See for example [NET88] for details on formats.)

(see Section 2.4.1) then no additional synchronization procedure is necessary; the possible lag between voice and video is less than what can be perceived. However, if longer end-to-end delays are accepted for simplex sessions, a voice-to-video synchronization is necessary since the lag could otherwise be noticeable. Any multimedia session, integrating data, voice and video, should be administrated by the application layer as well. An example of such a session is transmission of a film with a high-quality sound track and optional subtitles.

3.2 The presentation layer

The presentation layer holds functions which perform some type of signal conversion. Owing to the reduced number of digital video formats stemming from the application layer, presentation layer implementations can be restricted to one class (or standard) for each format. Both the signal separation and the compression of hierarchical source coding would be performed within the context of the presentation

layer. The other functions include ciphering, and error concealment through the use of visual redundancy. Ciphering should be performed after compression since it lowers the signal's redundancy. Error concealment has to be performed after decoding but before the signal's recombination.

Error concealment performed by utilizing visual redundancy, as explained in Section 2.3.2, relies entirely on the assumption that error propagation is limited and that the locations of lost data are known. Because of the variable bit rate, one cannot expect lost data to be aligned to particular points in a video-frame, such as the beginning of a scan-line; nor can the number of erased values be deduced. The information about the ideal splitting-points of the bit-stream is local to the presentation layer where the video format is known, while the packet length is known only at the transport layer. Consequently, limited error propagation has to be achieved locally at the presentation layer by adding control information to the bit-stream.

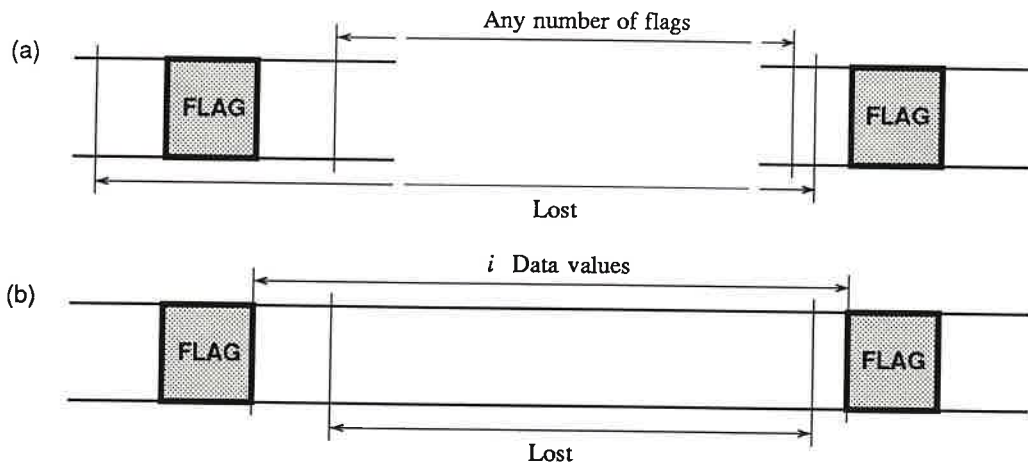


Figure 3.3 Two cases of lost data: (a) one or more flags are missing, and (b) no flags are missing but intermediate data values have nevertheless been lost.

A feasible way of doing this is to insert synchronization flags after every i th sample in a sub-signal [AAR88]. The distance between flags should be chosen so that the

values may be concealed reasonably well if lost, while adding little overhead. The flags have to consist of a unique identifier followed by a sequence number to indicate location and, in case of error, the number of erased segments. The flag's uniqueness has to be guaranteed by bit-stuffing or reserved codewords. The latter appears advantageous if variable length codewords are being used: the flags are inserted before the variable length coding and they are transformed, as any other value, into their reserved codeword. One lost flag would imply that two data segments have been erased at least partly, as in Fig. 3.3 (a), while the case shown in Fig. 3.3 (b) would be detected since the segment would not yield i samples. All correctly received segments of a video-frame would be decoded before the concealment is made, which in turn is done before the signal recombination.

If information about network load is made available to the presentation layer, congestion control may be performed in the most imperceptible manner. When the network load raises, the output rates of the sub-signals can be decreased until eventually entire sub-signals are denied transmission. The sub-signals would be controlled in regard to their standing in the hierarchy. Compare this to congestion control in the network where a priority class is either granted or denied transmission, not permitting any intermediate levels of output control. Thus, by utilizing knowledge about the sub-signals the most graceful degradation is possible in order to meet a temporarily reduced channel capacity. This could also be utilized for enforcement of capacity allocations (see Section 3.7). If a source exceeds its allocated quota the enforcement function would cut its flow. Note that this is analogous to the various compression modes of a coder for circuit-switched transmission. In that case the modes are used to eliminate overflow of the network's access buffer, while in the packet-switched case the mechanism reduces overflow of shared network buffers. Any quality degradation of the video is therefore correlated to the network

load, not the information contents of the video signal. By proper network control (resource allocation and congestion control) these degradations can be rare and of short duration. Of course, the mechanism could be used for fixed-rate transmission if requested.

3.3 The session layer

The session layer is mainly responsible for session set-up and tear-down. Additionally, in our model, the session layer implements the functions which would be invoked only a limited number of times over an entire session. These functions are in contrast to continuously invoked functions, such as the compression. The session layer should provide not only different types of sessions, but also flexibility in the quality of the sessions. The functions of the session layer are completely freed from the format of the video signal. The layer receives a set of sub-signals belonging to a single session and, owing to the format independence, some of these may carry sound information and, possibly, even data. Hence, it is here, at the session layer, that a complete session of integrated real-time transmission is created.

When a video session is initiated there are several issues which need to be decided. These include session type, coding quality and output rate, and transmission quality. Also, some identifier of the transmitted signal format must be passed on to the receiver so that a session cannot be initiated between incompatible coding systems, or so that proper format conversion is invoked. The session layer should possibly provide for renegotiation of quality for an ongoing session. However, it may be problematic to fix a point in a received bit stream where the new parameters will take effect.

3.3.1 Session types

One can foresee several types of video sessions, which can be end-to-end negotiated or locally negotiated, as shown in Fig. 3.4. The former can be the common point-to-point session, but also multi-cast and multi-drop sessions. By multi-cast we mean a one-to-many communication where transmission to each receiver has been negotiated, and all destinations are explicitly addressed during the data transfer phase. Multi-drop is analogous, except that the communication is from many to one, and the receiver chooses, at any point in time, which circuits to tap by switching between all the incoming ones. A conference session would typically consist of multi-cast outgoing transmission and multi-drop return for each user. Both these sessions could be set-up and torn-down together, and all session management could apply to them equally.

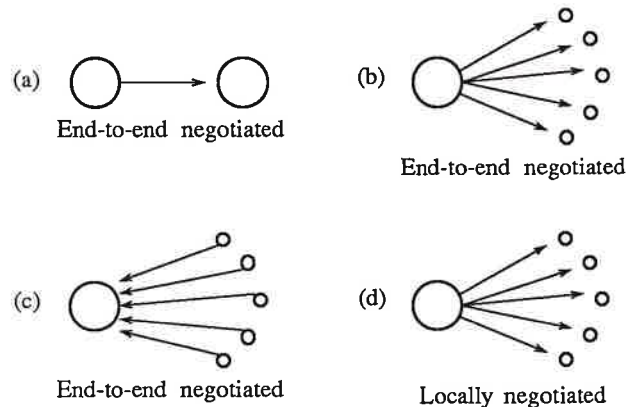


Figure 3.4 The four session types: (a) point-to-point, (b) multi-cast (one to many), (c) multi-drop (many to one), and (d) broadcast.

Only one type of locally negotiated session is conceived, namely broadcast. It is a service going from one source to so large a number of recipients that end-to-end negotiated set-up is rendered infeasible. Also, a new set-up would be required for each new channel selection which would add to its infeasibility. The session set-up should thus be negotiated as locally as possible, meaning at the local network

node. Since there is no set-up from the broadcast source the transmission could be based on permanent virtual-circuits. To receive a broadcast-channel, the associated virtual-circuit is tapped at the node (see Fig. 3.5), *i.e.*, the packets are duplicated at the network layer and the copy is passed to the recipient's transport layer. This procedure can be likened to present broadcast of television where the information is transmitted regardless of viewers, and the viewers choose program locally by tuning in to the desired frequency channel. Since the broadcast-circuits are permanent, and end-to-end delays of simplex sessions are not of importance (see Section 2.4.1), the routing can be optimized once and for all to permeate the network while maximizing channel utilization and offer a balanced network load. Anticipated video services like *video-on-demand* and *democratic television* (see Section 1.1.1) would be naturally obtained through end-to-end negotiated sessions (multi-cast that is).

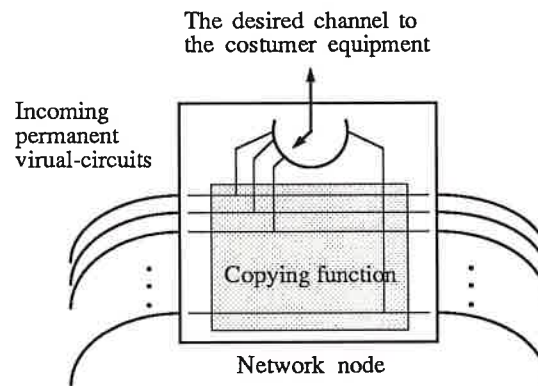


Figure 3.5 Broadcast through permanent virtual-circuits. Packets of the desired channel are duplicated and the copies are passed on to the customer equipment.

3.3.2 Session quality

The session quality depends on two parts: source coding and transmission. In conjunction with hierarchical coding, a desirable quality and output bit rate may be achieved through transmission of an appropriate set of sub-signals. The greater

the number of sub-signals, the higher the quality and the output rate. If the coding method can provide more than one compression mode, a greater flexibility in the cost/quality trade-off may be obtained by changing compression of the sub-signals. For transmission, at issue is the degree to which the session may be affected by network congestion. This is contingent upon the resource sharing policy used by the network management. A prodigal resource allocation may be likened to a circuit-switched system while a parsimonious one will yield a higher capacity utilization, but also more delay and packet loss. For a given resource allocation policy a more consistent quality is obtained if a higher number of the transmitted sub-signals are given high transmission priority, rather than only the most important one. The transmission quality may be raised further by requesting the transport layer to perform error control encoding on a desired set of sub-signals. So, the lossless session quality is set through the number of transmitted sub-signals, and the allowed deterioration under network congestion is controlled through the use of the transmission priorities, or the amount of guaranteed capacity, and error control coding.

Packet video, compressed with hierarchical coding, would be advantageously transmitted with a hybrid of fixed and statistical capacity-sharing. The hierarchy's most important sub-signal would be transmitted with a guaranteed share of network resources, while the less significant sub-signals would have to vie for their share of capacity. This hybrid of allocation policies reduces the stress on the error recovery mechanisms by reducing congestion-related loss of important information, as will be discussed in Section 3.7. Still, it yields higher channel utilization than does fixed allocation (*i.e.*, capacity reservations). The feasibility of such a scheme is discussed in conjunction with the simulation results given in Chapter 6.

3.4 The transport layer

Functions associated with the transport layer include the segmentation of a data stream into packets, the reassembly of the stream at the receiver, error control coding, delay jitter removal and video resynchronization. Since we assume that the network service is restricted to virtual circuits, all packets are guaranteed an orderly delivery and reordering is therefore not of concern. The transport layer serves all sub-signals emanating from the hierarchical coding at the presentation layer and other associated signals, such as sound, which have been added to the session at the session layer. Each such signal would be independently segmented, but the packetized signals would be multiplexed onto the same route at the network layer. Sound and video information will thereby follow a single path so that the delay difference between the two is minimized. Note that the segmentation process does not have information about the video format (such as beginning of a frame or scan-line). The bit-stream may therefore be cut at any point.

For error recovery relying on error correcting codes, the amount of data received has to equal the amount transmitted, or else it would place the error correcting encoder and decoder out of phase (see Fig. 3.6 (a)). Consequently, the transport layer at the destination has to detect lost packets and replace them with dummy packets so that synchronization can be maintained, as shown in Fig. 3.6 (b). Since orderly delivery is guaranteed, end-to-end sequence numbers may suffice for this purpose: a detected gap in the sequence indicates the loss of one or more packets. The necessary range of the sequence numbers has to be determined in relation to the channel-code used, so that all gap-lengths can be detected which can be corrected by the code. The sequence numbers should obviously not be included in the channel encoding. Note that packet intrusion would cause a similar problem by increasing

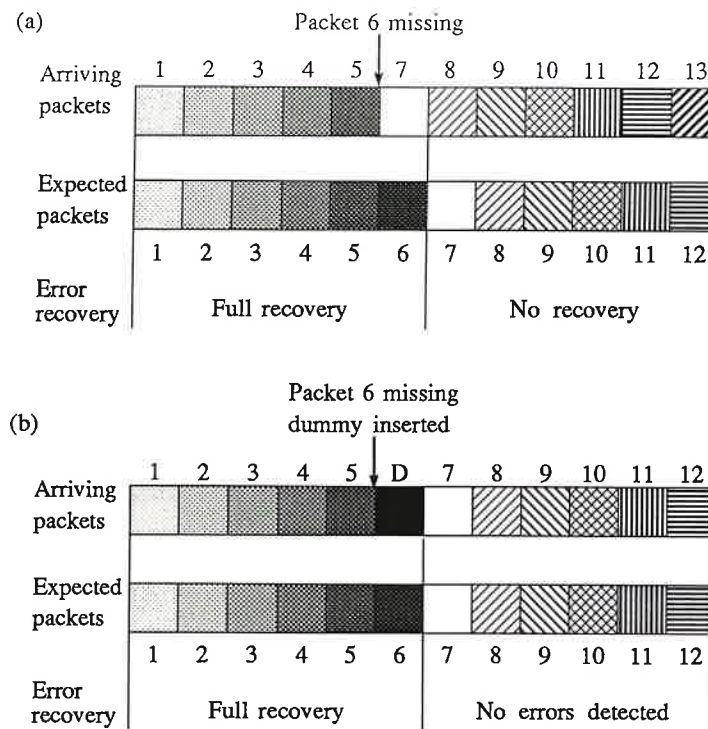


Figure 3.6 (a) The error correcting decoder loses synchronization with the encoder after a lost packet and recovery is no longer possible. In (b), the lost packet is detected and replaced by a dummy packet so that the subsequent decoding will work properly.

the number of packets received. However, we require the network layer to protect the address field of the packet so that intrusion cannot occur (see Section 3.5).

Delays introduced by the channel-coding depend on the code-length, and the packet size: long packets and codewords give longer delays but less overhead. However, this trade-off can be resolved without affecting the functions in the layers above, which are independent of packet format. Consequently, the choice between the use of variable or fixed length packets can be made locally as well. While fixed-length packets simplify segmentation and packet handling (*e.g.*, routing) along the transmission path, variable lengths of the packets could be used to keep the packetization delay constant. With fixed length packets, there is still

a debate between those who favor short packets, mainly to limit packet voice delays (*e.g.*, [COU88]), and those who argue for very long packets, to increase switching speed and reduce overhead (*e.g.*, [TUR88]). Variable length packets bring many complications such as the need for unique data-link frame delimiters, limited flexibility for the resource allocation, and more difficult error control coding. For the latter, recall that error control encoding is applied perpendicular to a block of packets to achieve bit-interleaving. When the packets are of different lengths the shorter packets must be extended. So, the code is not going to be well utilized for these extended parts.

Note that end-to-end retransmission is not a possible error recovery method. Firstly, multi-cast and broadcast sessions would not be feasible if a retransmission scheme was in effect. Consider that different packets may be lost on the various paths of transmission which can result in unreasonable requests for retransmission, proportional to the number of recipients. Secondly, there is a risk of positive feedback: assume a congested network where all video providers experience packet loss. The strict delay constraints bar random waiting times before retransmission. Therefore, if all providers rely on retransmission for error recovery, the congestion could be aggravated and might lead to more severe performance degradation [KAR87, KAR88b, KAR89a].

The input buffer for a receiver is part of the transport layer. The buffer size has to be determined in relation to the magnitude of the delay-jitter caused by the network (see Section 2.4.1). In connection with the removal of the jitter through the buffering, one may extract the clock frequency of the transmitter (see Section 2.4.2). This could be done by monitoring the buffer occupancy or from time-stamps included in the packets [COC85, VER88b].

3.5 The network layer

The purpose of the data link control layer is to ensure communication across a physical link. By relying on this, the network layer can provide end-to-end communication and shields the transport layer above from any physical aspects of the network [BER87, SCH87]. A network connection consists of an establishment phase, a data transfer phase, and a release phase. The functions associated with the network layer during a data transfer phase include: routing (switching), congestion control, and packet duplication for broad- and multi-cast sessions.

The necessity to keep end-to-end delays and packet delay jitter under control requires video transmission to be conducted over a fixed route, a virtual circuit, which also guarantees that packets are delivered in order. The logical channel number should be protected in order to avoid packet loss, or worse, intrusion (delivery of unwanted packets) due to bit error in the address field. (Note that this does not require encoding if the bit-codes of the channel numbers are chosen with appropriate Hamming distance from one another.)

The routing algorithm used for set-up of a virtual-circuit has to aim at minimizing the transmission delay. If a session uses more than one virtual-circuit, provision have to be made so that all of those circuits follow the same route. Note that sub-signals, stemming from the hierarchical source coding, which require order among themselves should be multiplexed onto a single circuit. However, there is in general no such requirement when the sub-signals are coded independently of one another (see Section 2.2). The network layer should provide for temporary as well as permanent virtual-circuits. A permanent circuit is intended as a broadcast television channel as discussed in connection with the session layer in Section 3.3.1. For multi-party sessions the addressing has to allow multiple destinations and the

node has to provide a packet duplication function.

3.6 The physical link and data link control layers

The requirements for the physical link are adequate capacity and low bit-error rate. It is however difficult to quantify these parameters—in general, they are determined by the physical limits of the state-of-the-art technology. Table 1.1 gives a sense of the needs involved in a general network which provides transmission for broadcast television and video telephony services. For example, in order to replace the current cable television system, about 30 channels should be carried. With broadcast video quality at 45 Mbps it would require links with capacity in excess of 1.35 Gbps. Note that the local subscriber loop does not need this capacity. The subscriber needs only a link with enough capacity for the number of channels that should be received simultaneously. Packet loss due to bit-error should be less than loss due to congestion (assuming discard of packets with bit-error). Although the level of tolerable packet loss depends on service type and error recovery procedures, let us assume that it should be kept below 10^{-9} (see Table 1.1). For a packet size of 1000 bits, the bit-error rate must then be far below 10^{-12} , say, by one order of magnitude (assuming that no error control coding is used).

Medium access forms the interface between the physical link and the data link control protocol. For a good overview of the rich field of media access control, see [SAC88]. Slotted media seem to be best suited for video transmission. The first reason is that capacity guarantees can be implemented (*i.e.*, movable boundary schemes), which will be discussed in Section 3.7. Secondly, packet loss due to access contention is avoided.

The data link control layer plays a very important role in data transmission by

converting the physical link into a error free bit-pipe to serve the layer above [BER87, SCH87]. However, most of the tasks associated with it work contrary to the goal of achieving real-time transmission. In order to maximize throughput, the data link control has to be kept to a minimum.

For data transmission the general goal of data-link control is to provide orderly and correct delivery of packets [BER87, SCH87]. In doing so the layer generally relies on automatic repeat request. Automatic repeat request (ARQ) is ruled out as an error handling method for real-time transmission. Retransmission could work but it would make the solution to the problem of packet loss and bit-error dependent on the transmission distance. That is to say, retransmission might be possible in a local area environment where the propagation delays are negligible but not over longer distances. Even though the propagation and processing delays may be short, retransmission will introduce delay variations which are to be avoided. In addition, the processing requirements are substantial for acknowledgments, retransmission requests and possible reordering (for selective reject). Also, in a protocol such as HDLC, a data-link frame (*i.e.*, a packet with added control information) contains a check sum for detection of bit-errors [BER87, SCH87]. When a frame is received in error it is simply discarded (and later retransmitted). However, discard of frames with bit-error only in the video information would be too strong a response, since it inhibits the use of more sophisticated error control at the upper layers. Therefore the error check sum embraces only the control fields of the frame and frames received in error will be discarded as before.

3.7 Congestion control

Since the delivery of video packets cannot be warranted by retransmission, the networks should, through the use of congestion control, maximize the probability

of successful and timely delivery. Given the high transmission rates of video and the small network buffers for short delays, the network can easily be clogged up. By excessive loss of packets the quality of an ongoing session can deteriorate to the point of termination. Congestion control may be performed by several methods for a single network, where each method operates on a different time-scale [TUR86]. For instance, reduction of buffer contents handles instantaneous control, while call blocking may help to reduce the event of congestion on a long term basis.

In order to perform such control in a sensible manner, the layer should provide transmission priorities [TUR86]. When needed, packets are discarded according to their respective priority, with lowest priority packets lost first. This can be performed by separate queues for each priority class or one queue for packets of all priorities. In the first case the queues are served in order of decreasing priority and each queue is served until it is empty. Thus, the system does not maintain order between packets of different priorities. In the second case, packets are discarded by increasing priority-order upon entering the queue when its length exceeds some threshold. Since the queue is FIFO, the packets that are retained maintain their relative order. Both of these congestion control techniques have been studied for integrated voice / data networks by Yin [YIN88].

When time-stamps are used, the packet that will be the oldest when served will be thrown away first. The time-stamp and the position in the queue can determine this if the queue's service rate is deterministic. When combined with priorities, a time-based discard is carried out over the priority classes in order of increased priority. However this requires a choice between age and priority: it is better to discard old packets from all priority classes before discarding newly arrived low-priority packets.

It is advantageous to exert congestion control as input-buffer limiting so that ideally

no packets are transmitted which would be discarded in the network. This requires an end-to-end feedback for monitoring the cut-off levels of all queues at the network nodes traversed by the circuit. An excellent investigation of this issue pertaining to packet voice can be found in [BAI80]. The two methods considered by Bailly *et alii* are aimed at finding the cut-off priority level of the bottleneck-node in the network and no priority class is included in the transmission which will not pass through the network. These methods, although attractive by not loading the network unnecessarily, may not work throughout a wide area network. The time it takes to communicate the state of each buffer in the network back to the sender may be too long to guarantee that the buffer states have not changed entirely. Nevertheless it could be applied to new traffic entering a node to assure uninterrupted service of the transit traffic. This case applies when congestion control is performed at the access points of the network but not within the network nodes (see for example the Prelude project [GON86b, DEV88]).

Priorities may not be necessary if capacity can be guaranteed for a user. That is, the signal has a preemptive right to the guaranteed amount of capacity, but other users can take advantage of unused portions of that capacity. For a slotted network with a frame-structure, or traffic cycles, the capacity guarantees are implemented as a movable boundary scheme [SCH87]. Lowered transmission quality due to congestion can be avoided by allocating generous amounts of capacity to important signals. This approach is suggested by [LAZ85, CID88, TUR88, VER88c, LAZ88, VER88d] among others. If dynamic guarantees cannot be provided, the capacity has to be booked at the set-up of the call. The user has to provide the network with parameters which describe the behavior of the signal that will be transmitted [TUR86, TUR88, VER88c, VER88d]. Candidate parameters include the source's mean rate, its variance, and the maximum rate. The call is blocked if there is no

path available to which the call can be added without degrading the performance of the existing calls. Once the call is accepted, it is monitored to assure that it stays within its allocated quota [TUR88, VER88c, VER88d]. Verbiest *et alii* refer to this function as the police function and places it at the network interfaces (*i.e.*, at the session or transport layers) [VER88c, VER88d]. They do not, however, discuss its realization. Such a function is described by Cidron and Gopal [CID88] and they refer to it as the input “throttle”. (Turner outlines a similar method, called a “leaky bucket” [TUR86].) It works as follows: a token is deposited into a pool at a constant rate of α tokens per second until the deposit has reached its maximum number of \mathcal{N} tokens. Each packet requires a token to leave the queue and to enter the network; no packets can enter when the deposit has been emptied. An upper limit on the departure rate can be enforced by placing a second throttle in series with the first, which works at the rate of β packets per second ($\beta > \alpha$) and which has a maximum deposit of a single token. The throttle scheme is fair but conservative; no user can take advantage of unused capacity. This can be relaxed by adapting the average rate α to the network load.

Lazar *et alii* [LAZ88] define three traffic classes, out of which one is not affected by congestion (*i.e.*, 0% packet loss due to contention) and it provides limited delay for real-time service. The next class may suffer packet loss but it still supports real-time service. The third class is for any information for which the delivery is not time-constrained. In the supporting architecture, each class has its virtual FIFO-queue which all are served by a single server according to a cyclic movable boundary scheme. This appears to be a system well-suited for packet video since the channel utilization can remain high while the information may still get a guaranteed quality of transmission in terms of loss and delay. Its disadvantage is that order is not maintained between traffic classes.

Finally, note that rate-control, *i.e.*, the process of controlling the transmission rate in order not to flood the receiver, cannot be an issue for packet video since the receiver is expecting the signal at the rate with which it is transmitted.

3.8 Summary

The upper layers have associated with them the functions specific to packet video transmission. Most of the traditional video processing functions, such as compression, would fall within the scope of the presentation layer. The exceptions are the format conversion and the resynchronization at the application layer, and the error control coding at the transport layer. Additionally, at the presentation layer, control information has to be inserted into the compressed bit-streams in order to limit the error propagation and enable concealment of lost data. The session layer should provide for various session types, and negotiable quality and cost for each session. At the transport layer lost packets must be detected to ensure that the error control encoder and decoder remain in phase. Also, fixed packet length appears to be favorable since it simplifies packetization, error control coding and routing.

The network layers should act as a real-time “packet pipe” where order is maintained. However, some of the delivered packets may contain bit-errors while others may be lost. The data-link control should be kept to a minimum, barring retransmission and allowing only limited error control. The network layer has to provide virtual-circuits and, for multi-party sessions, the addressing must allow multiple destinations. The network nodes must also be able to provide packet duplication.

If congestion control will be exercised in the network, the network ought to provide

priority classes. This will minimize the loss of perceptually important data and thereby simplify error recovery through the use of visual concealment. Priority-based congestion control may not be necessary when capacity can be guaranteed for the important information. This information will not be affected by congestion as will the less important information, which has to vie for its share of the transmission resources.

The main benefit of using the layered model is to incrementally limit the dependence on the video-format down through the layers. With the format elimination we have obtained, the session and transport layers may therefore be shared between sessions at the same site.