Meeting QoS Challenges for Scalable Video Flows in Multimedia Networking

Cristina Aurrecoechea, Andrew Campbell and Alexandros Eleftheriadis*
Center for Telecommunications Research
Columbia University, New York, NY 10027-6699.
{cris, campbell, eleft}@ctr.columbia.edu

Abstract

Many distributed audio and video applications exhibit robustness in adapting to fluctuations in the delivered quality of service (QoS). By trading off temporal and spatial quality to available bandwidth, or manipulating the playout time of continuous media in response to variation in delay, audio and video flows can be kept meaningful at the playout device with minimal perceptual distortion. In this paper we introduce*Dynamic QoS Management (DQM)* for the control and management of multi-layer coded flows operating in heterogeneous multimedia networking environments. Two key techniques are proposed to meet the challenge of supporting scalable flows over multimedia networks: i) *Dynamic Rate Shaping*, which is a novel source-based QoS filter for manipulating the rate of MPEG-coded flows, matching it to the available network resources while minimising the distortion observed at the receiver; and ii) an *Adaptive Network Service,* which offers "hard" guarantees to the base layer of multi-layer coded flows, and "fairness" guarantees to the enhancement layers based on a new bandwidth allocation technique called *Weighted Fair Sharing*.

Keywords

Scalable video flows, mutlimedia transport, dynamic QoS managment, QoS architecture

1 INTRODUCTION

The interplay between multi-layer coded flows Shacham (1992), end-to-end communication support Campbell (1994) and receiver-oriented Quality of Service (QoS) requirements Zhang (1995) is an interesting and active area of research Pasquale (1993) Delgrossi (1993) Hoffman (1993). The basic technique used by coders (e.g., MPEG1, MPEG2 and H261) for compression of audio-visual flows is to remove redundant information in the signal. Using creative design techniques that take into account the perceptual capabilities of the human aural and visual system, it is possible to eliminate substantial parts of the signal with little or no perceived loss of information.

The vast majority of continuos media applications exhibit robustness in adapting to fluctuations in end-to-end quality of service.The perceptible quality of service presented at the receiver constitutes a complex of dynamic quality of service management functions such as video and audio source coding, resource reservation, flow scheduling, and QoS filtering and adaptation. In this research we introduce the concept of *Dynamic QoS Management (DQM)* for control and management of multi-layer coded flows operating in heterogeneous multimedia networking environments. The motivation that underpins our scheme is to bridge the heterogenity gap that exists between applications, end-systems and networks - in support of multicast QoS management. QoS adaptors, QoS filters and QoS groups are key scalable objects used in resolving quality of service capability mismatch. QoS filters manipulate hierarchically coded flows as they progress through the communications system, QoS adaptors scale flows at the end-systems based on the flow's measured performance and user supplied QoS scaling policy, and QoS groups provide baseline quality of service for multicast communications. The focus of the work is driven by a) the special features of scalable video flows - in particular MPEG2, b) the needs of both scalable and single-layer video for transmission over multimedia networks such as ATM and CSMA/CD internetworks.

Two key techniques are proposed to meet the challenge of supporting scalable video over multimedia networks: *Dynamic Rate Shaping* and an *Adaptive Network Service* . Dynamic rate shaping is a novel source-based QoS filter for manipulating the rate of MPEG-coded flows, matching it to the available network resources while minimising the distortion observed at the receiver. For multi-layer coded flows the adaptive service offers "hard" guarantees to the base layer, and "fairness" guarantees to the enhancement layers based on a new bandwidth allocation technique called *Weighted Fair Sharing*.

The structure of the paper is as follows. In Section 2 we first motivate our approach by discussing some of the relevant end-to-end QoS challenges in transporting digital video flows over multimedia networks. We then discuss the main charcteristics of scalable flows in Section 3 - paying partilcular attention to MPEG-2. In Section 4. we present an application programming interface (API) for scalable flows, followed by a detailes of our adaptive service and rate based scheme in Section 5. This is followed by a description of our dynamic QoS management scheme in Section 6. DQM is in part built on existing techniques in the literature Shacham (1992) Pasquale (1993) Delgrossi (1993) Kanakia (1993) Jeffay (1992) Steenstrup (1994) in provinding end-to-end QoS management Campbell (1994) of digital video flows H.262 (1994) Kanakia (1993). Dynamic QoS Management is divided into three domains: (i) sender-oriented DQM, which provides novel algorithms for Dynamic Rate Shaping; (ii) receiver-oriented DQM, which provide delay jitter, late packet and bandwidth managment; and (iii) network-oriented DQM, which provides support for QoS filtering and adaptive service. Finally in section 7 we present status of the work and some concluding remarks

2 QoS CHALLENGES

The provision of quality of service support for multicast flows broadly encompasses: QoS-based routing, end-to-end resource reservation, flow scheduling, network support for multicast communication, group management, and importantly, resolving end-to-end hetrogenity issues which may be present. It is widely accepted that individual senders and receivers of multicast groups will have differing capabilities to produce and consume audio-visual flows respectively. Resolving heterogenity issues present in applications, end-systems and networks is a challenging research issue; moreover, it is the subject of this work.

The range of audio and video QoS requirements is likely to be very diverse and the capability of applications, end-systems and networks to handle continuous media flows is likely to be quite diverse too. For example multimedia conferencing may only require low resolution video but high resolution sound. Considering the end-system only, heterogeneity is present in: CPUs, I/O devices, storage capabilities, compression support, communication support, network interfaces, etc. - all place fundamental limits on the capability of end-systems to consume and generate digital video. End-systems are likely to be connected to a wide variety of networks which not only have differing bandwidth capabilities but also varying access delay, jitter and loss characteristics. For example, in the case of multipoint communications using mobile and wired links: since mobile links are typically lower in bandwidth capability than wired ones, by reducing the video rate at the base-to-mobile link, wired receivers can utilise the full bandwidth available to them without being compromised by the inability of the mobile end-systems to take the full signal. The same argument holds for heterogenous internetworks such as ATM interworking with CSMA/CD LANs; here flows experiences predictable quality of service comitment over ATM clouds and variable quality of service over the Ether. A key challenge is resolving potential quality of service mismatch. We suggest that *QoS filtering* is central in bridging this heterogenity gap Delgrossi (1993) and fundemental in support of multicast flows [].

Another challenge is the transmission of variable bit rate video over multimedia networks. In order to derive algorithms for supporting variable bit rate network transmission (e.g., admission control, resource reservation and packet scheduling) of digital video, significant effort has been expended to obtain statistical models for video traffic Lazar (1994). The problem is extremely complex, since the rate is intimately related to both the coding scheme and the actual source material through the use of entropy coding: video sequences that deviate from the "norm" will require more bits than the average, and may exhibit highly variable behaviour.

As a result, models tend to be quite complex for variable bit rate video, and there is currently no known universally acceptable one. The drawback of inadequate modeling is that it can result in either loss of information (by underestimating the traffic behaviour) or to wasteful use of network resources (through overestimation) or reliance on constant bit rate "circuits". Some general observations have been made regarding coded digital video Kanakia (1993): (i) burstiness is heavily dependent on the content of the image being coded and the coding algorithm used; (ii) source generation rates are highly sensitive to scene and background changes; and (iii) highly correlated source traffic is potentially persistent over very long periods. All this makes modeling video in a general sense extremely difficult and complex. Because modeling video sources is so difficult, and constant bit rate channels so inefficient in exploiting statistical multiplexing we focus our attention in this paper on an alternative approach: an adaptive channel, which supports *QoS adaptation* through the semantics of scalable video flows Delgrossi (1993) Kappner (1994) and resource fairness Kappner (1994) Tokuda (1992).

The degree of resource reservation required by audio-visual flows is an open reserach issue, however. Until recently it was argued that only constaint bit rate channel could accomodate isochronous communications. In constrast, some argue that multimedia flows are continuously adaptive and do not require any resource reservation at all. Furthermore, others suggest that there is a minimun QoS threshold below which the signal is meaningless at the receiver. Whatever the phiosophy (from the hard or soft school of end-to-end QoS) it is clear that audio can not operate below minimum bandwidth thresholds; and video can degrade to a still picture - but then this would not be video anymore! The view taken in this work is to provide support for multi-layed flows where the base signal is guaranteed and enhancements made available based on the availablity of resources; essestially to take an imtermediate position as regards the "reservation" issues. To meet this challenge we introduce the concept of an adaptive service network which understands the semantics of multi-layed flows []

In all the above cases, the common theme which runs through the work is the need to scale flows to match end-to-end resource availability. To meet the above challenges we introduce a number of QoS mechanisms for the manipulation of audio and video flows. Collectively, we describe the protocol which instantiates, controls and manages scalable flows as dynamic QoS management. DQM QoS mechanisms comprises of QoS filters, which manipulate multi-layered flows as they progress through the communications system; and QoS adapors, which scale flows at the end-systems. The role of adaption and filtering is collectively refered to as *QoS scalability* in DQM, and is governed by the dynamics of resource availability (cf. adaptive network service) and a user suplied *QoS scalability policy*.

3 SCALABLE VIDEO FLOWS

The fundamental principle of digital audio-visual information representation in the past several decades has been that of compactness: describe the signal's content with as few bits as possible. The algorithmic foundation has been built on the assumption that information transport occurs over constant bandwidth and constant delay channels. Current and future communication and application environments invalidate these assumptions through the use of packet-based systems, and have generated a lot of interest in the resultant interoperability issues from both the networking and video coding sides. From the networking point of view, flexibility is desirable for the realisation of statistical multiplexing gains in terms of network resource utilization. From the traditional video coding side, flexibility requirements generate severe problems because algorithms are typically designed for maximum coding efficiency: they use very specific design parameters, and employ carefully chosen constraints. Although techniques have been proposed to provide flexibility, or adaptability, to live compressed video [Kanakia,93][Eleftheriadis,94], no techniques have been presented for stored video applications.

Our primary focus in this paper is MPEG-2 coded video. Within this framework, there are two alternatives techniques which underpin QoS adaptation: *intrinsic* and *extrinsic* techniques. The former are provided by the encoder, and are embedded in the coded bitstream. The latter are provided by QoS filters that operate directly on the compressed bitstream, performing the desired manipulations. Their difference lies in their complexity and performance. Intrinsic techniques can have very simple implementation, but as we will see offer only a discrete range of possibilities. Extrinsic techniques are computationally more complex but can operate on a continuum of constraints. In the following we briefly describe the architecture of MPEG-2, with particular emphasis on the intrinsic adaptation capabilities it provides in the form of scalability profiles. Extrinsic adaptation can be provided through the use of dynamic rate shaping QoS filters, which are discussed in section 4.4.

3.1 MPEG-2

The algorithmic foundation of MPEG is motion-compensated, block-based transform coding [MPEG,94]. Each block is transformed using the Discrete Cosine Transform (DCT), and is subsequently quantized. Quantization is the sole source of quality loss in MPEG, and of course a major source of compression efficiency. The quantized coefficients are converted to a one-dimensional string using a zig-zag pattern and then run-length encoded. There are three types of pictures in a sequence: I, P, and B. I or intra pictures are individually coded, and are fully self-contained. P pictures are predicted from the previous I or P picture, while B (or bi-directional) pictures are interpolated from the closest past and future I or P pictures. Prediction is motion-compensated: the encoder finds the best match of each macroblock in the past or future picture, within a prespecified range. The displacement(s), or motion vector(s), is sent as side information to the decoder.

In order to increase the coding efficiency, MPEG relies heavily on entropy coding. Huffman codes (variable length codewords) are used to represent the various bitstream quantities (run-length codes, motion vectors, etc.). As a result, the output of an MPEG encoder is inherently a variable rate bitstream: the ratio of bits per pixel varies from one block to the next. In order to construct a constant bit rate bitstream, rate control is used. This is achieved by connecting a buffer to the output of the encoder. The buffer is emptied at a constant rate (the channel rate), and its occupancy is fed back to the encoder. This information is used to control the selection of the quantizer for the current macroblock. High buffer occupancy leads to more coarsely quantized coefficients, and hence less bits per block, and vice versa. Through this self-regulation technique one can achieve a constant output rate; clever design is needed in order to avoid buffer overflows which lead to loss of information, or underflows which can lead to loss of synchronisation in certain channels.

For transport purposes, video and audio are multiplexed according to the system layer of MPEG, which defines the packetization structure and synchronisation algorithms between the audio and video signals. Two different packetization structures are defined, namely program streams and transport streams. Both are logically constructed from Packetized Elementary Stream (PES) packets; these are the basic units in which individual audio, video, and control information is carried. Program streams are designed for use in relatively error-free environments, use variable length packets, and combine PES packets that have a common time base. Transport streams are designed for noisy channels, utilise fixed-length packets (188 bytes), and can carry programs with independent time bases. The system layer's timing model is based on the assumption that a constant delay transport mechanism is used. Although deviations from this are allowed, the way to address them are not specified. All timing information is based on a common system clock, and timestamps ensure proper inter-media synchronisation between the audio and video signals upon presentation.

From the above discussion it becomes clear that the current design is especially tuned for transmission over traditional constant bandwidth and delay channels. Some support for flexible and/or robust transmission is provided through the use of scalability modes for channels exhibiting dynamic behaviour. The situation is even more challenging for scalable flows, i.e. in cases where the available bandwidth may vary over time. However, here benefits of multiplexing gain are possible.

Scalable modes

MPEG2 H.262 (1994) provides for the simultaneous representation of a video signal at various different levels of quality, through the use of multiple independent bitstreams or sub-signals. This is achieved through the use of pyramidal, or hierarchical coding: one first constructs a coarse or base representation of the signal, and then produces successive enhancements. The latter assume that the base representation is available, and only encode the incremental changes that have to be performed to improve the quality. There are four different scalability modes: spatial, SNR, temporal, and data partitioning. MPEG-2 allows the simultaneous use of up to two different scalability modes (except from data partitioning) in any combination, hence resulting in a 3-level representation of the signal. In spatial scalability, the base and enhancement layers operate at different spatial resolutions (e.g., standard TV and HDTV). In SNR scalability, both layers have the same resolution and the enhancement refines the quantization process performed in the base layer. In temporal scalability the enhancement layer increases the number of frames per second of the base layer (e.g., from 30 frames per second to 60 fps).

Data partitioning is slightly different from the other three, in the sense that the encoder does not maintain two different prediction loops and hence the base layer is not entirely self-contained. Its benefit is that it can be applied even in single-layer encoders. The primary purpose of scalability modes is to provide flexibility, scalability and robustness.

As an example, the spatial scalability mode can be used to transmit digital TV in both standard and HDTV formats. Although two sub-signals are actually generated, this is not identical to simulcast: the total bandwidth required is much smaller, since the HDTV layer uses the base, standard TV layer as a reference point. Scalability can be very useful in transmission of video over adaptive channels.

3.3 QoS adaptation

Although MPEG's scalability features are useful in resolving heterogeneity problems described above Delgrossi (1993), and are useful in numerous applications, their use in continually QoS-varying channels is problematic. This is because they intrinically only allow the representation of the signal at a number of discrete quality points (temporal or spatial resolution, or spatial quality). These points are typically significantly apart, and transitions between the two are perceptually significant. Table 1 Paek (1995) shows an example of hybrid scalability with spatial (E1) and SNR (E2) enhancement layers.

Table 1 MPEG-2 hybrid scalable bitstream using spatial and SNR scalability (24 fps)Paek (1995)

layer name	*profile*	*symbol*	*frame size*	*bit rate*	*subjective QoS*
base layer	main	BL	304x112	0.32 Mbps	VHS
enhancement 1 layer	spatial	E1	608x224	0.83 Mbps	super VHS
enhancement 2 layer	SNR	E2	608x224	1.85 Mbps	laser disc

Consider for example a channel that temporarily sustains rate variability for a period of a few seconds. Switching to a lower quality point (by discarding the enhancement layer(s)) for such a brief interval will essentially create a "flash" that is very annoying to viewers (we call this *discrete adaptation* in DQM). An additional issue is that, as soon as compression parameters are established, it is impossible to modify them later on (after compression is completed). Hence scalability modes can be used for well-defined, simple channels that vary slowly. Since the variety of different access mechanism to multimedia information makes it very difficult to select a priori a set of universally interoperable coding parameter, it is necessary to provide extrininic mechanisms that allow the representation of the "signal at a continuum of qualities and rates" (we call this *continuous adaptation)* Delgrossi (1993), so that scalable flows can be accommodated. This is possible through the use of dynamic rate shaping QoS filters Eleftheriadis (1995) and the provision of adaptive network services - providing a QoS continuum for fully scalable flows. The adaptive service introduced in this paper uses explicit feedback from network resource management to dynamically shape the video source based on the available network resources. Some benefits of an adaptive scheme are non-reliance on video modeling techniques and statistical QoS specification and specific support for the semantics of scalable video flows e.g., MPEG scalable profiles. Dynamic rate shaping filters manipulate the rate of MPEG-coded video, matching it to the available bandwidth (indicated by the adaptive service) while minimising the distortion observed by the receiver.

4 APPLICATION PROGRAMMING INTERFACE

4.1 Scalable objects

QoS adaptors, QoS filters and QoS groups are key scalable objects in dynamic QoS management. These objects are used to resolve quality of service capability mismatches in the end-systems and network and provide communication support for single and scalable video flows. QoS adaptation relates to the monitoring (using *flow monitors*) and adjustment (using QoS adaptors) of flows at the edge of the network to ensure that the user and provider quality of service is maintained. In this role QoS adaptors are seen as quality of service arbiters between the user and network. QoS adaptors scale flows at the end-systems based on a user supplied QoS scaling policy (see section 4.2) and the measured performance of on-going flows. In contrast to adapting flows at the end-systems, QoS filters manipulate hierarchically coded flows Shacham (1992) Hoffman (1993) at the end-systems and as they progress through communications systems. In dynamic QoS management we refer to *scaling* Delgrossi (1993) Kappner (1994) as an "umbrella" term to cover the combination of QoS adaptation in end-systems, and QoS filtering Pasquale (1993) Campbell (1994) in end-systems and the network. The QoS scaling policy is central to DQM and is the driver of QoS adaptation and QoS filtering mechanisms for end-to-end QoS management.

We describe three styles of QoS filters in dynamic QoS management:(i) *shaping filters*, which manipulate coded video and audio by exploiting the structural composition of flows to match network, end-system or application QoS capability. Shaping filters are generally situated at the edge of the network; (ii) *selection filters*, which are used for sub-signals source selection and media dropping are of low complexity and low computational intensity. Selection filters are designed to operate in the network and are located at switches; and (iii) *temporal filters*, which manipulate the timing characteristics of media to meet delay bound QoS are also low in complexity and trivial computationally. Temporal filters are generally placed at receivers or sinks of continuous media were jitter compensation or orchestration of multiple related media Campbell (1994) is required.

The first and third types of QoS filters are predominantly located in the edges of the network. Shaping filters utilise knowledge of the coding details of the flow they are processing, and require non-trivial computational power. Selection filters, on the other hand, perform simple packet filtering and hence can be located in internal network nodes. In some cases, shaping filters can be located in special network nodes, either as a bearer service, or as part of special environments such as mobile communication links. As an example, rate shaping filters placed at base stations can be useful for multi-point communication Shacham (1992) with both mobile and wired hosts. Through the use of rate shaping, wired users can utilise the full bandwidth at their disposal, without compromising quality to that attainable to the least capable link of the session. A distributed object-based facility (e.g. CORBA OMG (1993)) can be particularly effective in providing the foundation for the incorporation of filters throughout the networking infrastructure (see section 6.) Kappner (1994).

Before receivers or senders bind Lazar (1994) media source and sink devices, media protocols and scalable objects to form end-to-end connections they must first join a QoS group Aurrecoechea (1994). A flow is represented by a QoS group in our adaptive (CORBA-based) environment. Group management announces QoS group capability in terms of its quality of service capability. The concept of a QoS group is used to associate a baseline quality of service capability to a particular flow. All sub-signal of a multi-resolution stream can be mapped into a single flow and multicast to multiple receivers Shacham (1992). Each receiver can select to take the complete signal advertised by the QoS group or a partial signal based on resource availability. Alternatively each sub-signal can be associated with a distinct QoS group. In this case receivers "tune" into different QoS groups (using signal selection) to build up the overall signal. Both methods are supported in DQM. Receivers and senders interact with QoS groups to determine what the baseline service is, and tailor their capability to consume the signal by selecting filter styles and specifying the degree of adaptability sustainable (viz. discrete, continuous).

4.2 QoS specification

In Campbell (1994) we formalised the end-to-end QoS requirements of the user and the potential degree of service commitment of the provider in a service contract. In this work, we focus on the extensions to the flow specification, QoS commitment and QoS scaling clauses required to accommodate the special needs of adaptive multi-layer flows over multimedia networks. Multi-layered flows are characterised by three sub-signals in the flow specification: a base layer (BL), and up to two enhancement layers (E1 and E2, respectively). Each layer is represented by a frame size, bit rate and subjective or perceptive quality of service as illustrated in Table 1. Based on these characteristics the MPEG2 coder Paek (1995) Eleftheriadis (1995) determines approximate bit rate for each sub-layer. In the case of MPEG-2's hybrid scalability, BL would represent the main profile bit rate requirement (e.g., 0.32 Mbps) for basic quality, E1 would represent the spatial scalability mode bit rate requirement (e.g., 0.83 Mbps) for enhancement, and finally E2 would represent the SNR scalability mode bit rate requirement (e.g., 1.85 Mbps) for further enhancement. For full details of deriving these bit rates. The remaining flow specification performance parameters for jitter, delay and loss are assumed to be common across the all sub-signals. We use the term *sub-signal* to represent a single layer of a multi-layer video flow; and the term *flow* as a non-assured simplex uni-media stream, comprising of one source and potentially many receivers; flows always have end-to-end QoS associated with them Campbell (1994).

```
typedef struct {                typedef   struct {
      gid         flow_id;            adaptMode   adaptation;
      commit      commitment;         filterStyle filtering;
      mediaType   media;              events      adaptEvents;
      bitRate     BL;                 actions     newQoS;
      bitRate     E1;                 signal      bandwidth;
      bitRate     E2;                 signal      loss;
      int         delay;              signal      delay;
      int         loss;               signal      jitter;
      int         jitter;         } QoSscalingPolicy;
} flowSpec;

typedef enum {MPEG1, MPEG2, H261, JPEG} mediaType;
typedef enum {drs,sbr,sub_signal,hierarch,hybrid,sync,orch} filterStyle;
typedef enum {besteffort, adaptive, deterministic} commit;
typedef enum {continuous, discrete} adaptMode;
```

The scaling policy characterises the degree of adaptation that the flow can tolerate and still achieve meaningful QoS. The scaling policy has been extended to capture the special needs of multi-layer flows, and includes adaptation modes, QoS filter styles, and user level notifications options for bandwidth, delay, jitter and loss QoS signals. Two types of adaptive mode are supported Delgrossi (1993): continuous mode, for applications that can exploit any availability of bandwidth above the base layer; and discrete mode that is suitable for applications which can only accept discrete improvement in bandwidth based on a full enhancement (viz. E1, E2). The adaptive modes option covers both highly adaptive (e.g., MPEG2 using dynamic rate shaping) and coarsely adaptive (e.g., MPEG2 scalable profiles).
The QoS scaling policy provides user-selectable QoS adaptation and QoS filtering. QoS filters are broadly divided into source-based (viz., drs-filter and sbr-filter - see section 6.1), network-based (viz., sub_signal-filter, hierarch-filter, hybrid-filter - see section 6.3), and receiver-based (viz., sync-filter, orch-filter - see section 6.2) filters. While receivers select filter styles to match their capability to consume media at the receiver, senders select filter styles to shape flows in response to the availability of network resources such as bandwidth and delay. Receivers and senders can select periodic performance notifications including available bandwidth, measured delay, jitter and losses for an on-going flows. The QoS signal field in the scaling policy allows the user to specify the interval over which a QoS parameter is to be monitored and the user informed. Both single and multiple quality of service signals can be selected depending on the applications needs. For full details on the service contract see Campbell (1994). In addition, the QoS commitment clause has been extended to offer an adaptive network service that specifically caters for the needs of scalable audio and video flows.

5 ADAPTIVE NETWORK SERVICE

The adaptive service provides "hard" guarantees to the base layer (BL) of a multi-layer flow and *Weighted Fair Share (WFS)* to each of the enhancement layers (E1 and E2). To achieve this, the base layer undergoes a full end-to-end admission control test Coulson (1995). On the other hand, enhancement layers are admitted without any such test but must compete for *residual bandwidth* among all other adaptive flows. Enhancement layers are rate controlled based on explicit feed back about the current state of the on-going flow and the availability of residual bandwidth.

5.1 Service goals and requirements

Both end-system and network communication resources are partitioned between the deterministic and adaptive service commitment classes. This is achieved by creating and maintaining "firewall" capacity regions for each class. Resources reserved for each class, but not currently in use can be borrowed by the best effort service class on condition of pre-emption Coulson (1995). The adaptive service capacity region (called the available capacity region and denoted by B_{avail}) is further sub-divided into two regions: (i) guaranteed capacity region (B_{guar}), which is used to guarantee all base rate layer flow requirements; (ii) and residual capacity region (B_{resid}), which is used to accommodate all enhancement rates were competing flows share the residual bandwidth.

Three goals motivate our adaptive service design: First, to admit as many base layer (BL) sub-signals as possible. As more base layers are admitted the guaranteed capacity region B_{guar} grows to meet the hard guarantees for all base signals. In contrast, the residual capacity region B_{resid} shrinks as enhancement layers compete for diminishing residual bandwidth resources. The following invariants must be maintained at each end system and switch:

$$B_{avail} = B_{guar} + B_{resid} \text{ , and } \sum_{i=1}^{N} BL_{(i)} \leq B_{avail} \qquad (1)$$

Second, to share Steenstrup (1994)Tokuda (1992) the residual capacity B_{resid} among competing enhancement sub-signals based on a flow specific *weighted factor* (W_{fact}), which allocates residual bandwidth in proportion to the range in bandwidth requested that in turn is related to the range of perceptual QoS acceptable to the user. In DQM, residual resources are allocated based on the range of bandwidth requirements specified by the users (i.e., BL.. BL+E1+E2 is the range of bandwidth required e.g., from 0.32 Mbps to 3 Mbps for the hybrid scalable MPEG2 flow in Table 1). As a result, as resources become available each flow experiences the same "percentage increase" in the perceptible QoS, we call this weighted fair share. W_{fact} characterises the notion of WFS (see (2)) and is calculated for each flow as the ratio of a flow's perceptual QoS range to the sum of all perceptual QoS ranges.

$$W_{fact(i)} = (BL_i + E1_i + E2_i) \,/\, \sum_{j=1}^{N} (BL_j + E1_j + E2_j) \qquad (2)$$

All residual resources B_{resid} are allocated in proportion to $W_{fact(i)}$ metric. Using this factor we calculate the proportion of residual bandwidth allocated to a flow to be $B_{wfs(i)}=W_{fact(i)}B_{resid}$ and the proportion of the available bandwidth allocated to be $B_{flow(i)}= B_{wfs(i)}+ BL_{(i)}$. We describe this aggregate as the flow bandwidth (B_{flow}).

Third, to adapt flows both discretely and continuously based the adaptation mode. In the discrete mode no residual bandwidth is allocated by the WFS mechanism unless a complete enhancement can be accommodated (i.e., $B_{wfs(i)}$ = E1(i) | E1(i)+E2(i) e.g., 0.83 Mbps or 2.68 Mbps from Table 1). While in continuous mode any increment of residual bandwidth $B_{wfs(i)}$ can be utilised (i.e., $0 < Bwfs(i) \leq E1(i) + E2(i)$ e.g., from 0 to 2.68 Mbps from Table 1). Adaptive applications can be considered to be either coarsely (e.g., MPEG-2 scalable profiles) or highly adaptive (e.g., scalable MPEG2 using a dynamic rate shaping). By selecting continuous adaptation highly adaptive applications can take advantage of any availability in bandwidth to enhance QoS. While coarsely adaptive applications are more suited to the discrete mode were only E1 and E2 sub-signals can be accommodated, nothing more nothing less. In addition, WFS always accommodates the full E1 signal before attempting to deliver the E2.

5.2 Rate based scheme

We build on the rate-based scheme described in Campbell (1994) were the QoS-A transport protocol at the receiver measures the bandwidth, delay, jitter and loss over an interval which we call an "era". An era is currently defined as simply the reciprocal of the frame rate in the flow specification (e.g., for a frame rate of 24 frames per second as shown in Table 1 the interval era is approx. 42 ms.). The receiver-side transport protocol periodically inform the sender-side about the currently available bandwidth, and measured delay, loss and jitter. This rate control information is used by the source or virtual source (see later) as the rate over the next interval. The reported rate is temporally correlated with the on-going flow. An important result in Kanakia (1993) shows that variable rate encoders can track quality of service variations as long as the QoS feed back is within four frame times or less. This feed-back is used by the dynamic rate shaping filter and network based filters to control the data generation of the video or the selection of the signal respectively. In the case of dynamic rate shaping the rate is adjusted while keeping the perceptual quality of the video flow meaningful to the user.

Based on the concept of eras, control messages are forwarded from the receiver-side transport protocol to either virtual source or the source-side transport using reverse path forwarding. We use the term virtual source to represent a network switch that modifies the source flow via filtering. A core-switch Ballardie (1993) Aurrecoechea (1994) were flows are filtered is always considered to be a virtual source for one or more receivers. The WFS mechanism updates the advertised rate as the control messages traverse the switch on the reserve path to the source or virtual source. Therefore any switch can adjust the flow's advertised rate before the source or virtual source receives the rate based control message. The source-side transport hands the measured delay and aggregate bandwidth off (B_{flow}) to the dynamic rate shaping filter.

DQM maintains *flow state* at each end-system and switch that a flow traverses. Flow state is updated by the WFS algorithm and the rate-based flow control mechanism. Flow state maintained in the network constitutes: (i) capacity (viz. B_{avail}, B_{guar}, B_{resid}); (ii) policy (viz. filterStyle, adaptMode); (iii) flowSpec (viz. BL, E1, E2) ; and (iv) wfsShare (B_{flow}, B_{wfs}, W_{fact}). The end-systems holds an expanded share turple for measured delay, loss and jitter metrics. An admission control test is conducted at each end-system and switch on route to the core for base layer signal. This test simply determines whether there is sufficient bandwidth available to guarantee the base layer BL given the current network load:

$$\sum_{j=1}^{N} BL_{(j)} \leq B_{avail} \tag{3}$$

If the admission control test is successful, WFS determines the additional percentage of the residual bandwidth made available B_{wfs} to meet any enhancement requirements in the flowSpec:

$$B_{wfs(i)} = W_{fact(i)}.(B_{avail} - \sum_{j=1}^{N} BL_{(j)}) \tag{4}$$

The WFS rate computation mechanism can causes new B_{wfs} rates to be computed for all adaptive enhancement signals that traverse the output link of a switch; switches are typically non-blocking which means the critical resource are the output links, however, our scheme can be generalised to other switch architecture Coulson (1995).

6 DYNAMIC QoS MANAGEMENT

Dynamic QoS management, illustrated in Figure 1, is broadly divided into three "middleware" domains (which are represented as "slices" in the diagram) for end-to-end dynamic QoS management: (i) *sender-oriented DQM,* senders select source filters (i.e. drs, sbr filters) and adaptation modes, and setup flow specifications for video and audio communications. The sender-side transport protocol provides periodic bandwidth and delay assessments to the dynamic rate shaping filter which regulates the source flow. Senders creates QoS groups which announce the quality of service of the flow to receivers via QoS group management Aurrecoechea (1994); (ii) *network-oriented DQM,* provides the adaptive service to receivers and senders by guaranteeing the base signal and provides weighted fair share using a novel rate base flow control mechanism to switch in discrete or partial enhancements. Network level QoS filters (i.e., sub-signal, hierarch and hybrid-filters) are instantiated based on the user selection, and propagated in the network under the control of filter managementYeadon (1994); and (iii) *receiver-oriented DQM,* receivers join QoS groups and select the portion of the signal which matches their QoS capability. Receiver selected filters propagate in the network for source and signal selection. In addition, receiver-based QoS filters (i.e., sync-filter and orch-filter) are instantiated by default unless otherwise directed. These filters are used to smooth and synchronise multiple media. The receiver-side transport provides essential bandwidth management for enhancement announcements; delay-jitter and late packet management trading off timeliness against loss.

Based on the receiver supplied scaling policy, QoS adaptors can take remedial actions to scale flows, inform the user of a QoS indication and degradation, fine tune resources and initiate complete end-to-end QoS renegotiation based on a new flowSpec Campbell (1994). The QoS scaling policy also allows the user to modify existing QoS filters; and based on this policy, filter management Yeadon (1994) installs new filters at optimal points in the media path.

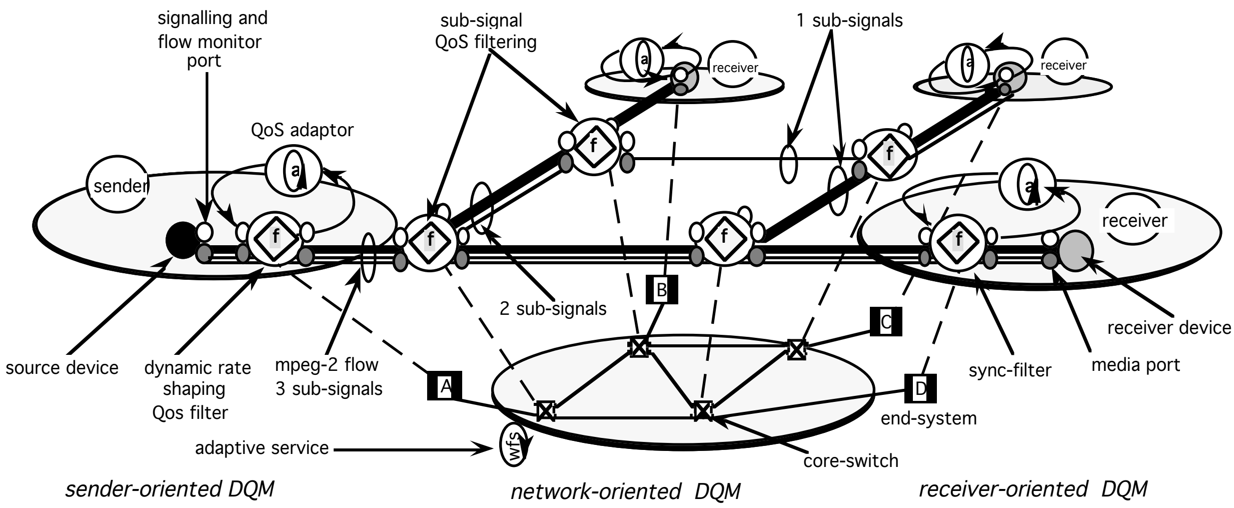

Figure 1 Dynamic QoS management of scalable flows

Components of a distributed adaptive environment consist of a number of algorithms for QoS group management, flow management, filter management (we neglect to illustrate the interaction of these components with senders and receivers in Figure 1 for reasons clarify) and dynamic QoS management. Each element provides a set of interfaces and methods to manage flows in multimedia networks. Communication between interfaces is based on CORBA running over AAL5/ATM in our distributed environment at Columbia Lazar (1994) Aurrecoechea (1994) Figure 1 illustrates how adaptive receivers (at nodes B, C and D) and senders (node A) are built on top of multimedia networks which consist of Fore Systems ASX200 ATM switches. The middleware includes the sender, network and receiver-oriented DQM infrastructure and represents the system software components lying in the region between the switching and transmission firmware and specific multimedia applications. CORBA OMG (1993) runs on the end-systems and in the ATM switches, providing a seamless object oriented environment where filters, adaptors, QoS group manager, flow monitors and flow managers can propagate Pasquale (1993) Yeadon (1994) throughout the communication system base. In DQM each sub-signal (i.e. base and enhancement layers) can be carried as sub-signal multiplexed onto a single flow, or independently by distinct flows. DQM can handle either case, and leaves it up to the receivers and senders to determine which approach is more suitable.

In Figure 1 a sender at end-system A creates a flow by instantiating a QoS group which announce the characteristics of the flow and its adaptation mode i.e., for MPEG-2 in Table 1 (viz. layer, frame size, bit rate, subjective quality) for BL, E1 and E2 respectively. Receivers join the QoS group. In the example scenario shown in Figure 1 three end-systems join the QoS group created by sender A and "tune" into different parts of the multi-layer signal. The example shows B taking BL the main profile (which constitutes a bandwidth of 0.32 Mbps for VHS perceptual QoS), C taking BL and E1 (which constitutes an aggregate bandwidth of 1.15 Mbps for super VHS perceptual QoS), and D taking the complete signal BL+E1+E2 (which constitutes an aggregate bandwidth of 3 Mbps for laser disc perceptual QoS). In this example the complete signal is multiplexed onto a single flow, therefore, sub-signal selection filters are propagated by filter management Yeadon (1994). Receivers, senders, or any third party or filter management object can select, instantiated and modify source, network and receiver-based QoS filters.

6.1 Sender-oriented DQM

A sender-oriented end system architecture illustrated in Figure 2 shows the functions of the sender-side transport that support dynamic QoS management and the interface to a dynamic rate sharing filters. Currently senders can select from two types of shaping filter at the source: dynamic rate shaping (DRS) and soure bite rate (SBR) QoS filters. Both of these QoS filters adapt the signal to meet the available bandwidth by keeping the signal meaningful at the receiver or core. The sender-side transport mechanisms includes a QoS adaptor, flow monitor and media scheduler. Bandwidth updates are synchronously received by the flow monitor mechanism from the network as part of the adaptive service (described in section 5). The QoS adaptor is responsible for synchronously informing the drs-filter of the current bandwidth availability (B_{flow}) and measured delay (D_{flow}), and calculating new schedule and deadliness for transport service data units Coulson (1995). Media progresses from drs-filter at the TSAP, and is scheduled by the media scheduler to the network at the NSAP based on the calculated deadlines.

The QoS adaptor is responsible for informing the sender of the state of the on-going QoS based on options selected in the QoS scaling policy. As in the receiver side, senders can request periodic updates of bandwidth, delay, loss and jitter. Alternatively, senders can select the QoS monitoring selection Campbell (1994) to receive periodic updates of all QoS parameters as part of the QoS maintenance operation. Senders, therefore, have the option to discretely select specific QoS parameters, or a set, or all QoS parameters as in the case of QoS assessments Campbell (1994). Informing the application of the current state of the resources associated with a specific flow is key in implementing adaptive application in end-systems. In this case the sender is simply used to managed the flow by receiving updates and interacting with the QoS adaptor to adjust the flow e.g., change adaptation mode from continuous to discrete, request more bandwidth for BL, E1 and E2, or change the characteristics of the source filter, etc.

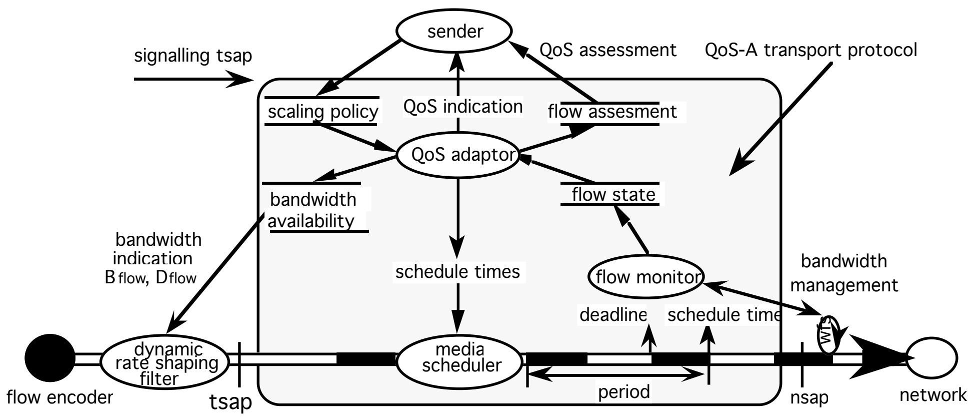

Figure 2 Sender-side transport QoS mechanisms

Dynamic rate shaping QoS filter

Dynamic rate shaping of compressed digital video Eleftheriadis (1995) (as shown in Figure 2) is a technique to adapt the rate of compressed video bitstreams (MPEG1, MPEG2, H261, as well as JPEG) to dynamically varying rate (and delay) constraints. The approach provides an interface (or filter) between the encoder and the network, with which the encoder's output can be perfectly matched to the network's quality of service characteristics. Although a number of techniques have been developed for the control of live source (which call source bit rate filters) Kanakia (1993) Eleftheriadis (1995) they cannot be used for the transmission of pre-compressed material (e.g., in on-demand video systems). Dynamic rate shaping filters do not require interaction with the encoder and hence are fully applicable to both live and stored video applications. The DRS-filter operate directly in the compressed domain of the video signal, manipulating the bitstream so that rate reduction can be effected.

By providing decoupling of the encoder and the network, universal interoperability can be achieved both between codecs and networks, and also among codecs of different specifications. An attractive aspect is the existence of low-complexity algorithms which allow even software-based implementation in high-end computers. In order for rate shaping to be viable it has to be implementable with reasonable complexity and yield acceptable visual quality. With respect to complexity, the straightforward approach of decoding the video bitstream and recoding it at the target rate would be obviously unacceptable; the delay incurred would also be an important deterrent. Hence only algorithms of complexity less than that of a cascaded decoder and encoder are of practical interest. These algorithms operate directly in the compressed domain of the video signal, manipulating the bitstream so that rate reduction can be effected. In terms of quality, it should be noted that recoding does not necessarily yield optimal conversion; in fact, since an optimal encoder (in an operational rate-distortion sense) is impractical due to its complexity, recoding can only serve as an indicator of an acceptable quality range. In fact, regular recoding can be quite lacking in terms of quality, with DRS providing significantly superior results. In the following be briefly outline the operation of a dynamic rate shaper.

We define rate shaping as an operation which, given an input video bitstream and a set of rate constraints, produces a video bitstream that complies with these constraints. For our purposes, both bitstreams are assumed to meet the same syntax specification, and we also assume that a-possibly motion-compensated-block-based transform coding scheme is used. This includes both MPEG-1 and MPEG-2, as well as H.261 and so-called "motion" JPEG, although in this paper our focus is primarily on MPEG.

The rate shaping operation is depicted in Figure 6. Note that no communication path exists between the rate shaper and the source of the input bitstream, which ensures that no access to the encoder is necessary. Of particular interest is the source of the rate constraints $B_{flow(t)}$. In the simplest of cases, $B_{flow(t)}$ may be just a constant and known a priori (e.g. the bandwidth of a circuit-switched connection). It is also possible that $B_{flow(t)}$ has a well (a priori) known statistical characterisation (e.g. a policing function). Finally, another alternative is that $B_{flow(t)}$ is generated by the network over which the output bitstream is transmitted; this is provided by the adaptive service mechanism described in Section *.*. In this paper we are particularly interested in the case where the information is provided by the send-side transport based on the DQM adaptive service.

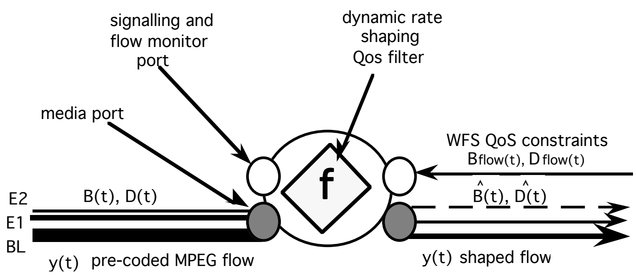

Figure 6 Dynamic rate shaping scheme

The objective of a rate shaping algorithm is to minimise the conversion distortion, i.e.:

$$\min_{B(t)<B_T(t)} \left\| y(t) - \hat{y}(t) \right\|$$

The attainable rate variation $(\hat{B}/B)$ is in practice limited, and depends primarily on the number of B pictures of the bitstream; no assumption is made on the rate properties of the input bitstream, which can indeed by arbitrary.

Assuming that a motion-compensated block-based transform coding technique is used to generate the input bitstream and decode the output one, there are two fundamental ways to reduce the rate: (i) modifying the quantized transform coefficients by employing coarser quantization, and (ii) eliminating transform coefficients. In general, both schemes can be used to perform rate shaping; requantization, however, leads to recoding-like algorithms which are not amenable to fast implementation and do not perform as well as selective-transmission ones. The selective transmission approach gives rise to a family of different algorithms, that perform optimally under different constraints [].

6.2 Receiver-oriented DQM

Receiver-oriented adaptation can be broken down into a number of receiver-side transport functions: (i) delay-jitter management, which calculates flow playout points based on the actual measured delay-jitter from the network; (ii) late-frame management, which monitors late arrivals in relation to the loss metric and the current playout times and takes appropriate action to trade of timeliness and loss; and (iii) bandwidth management, which receives bandwidth indications in the control message portion of the TSDU and adapts the receiver appropriately. In essence the transport protocol "controls" the progress of the media while the receiver "monitors and adapts" to the flow based on the flow specification and the scaling policy

QoS adaptors, which are resident at senders and receivers, are transport-based QoS adaptation managers that arbitrate between the receiver specified QoS and the monitored QoS of the on-going flow. When the transport is in monitoring mode Campbell (1994) the flow monitor uses an absolute timing method to determine frame receptions times based on timestamps Jeffay (1992) Shenker (1993). The flow monitor, as shown in Figure 3, updates the flow state to include these measured reception times statistics. Based on these flow statistics, the sync-filter derives new playout times which are used by the media scheduler to adjust the playout point of the flows to the decoding delivery device.

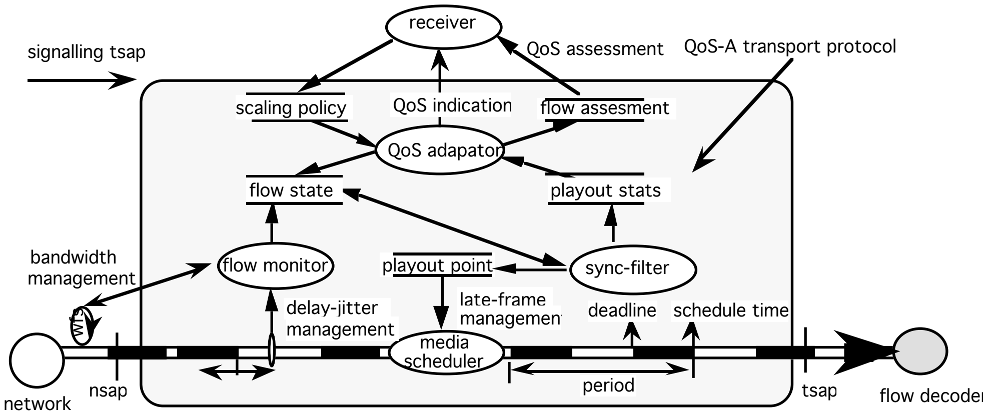

Figure 3 Receiver-side transport QoS mechanisms

Packets that arrive after their expected playout points are discarded by the media scheduler and the late-packet metrics in the playout statistics are updated. The media scheduler simply discards late packets which have missed their schedule time. The media scheduler is based on a split-level scheduler architecture Coulson (1995) which provides hard deadline guarantees to base layer flows via admission control, and best effort deadlines to the enhancements. Some remedial action may be taken by the QoS adaptor should the loss metric exceed the loss parameter in the flow specification. If the QoS adaptor determines that too many packet losses have occurred over an era, it pushes out the playout time to counter act the late state of packets from the network. Similarly, if loss remains well within the prescribed ranges then the QoS adaptor will automatically and incrementally "pull in" the playout time until loss is detected.

Another important receiver-side transport function is bandwidth management. The adaptive service built on the notion of WFS, periodically informs the receiver that more bandwidth is available or announces that the flow is being throttled back. Bandwidth management only covers the enhancement signals of multi-resolution flows. The baseline is not included since resources are guaranteed to the baseline and shared only amongst the enhancements. The announcement of available bandwidth on a flow allows the receiver to take either a full or partial enhancement layer. This depends on whether the flow is in continuous or discrete adaptation mode.

Sync QoS filter

In scalable MPEG2 coding, sources (or virtual sources) produce a base layer signal and up to two enhancement signals, packetize them, and then transmit multiple flows over the network towards the core-switch Aurrecoechea (1994). The network inevitably introduces some variation in delay of the delivered flows. The receiver depacketizes the flow and then attempts to faithfully play back the complete or partial layers to the decoder. This is accomplished in part by buffering the incoming flow to remove the network-induced jitter and then playing the signal back at some fixed delay offset from the original departure; the term playback point Jeffay (1992) Shenker (1993) refers to this point in time, which is offset from the original departure time by this fixed delay. The transport protocol utilises sync-filters object for delay-jitter management by calculating playout times of on-going flows based on the user supplied jitter parameter in the flow specification. Sync-filters can also operate on multiple related audio and video flows to provide low-level orchestration management (via the orch-filter).

For delay-jitter management sync-filters calculate the mean and variation in the end-to-end delay based on measured reception times by the "flow monitor". Sync-filters take the absolute, mean and variation in delay into account when calculating the playout estimate. A smoothing factor is used to dampen the movement of the playout adjustment. This is based on a linear recursive filtering mechanism which is characterised by a smoothing constant. From an intuitive standpoint the playout time needs to be set "far enough" beyond the delay estimate so that only a small fraction of the arriving packets are lost due to late packets. The QoS adaptor can trade off late packets and timeliness. This is based on the delay and loss parameters in the flow specification. The objective of delay-jitter management is to pull in the playout offset, while the objective of late-packet management is not to exceed the loss characterised in the service contract. The QOS adaptation manager moderates between timeliness and loss. Based on these QoS metrics the adaptation policy can adjust the damping factor, and acceptable ranges over which the playout point can operate.

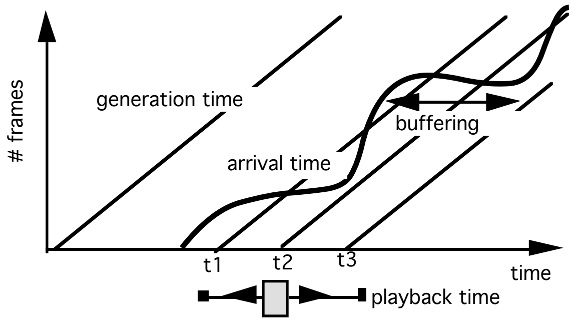

Figure 4 Timeliness versus packet loss

Figure 4 shows how packets arriving at the transport end-systems. Each packet includes a timestamp used in calculating the flow statistics for delay-jitter management. Selection of the playout point is important. An aggressive playout time which favour timeliness (such as t1) will results in a large number of late-packets. In contrast, a conservative playout point (such as t3) will be less responsive and timely but will result in no identifiable packet loss. In the DQM scheme a late packet is the same as a lost packets, and therefore the loss parameter in the flow specification moderates. An optimum playout schedule is represented by t2 in the diagram; here, continuous media delivery benifits from timely delivery with the exception of some packet loss - which may be deemed acceptable to the receiver.

6.3 Network-oriented DQM

Filter management Pasquale (1993) Yeadon (1994) includes filter placement algorithms which selects the optimum position on a core-based tree Ballardie (1993) to locate filters. Filter placement criteria is based on the current flow topology and the QoS specified by the receiver. As illustrated in Figure 1, QoS filters can be placed at any switch on the media path to meet the needs of receivers. DQM supports a QoS-based multicast scheme where senders and receivers can independently join QoS groups and connect to a core (which is addressed as a core-switch and core-id) Aurrecoechea (1994). Receivers and senders are said to be "decoupled" at cores. Currently DQM supports three selection filters at the network. These are low complexity and computationally simple filters for selecting sub-signals. Selection filters do not transform the structure of the internal stream i.e. have no knowledge of the format of the encoded flow above differentiating between BL, E1 and E2 sub-signals: (i) *sub-signal filters,* manipulate base and enhancement layers of multi-layer video which are multiplexed on a single flow. The definition of sub-signals is kept general here; since a flow may be comprised of an anchor and scalable extensions or the I and P pictures of MPEG2's simple profile, or the individual hybrid scalable profile. Sub-signal filters are installed in switches when a receiver joins on-going sessions; (ii) *hierarchical-filters*[], manipulate base and enhancement layers which are transmitted and received on independent flows in a non multiplexed fashion. In functional terms sub-signal and hierarchical filters can be considered to be equivalent in some cases. In sub-signal filtering one flow characterises the complete signal and in hierarchical-filtering a set of flows characterise the complete signal; and (iii) *hybrid-filters,* combine the benefits of sub-signal and hierarchical filtering techniques to meet the needs for complex sub-signal selection. For example hierarchical filter allows the BL, E1 and E2 to be carried over distinct flows, and the user can accordingly tune into each sub-signal. As an example, the base and enhancement layers of the hybrid scalable MPEG2 flow are each in turn made up of I,P and B pictures at each layer i.e., BL(I,P,B), E1(I,P,B) and E2(I,P,B). Using hybrid-filters the receiver can join the BL QoS group for the main profile and E1 QoS group for the spatial enhancement and then select sub-signals within each profile as needs be (e.g., the I and P pictures of the BL or E1 flows).

7 RELATED WORK

8 CONCLUSION

We have introduced a scheme for the dynamic management of multi-layer flows in heterogeneous multimedia networking environments. Dynamic QoS management (DQM) manipulates and adapts hierarchically coded flows at the end-systems and in the network using a set of scalable objects. The approach is based on three basic concepts: weighted fair share (WFS) service for adaptive flows, the scalable profiles of the MPEG2 standard that can provide discrete adaptation, and dynamic rate shaping algorithms for compressed digital video that provide continuous adaptation. At the present time DQM his being implemented at Lancaster University. The experimental infrastructure at Lancaster is based on 80486 machines running a multimedia enhanced Chorus mirco-kernel Coulson (1995) using programmable Olivetti Research Limited 4x4 ATM switches. At Columbia we currently using CORBA Lazar (1994) to propagate selection filters in the network using ASX200 switches. In addition to the implementation we are conducting an extensive simulation study into the feasibility of the adaptive network service for large scale use, and investigating the feasibility of extending the adaptive service concept into the micro-kernel itself Coulson (1995) for end-to-end QoS support. The results of this phase of the work will be the subject of a forthcoming paper.

9 REFERENCES

Shacham, N, (1992) “Multipoint Communication by Hierarchically Encoded Data”, *Proc. IEEE INFOCOM'92,* Florence, Italy, Vol.3, pp. 2107-2114.

Campbell, A., Coulson, G. and Hutchison, D., (1994) “A Quality of Service Architecture”, *ACM Computer Communications Review.*

Zhang, L., et. al., (1995) “RSVP Functional Specification”, Working Draft, draft-ietf-rsvp-spec-03.ps, 1995.

Pasquale, G., Polyzos, E., Anderson, E., and V. Kompella, (1993) "Fitter Propagation in Dissemination Trees: Trading Off Bandwidth and Processing in Continuos Media Networks", *Proc. Forth International Workshop on Network and Operating System Support for Digital Audio and Video*, Lancaster..

Delgrossi, L., Halstrinck, C., Henhmann, D.B, Herrtwich R.G, Krone, J., Sandvoss, C., and C. Vogt, (1993) "Media Scaling for Audio-visual Communication with the Heidelberg Transport System", *Proc ACM Multimedia'93* Anaheim, USA.

Hoffman, D., Speer, M. and G. Fernando, (1993) "Network Support for Dynamically Scaled Multimedia Data Streams", *Fourth International Workshop on Network and Operating System Support for Digital Audio and Video*, Lancaster.

Kanakia, H., Mishra, P., and A. Reibman, (1993) "An Adaptive Congestion Control Scheme for Real Time Packet Video Transport", *Proc. ACM SIGCOMM ‘93,* San Francisco, USA, October 1993.

Eleftheriadis, A., and D. Anastassiou, (1995) "Meeting Arbitrary QoS Constraints Using Dynamic Rate Shaping of Code Digital Video", *Fifth International Workshop on Network and Operating System Support for Digital Audio and Video*, Durham, New Hampshire.

Jeffay K., Stone, D.L., Talley T. and F.D. Smith, (1992) "Adaptive, Best Effort Delivery of Digital Audio and Video Across Packet-Switched Networks", *Proc. Third International Workshop on Network and Operating System Support for Digital Audio and Video*, San Diego.

Shenker, S., Clark, D., and L. Zhang, (1993) "A Scheduling Service Model and a Scheduling Architecture for an Integrated Service Packet Network", Working Draft available via anonymous ftp from parcftp.xerox.com: /transient/service-model.ps.Z.

Lazar, A. A., Pacifici, G. and D. Pendarakis, (1994) "Modeling Video Source for Real Time Scheduling", Multimedia Systems.

Kappner, T. and L. Wolf, (1994) “Media Scaling in Distributed Multimedia Object Service", *2nd International Workshop on Advanced Teleservices and High Speed Communication Architectures,* Heidelberg, Germany.

Steenstrup, M., (1992) “Fair Share for Resource Allocation”, pre-print.

Tokuda, H., Tobe, Y., Chou, S.T.C. and Moura, J.M.F., (1992) "Continuous Media Communication with Dynamic QOS Control Using ARTS with an FDDI Network", *Proc. ACM SIGCOMM ‘92,* Baltimore.

H.262, (1994) “Information Technology - Generic Coding of Moving Pictures and Associated Audio", Committee Draft, ISO/IEC 13818-2, International Standards Organisation, UK, March 1994.

Paek, S., Bocheck, P., and Chang S.-F., (1995) “Scalable MPEG-2 Video Servers with Heterogeneous QoS on Parallel Disk Arrays”, *Fifth International Workshop on Network and Operating System Support for Digital Audio and Video*, Durham, New Hampshire.

Lazar, A. A., Bhonsle S., Lim, K.S., (1994) "A Binding Architecture for Multimedia Networks", Proceedings of COST-237 Conference on Multimedia Transport and Teleservices, Vienna, Austria.

Aurrecoechea, C., Campbell, A., Hauw, L. and Hisaya Hadama, (1995), “A Model for Multicast for the Binding Architecture”, Technical Report, Center for Telecommunications Research, Columbia University.

Coulson, G., Campbell, A and P. Robin, (1995) “Design of a QoS Controlled ATM Based Communication System in Chorus”, *IEEE Journal of Selected Areas in Communications (JSAC),* Special Issue on ATM LANs: Implementation and Experiences with Emerging Technology, (to appear)

Ballardie, T., Francis, P. and Jon Crowcroft, (1993) "Core Based Tree (CBT) An Architecture for Scalable Inter-Domain Multicast Routing", Proc. ACM SIGCOMM ‘93, San Francisco.

Yeadon, N., Garcia, F., Campbell, A and D. Hutchison, (1994) “QoS Adaptation and Flow Filtering in ATM Networks”, 2nd International Workshop on Advanced Teleservices and High Speed Communication Architectures, Heidelberg, Germany.

OMG, (1993) "The Common Object Request Broker: Architecture & Specification, Rev 1.3., December 1993.