DISTRIBUTED SYSTEMS
Unit-5

Distributed Multimedia Systems: Introduction and Characteristics of Multimedia Data.
Quality of Service Management: Quality of Service negotiation, Admission Control.
Resource Management: Resource Scheduling.

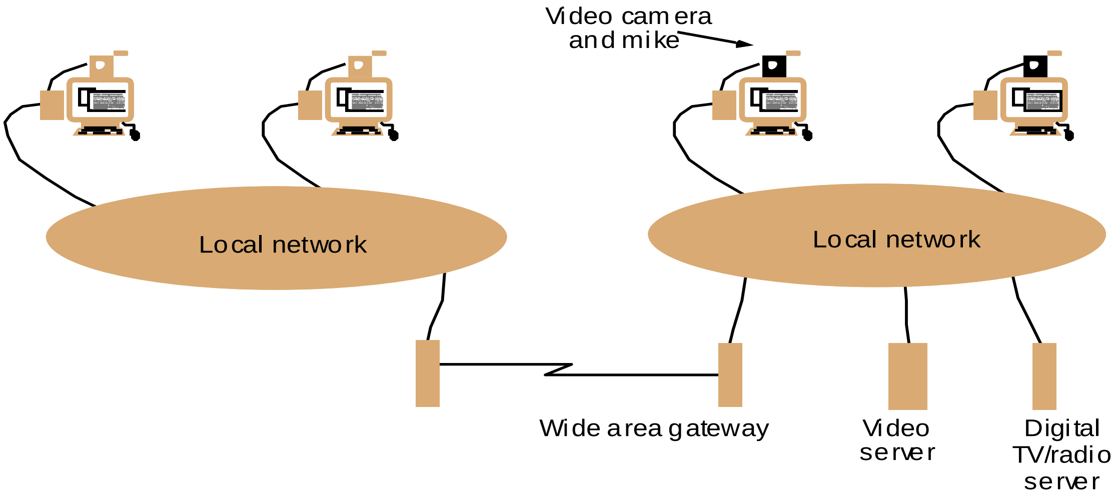
INTRODUCTION

Modern computers can handle streams of continuous, time-based data such as digital audio and video. This capability has led to the development of distributed multimedia applications such as networked video libraries, Internet telephony and videoconferencing. Such applications are viable with current general-purpose networks and systems although the quality of the resulting audio and video is often less than satisfactory. More demanding applications such as large-scale video conferencing, digital TV production, interactive TV and video surveillance systems are beyond the capabilities of current networking and distributed system technologies.

Multimedia applications **demand the timely delivery of streams** of multimedia data to end-users. Audio and video streams are generated and consumed in real time and the timely delivery of the individual elements (audio samples, video frames) is essential to the integrity of the application. In short, multimedia systems are real-time systems: they must perform tasks and deliver results according to a schedule that is externally determined. The degree to which this is achieved by the underlying system is known as the **QUALITY OF SERVICE (QOS)** enjoyed by an application.

The planned allocation and scheduling of resources to meet the needs of multimedia and other applications is referred to as **QUALITY OF SERVICE MANAGEMENT**.

Multimedia applications require the continuous processing and transmission of bulky streams of data at high bandwidths and with frequent deadlines (e.g. there is a deadline for the delivery of each video frame to its destination) but the consequences of failure are less serious – a small proportion of missed deadlines can often be tolerated.



The consequences of failure to meet deadlines in multimedia applications can be serious, especially in commercial environments such as video-on-demand services, business conferencing applications and remote medicine, but the requirements differ significantly from those of other real-time applications:

Multimedia applications are often highly distributed and operate within general purpose distributed computing environments. They therefore compete with other distributed applications for network bandwidth and for computing resources at users' workstations and servers.

The resource requirements of multimedia applications are dynamic. A video conference will require more or less network bandwidth as the number of participants grows or shrinks. Its use of computing resources at each user's workstation will also vary, since, for example, the number of video streams that have to be displayed varies. Multimedia applications may involve other variable or intermittent loads. For example, a multimedia lecture might include a processor-intensive simulation activity.

QoS management systems are intended to meet all of these needs, managing the available resources dynamically and varying the allocations in response to changing demands and user priorities. A QoS management system must manage all of the computing and communication resources needed to acquire process and transmit multimedia data streams, especially where the resources are shared between applications.

A typical distributed multimedia system capable of supporting a variety of applications such as desktop conferencing and providing access to stored video sequences, broadcast digital TV and radio. The resources for which QoS management is required include network bandwidth, processor cycles and memory capacity. Disk bandwidth at the video server may also be included. We shall adopt the generic term resource bandwidth to refer to the capacity of any hardware resource (network, central processor, disk subsystem) to transmit or process multimedia data.

MULTIMEDIA APPLICATIONS DEPLOYED TODAY WITH QOS-LESS COMPUTING INCLUDE:

1. Web-based multimedia: These are applications that provide access to : streams of audio and video data published via the Web. They have been successful when there is little or no need for the synchronization of the data streams at different locations. Their performance is constrained by the limited bandwidth and variable latencies found in current networks and by the inability of current operating systems to support real-time resource scheduling. For audio and low-quality video sequences, the use of extensive buffering at the destination to smooth out the variations in bandwidth and latency results in continuous and smooth display of video sequences but with a source-to-destination delay that may reach several seconds.
2. Network phone and audio conferencing: This type of application has relatively low: bandwidth requirements, especially when efficient compression techniques are used. But its interactive nature demands low round-trip delays and these cannot always be achieved.
3. Video on demand services: These supply video information in digital form:, retrieving the data from large online storage systems and delivering them to the end-user's display. These are successful where sufficient dedicated network bandwidth is available and where the video server and the receiving stations are dedicated. They also employ considerable buffering at the destination.

MULTIMEDIA APPLICATIONS DEPLOYED TODAY WITH QOS-HIGH COMPUTING INCLUDES:

Many multimedia applications are cooperative (involving several users) and synchronous (requiring the users' activities to be closely coordinated). They span a wide spectrum of application contexts and scenarios. For example:

- A simple video conference involving two or more users, each using a workstation equipped with a digital video camera, microphone, sound output and video display capability. Application software to support simple teleconferencing is widely available, but its performance is severely limited by today's computing and network environments.
- A music rehearsal and performance facility enabling musicians at different locations to perform in an ensemble. This is a particularly demanding multimedia application because the synchronization constraints are so tight.

MULTIMEDIA APPLICATIONS REQUIRE:

1. **Low latency communication:** Round trip delays < 100 ms, so that interaction: between users appears to be synchronous.
2. **SYNCHRONOUS DISTRIBUTED STATE:** If one user stops a video on a given frame, the other: users should see it stopped at the same frame.
3. **MEDIA SYNCHRONIZATION:** All participants in a music performance should hear the: performance at approximately the same time. **Separate soundtrack and video streams should maintain 'lip sync',** e.g. for a user commenting live on a video playback, or a distributed Karaoke session.
4. **EXTERNAL SYNCHRONIZATION:** In conferencing and other cooperative applications, there: may be active data in other formats, such as computer-generated animations, CAD data, electronic whiteboards, and shared documents. Updates to these must be distributed and acted upon in manner that appears at least approximately synchronized with the time-based multimedia streams.

CHARACTERISTICS OF TYPICAL MULTIMEDIA STREAMS

It is likely that multimedia applications will remain in the window of scarcity (Say resources sufficient or not) for the foreseeable future. Advances in system performance are likely to be used to improve the quality of multimedia data, to include higher frame rates and greater resolution for video streams or to support many media streams concurrently, for example in **a video conferencing** system. More demanding applications, including virtual reality and real-time stream manipulation ("special effects") can extend the window of scarcity almost indefinitely.

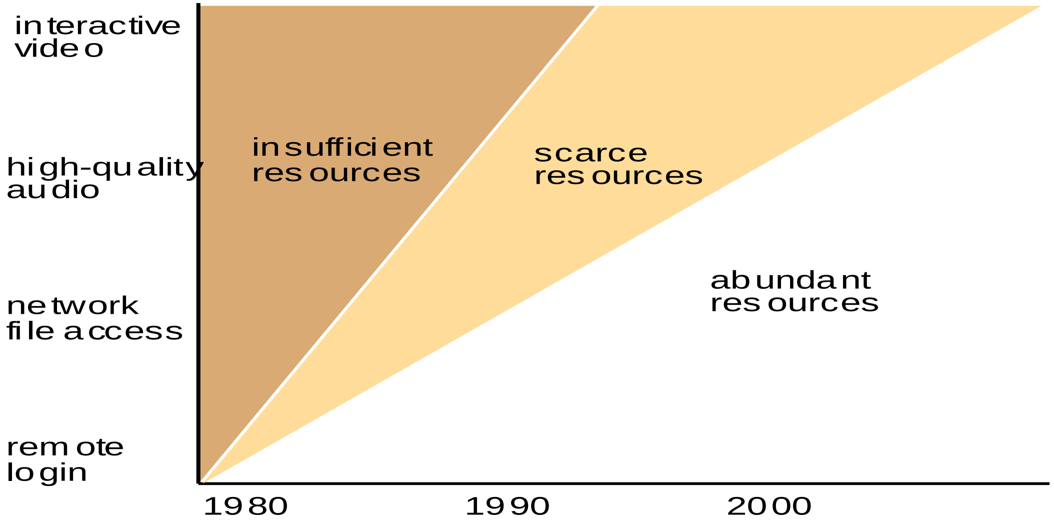
The term 'continuous' refers to the **user's view of the data**. Internally, continuous media are represented as sequences of discrete values which replace each other over time. For example, the value of an image array is replaced 25 times per second to give the impression of a TV-quality view of a moving scene; a sound amplitude value is replaced 8000 times per second to convey telephone quality speech.

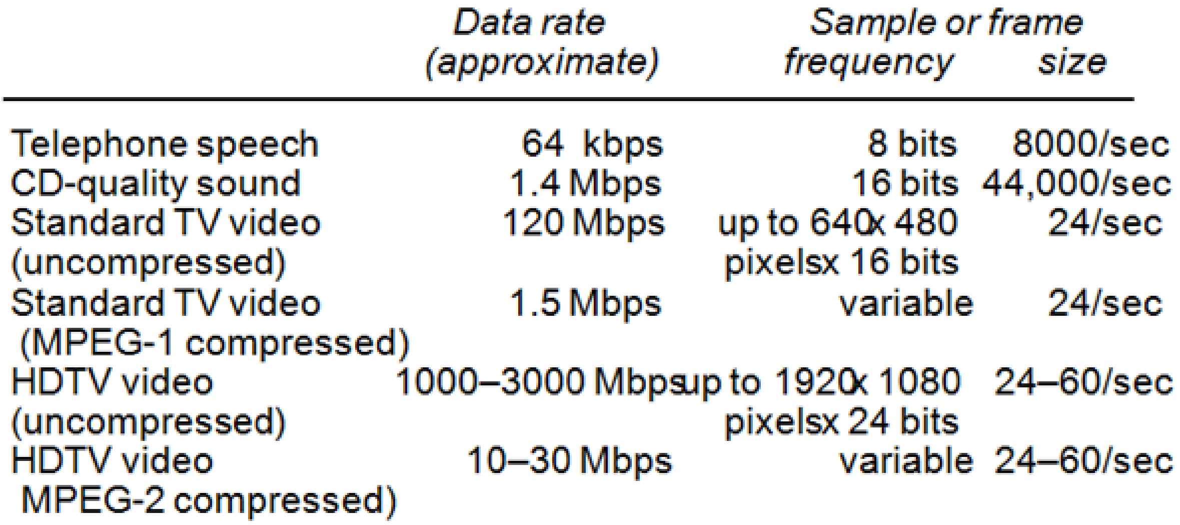
Multimedia streams are said to be time-based (or isochronous) because timed data elements in audio and video streams define the semantics or 'content' of the stream. The times at which the values are played or recorded affect the validity of the data. Hence systems that support multimedia applications need to preserve the timing when they handle continuous data.

Compression can reduce bandwidth requirements by factors between 10 and 100, but the timing requirements of continuous data are unaffected. Various compressed data formats such as GIF, TIFF and JPEG for still images and MPEG-1, MPEG-2 and MPEG-4 for video sequences.

For Compression we use of special-purpose hardware to process and despatch video and audio information – the video and audio coders/decoders (Codecs) found on video cards manufactured for personal computers. The compression method used for the MPEG video formats is asymmetric, with a complex compression algorithm and simpler decompression.

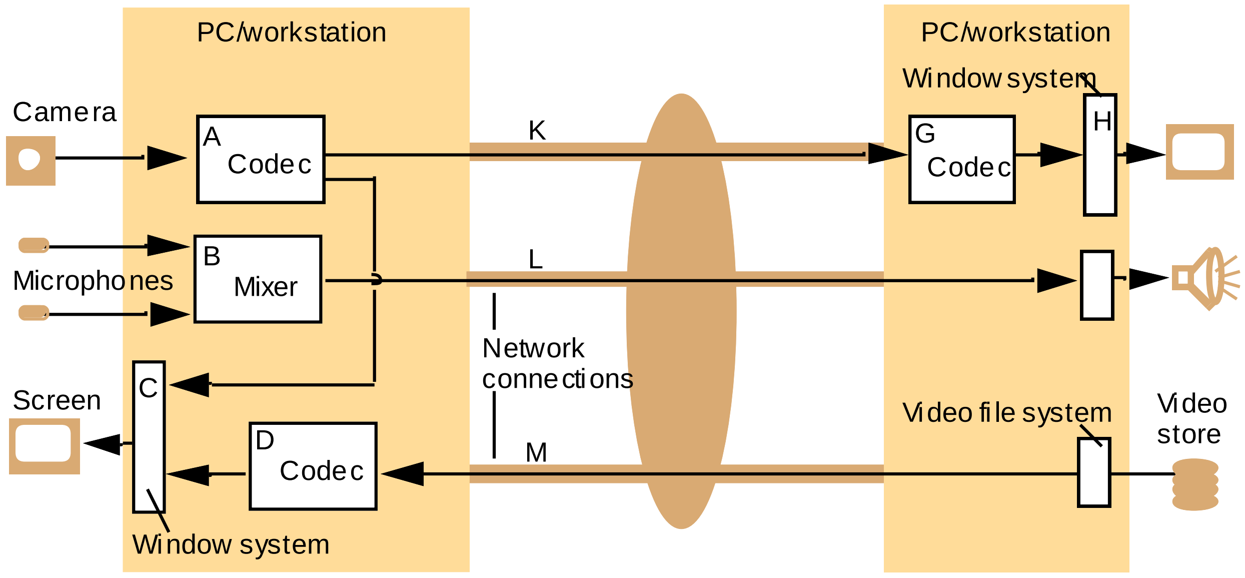
The window of scarcity for computing and communication resources





	Data rate (approximate)	Sample or frame frequency	size
Telephone speech	64 kbps	8 bits	8000/sec
CD-quality sound	1.4 Mbps	16 bits	44,000/sec
Standard TV video (uncompressed)	120 Mbps	up to 640x 480 pixelsx 16 bits	24/sec
Standard TV video (MPEG-1 compressed)	1.5 Mbps	variable	24/sec
HDTV video (uncompressed)	1000–3000 Mbps	up to 1920x 1080 pixelsx 24 bits	24–60/sec
HDTV video MPEG-2 compressed)	10–30 Mbps	variable	24–60/sec

TYPICAL INFRASTRUCTURE COMPONENTS FOR MULTIMEDIA APPLICATIONS



→ : multimediastream
White boxes represent media processing components,
many of which are implemented in software, includingodec: coding/decodingfilter
mixer: sound-mixingcomponent

QUALITY OF SERVICE MANAGEMENT

The management and allocation of resources to provide such guarantees is referred to as **quality of service management.**

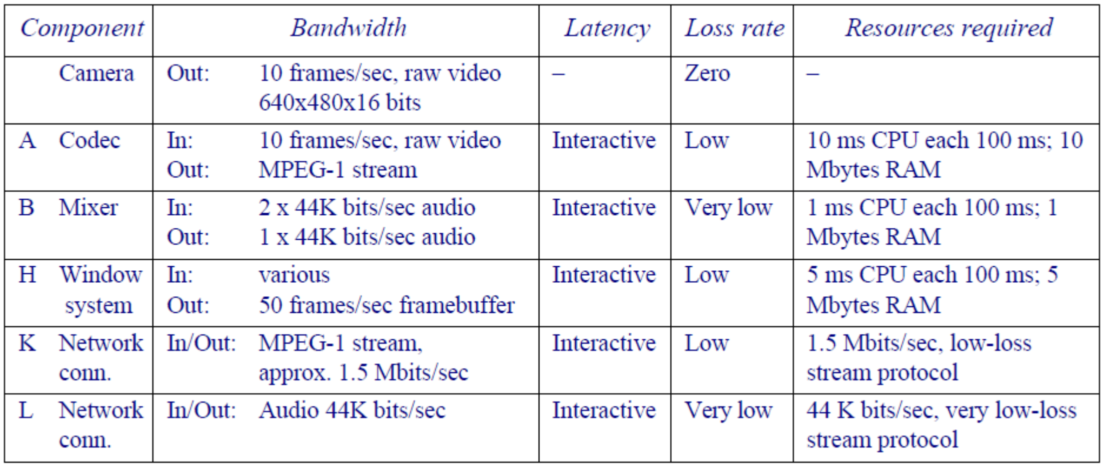
QOS MANAGER is a system component of **QUALITY OF SERVICE MANAGEMENT** responsible for the allocation and scheduling of those resources.

When multimedia applications run in networks of personal computers they compete for resources at the workstations running the applications (processor cycles, bus cycles, buffer capacity) and in the networks (physical transmission links, switches, gateways). Workstations and networks may have to support several multimedia and conventional applications. There is competition between the multimedia and conventional applications, between different multimedia applications and even between the media streams within individual applications.

The concurrent use of physical resources for a variety of tasks has long been possible with multi-tasking operating systems and shared networks. In multi-tasking operating systems the central processor is allocated to individual tasks (or processes) in a round-robin or other scheduling scheme that shares the processing resources on a best-efforts basis amongst all of the tasks currently competing for the central processor.

Collisions can occur and when they do sending nodes wait for random back off periods in order to prevent repeated collisions.

QoS specs for components of the application shown



Component	*Bandwidth*		*Latency*	*Loss rate*	*Resources required*
Camera	Out:	10 frames/sec, raw video 640x480x16 bits	–	Zero	–
A Codec	In: Out:	10 frames/sec, raw video MPEG-1 stream	Interactive	Low	10 ms CPU each 100 ms; 10 Mbytes RAM
B Mixer	In: Out:	2 x 44K bits/sec audio 1 x 44K bits/sec audio	Interactive	Very low	1 ms CPU each 100 ms; 1 Mbytes RAM
H Window system	In: Out:	various 50 frames/sec framebuffer	Interactive	Low	5 ms CPU each 100 ms; 5 Mbytes RAM
K Network conn.	In/Out:	MPEG-1 stream, approx. 1.5 Mbits/sec	Interactive	Low	1.5 Mbits/sec, low-loss stream protocol
L Network conn.	In/Out:	Audio 44K bits/sec	Interactive	Very low	44 K bits/sec, very low-loss stream protocol

QOS MANAGER'S TWO MAIN SUB-TASKS:

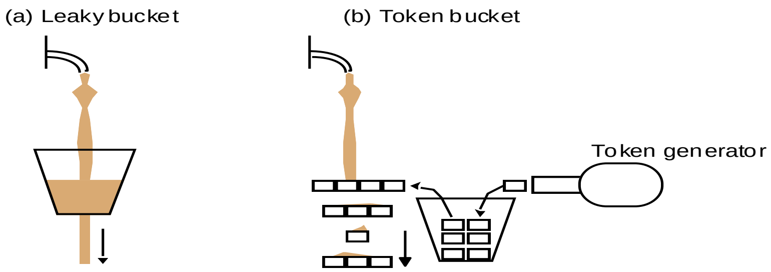
1. Quality of service negotiation. 2. Admission control.

QUALITY OF SERVICE NEGOTIATION

The application indicates its resource requirements to the QoS manager. The QoS manager evaluates the feasibility of meeting the requirements against a database of the available resources and current resource commitments and gives a positive or negative response. If it is negative, the application may be reconfigured to use reduced resources and the process is repeated.
To negotiate QoS between an application and its underlying system, an application must specify its QoS requirements to the QoS Manager. This is done by the transmission of a set of parameters. Three parameters are of primary interest when it comes to processing and transporting multimedia streams: bandwidth, latency, and loss rate.

BANDWIDTH: It of a multimedia stream or component is the rate at which data flows through it. Bandwidth is the characterization of burstiness.
LATENCY: It is the time required for an individual data element to move through a stream from the source to the destination. Of course this may vary depending on the volume of other data in the system and other characteristics of the system load. This variation is termed jitter – formally, jitter is the first derivative of the latency.
LOSS RATE: Since the late delivery of multimedia data is of no value, data elements will be dropped when it is impossible to deliver them before their scheduled delivery time. To avoid the data loss we either use leaky bucket or token bucket algorithm.

TRAFFIC SHAPING ALGORITHMS



NOTE:
The model of linear-bounded arrival processes (LBAP) used in defines the maximum number of messages in a stream during any time interval t as Rt + B where R is the rate and B is the maximum size of burst. The burst parameter defines the amount of buffer space required to avoid loss.

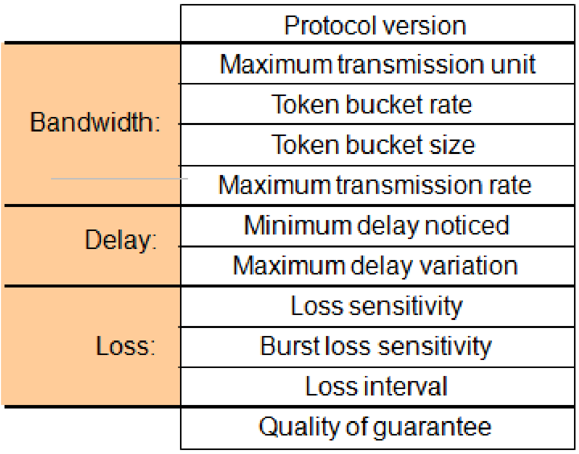
ADMISSION CONTROL

If the result of the resource evaluation is positive, the requested resources are reserved and the application is given a Resource Contract, stating the resources that have been reserved. The contract includes a time limit. The application is then free to run. If it changes its resource requirements it must notify the QoS Manager. If the requirements decrease, the resources released are returned to the database as available resources. If they increase, a new round of negotiation and admission control is initiated.

TRAFFIC SHAPING

Traffic shaping is the term used to describe the use of output buffering to smooth the flow of data elements. The bandwidth parameter of a multimedia stream typically provides an idealistic approximation of the actual traffic pattern that will occur when the stream is transmitted. The closer the actual traffic pattern matches the description, the better a system will be able to handle the traffic, in particular when it uses scheduling methods that are designed for periodic requests. The LBAP model of bandwidth variations calls for regulation of the burstiness of multimedia streams. Any stream can be regulated by inserting a buffer at the source and by defining a method by which data elements leave the buffer.

NOTE:
For traffic shaping we user flow specification format (RFC 1363 Flow Spec.).



	Protocol version
Bandwidth:	Maximum transmission unit
	Token bucket rate
	Token bucket size
	Maximum transmission rate
Delay:	Minimum delay noticed
	Maximum delay variation
Loss:	Loss sensitivity
	Burst loss sensitivity
	Loss interval
	Quality of guarantee

FLOW SPECIFICATIONS

A collection of QoS parameters is typically known as a flow specification, or flow spec for short. Several examples of flow specs exist and are all similar. In Internet RFC 1363, a flow spec is defined as eleven 16-bit numeric values (Figure Above) that reflect the QoS parameters discussed above in the following way:

- The maximum transmission unit and maximum transmission rate determine the maximum bandwidth required by the stream.
- The token bucket size and rate determine the burstiness of the stream.
- The delay characteristics are specified by the minimum delay that an application can notice (since we wish to avoid over-optimization for short delays) and maximum jitter it can accept.
- The loss characteristics are defined by the total acceptable number of losses over a certain interval and the maximum number of consecutive losses.

NEGOTIATION PROCEDURES

For distributed multimedia applications, the components of a stream are likely to be located in several nodes. There will be a QoS manager at each node. A straightforward approach to QoS negotiation is to follow the flow of data along each stream from the source to the target. A source component initiates the negotiation by sending out a flow spec to its local QoS manager.

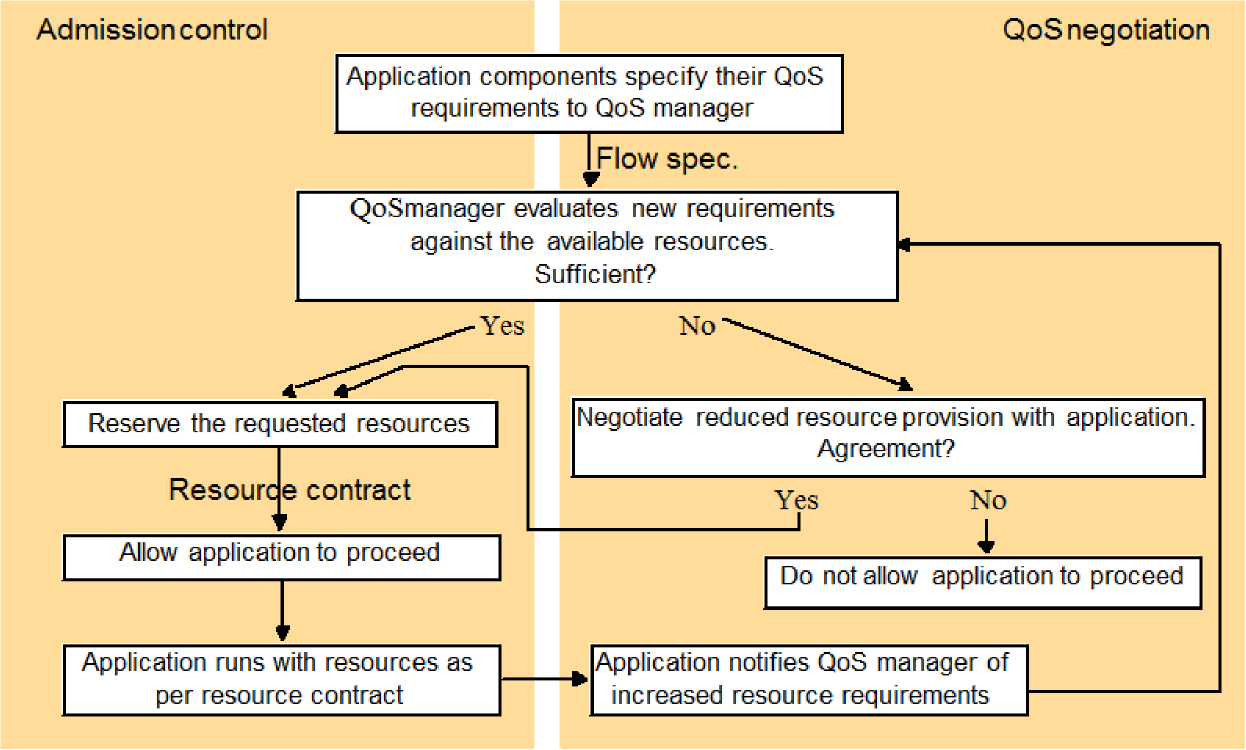
ADMISSION CONTROL regulates access to resources to avoid resource overload and to protect resources from requests that they cannot fulfil. It involves turning down service requests should the resource requirements of a new multimedia stream would violate existing QoS guarantees.

An admission control scheme is based on some knowledge of both the overall system capacity and the load generated by each application. The bandwidth requirement specification for an application may reflect the maximum amount of bandwidth that an application will ever require, the minimum bandwidth it will need to function, or some average value in between. Correspondingly, an admission control scheme may base its resource allocation on any of these values.

For resources that have a single allocator, admission control is straightforward. Resources that have distributed access points, such as many local area networks, require either a centralized admission control entity or some distributed admission control algorithm that avoids conflicting concurrent admissions. Bus arbitration within workstations falls into this category – however, even multimedia systems that perform bandwidth allocation extensively do not control bus admission as bus bandwidth is not considered to be in the window of scarcity.

STATISTICAL MULTIPLEXING Because of the potential under-utilization that can occur, it is common to overbook resources. The resulting guarantees, often called statistical or soft guarantees to distinguish them from the deterministic or hard guarantees introduced before, are only valid with some (usually very high) probability. Statistical guarantees tend to provide better resource utilization as they do not consider the worst case.

Statistical multiplexing is based on the hypothesis that for a large number of streams the aggregate bandwidth required remains nearly constant regardless of the bandwidth of individual streams.


Admission control
QoS negotiation
Application components specify their QoS requirements to QoS manager
Flow spec.
QoSmanager evaluates new requirements against the available resources. Sufficient?
Yes
No
Reserve the requested resources
Negotiate reduced resource provision with application. Agreement?
Resource contract
Yes
No
Allow application to proceed
Do not allow application to proceed
Application runs with resources as per resource contract
Application notifies QoS manager of increased resource requirements

RESOURCE MANAGEMENT

To provide a certain QoS level to an application, not only does a system need to have sufficient resources (performance), it also needs to make these resources available to an application when they are needed (scheduling).

RESOURCE SCHEDULING

Processes need to have resources assigned to them according to their priority. A resource scheduler determines the priority of processes based on certain criteria. Traditional CPU schedulers in time-sharing systems often base their priority assignments on responsiveness and fairness: I/O intensive tasks get high priority to guarantee fast response to user requests, CPU-bound tasks get lower priorities and overall, processes in the same class are treated equally.

RESOURCE SCHEDULING TYPES

1. Fair scheduling.
2. Real-time scheduling.

FAIR SCHEDULING If several streams compete for the same resource it becomes necessary To consider fairness and to prevent ill-behaved streams from taking too much bandwidth. A straightforward approach to ensure fairness is to apply round-robin scheduling to all streams in the same class. Whereas in [Nagle 1987] such a method was introduced on a packet-by-packet basis, in [Demers et al. 1989] the method is used on a bit-by-bit basis which provides more fairness with respect to varying packet sizes and packet arrival times. These methods are known as fair queuing. It is done by bit by bit or packet by packet.

All basic round-robin schemes assign the same bandwidth to each stream. To take the individual bandwidth of streams into account, the bit-by-bit scheme can be extended so that for certain streams a larger number of bits can be transmitted per cycle. This method is called weighted fair queuing.

REAL-TIME SCHEDULING Traditional real-time scheduling methods suit the model of regular continuous multimedia streams very well. Earliest-deadline-first (EDF) scheduling has almost become a synonym for these methods. An EDF scheduler uses a deadline that is associated with each of its work items to determine the next item to be processed: the item with the earliest deadline goes first. In multimedia applications, we identify each media element arriving at a process as a work item. EDF scheduling is proven to be optimal for allocating a single resource based on timing criteria: if there is a schedule that fulfils all timing requirements, EDF scheduling will find it.

RATE-MONOTONIC (RM) SCHEDULING is a prominent technique for real-time scheduling of periodic processes that achieves just this. RM scheduling has been shown to be optimal for situations that only utilize a given bandwidth by less than 69% . Using such an allocation scheme, the remaining bandwidth could be given to non-real-time applications.

STREAM ADAPTATION

Whenever a certain QoS cannot be guaranteed or can only be guaranteed with a certain probability, an application needs to adapt to changing QoS levels, adjusting its performance accordingly. For continuous-media streams, the adjustment translates into different levels of media presentation quality.

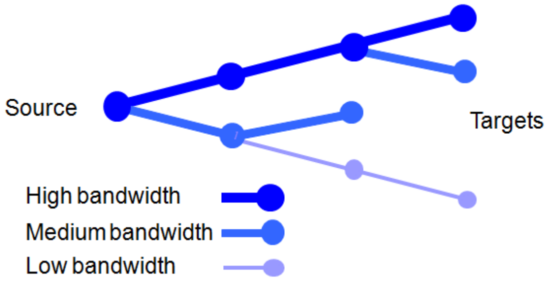
The simplest form of adjustment is to drop pieces of information. This is easily done in audio streams where samples are independent from each other, but it can immediately be noticed by the listener. Drop-outs in a video stream encoded in Motion JPEG, where each frame stands for itself are more tolerable. Encoding mechanisms such as MPEG, where the interpretation of a frame depends on the values of several adjacent frames, are less robust against omissions: it takes a longer time to recover from errors and the encoding mechanism may, in fact, amplify errors.

For non-interactive applications this may be acceptable, although it can eventually lead to buffer overflows as data is accumulated between the source and sink. For conferencing and other interactive applications, increasing delays are not acceptable, or must exist only for a short period.

SCALING (RESIZING)

If adaptation is performed at the target of a stream, the load on any bottleneck in the system is not decreased and the overload situation persists. It is useful to adapt a stream to the bandwidth available in the system before it enters a bottleneck resource in order to resolve contention. This is known as scaling. Scaling algorithms are media-dependent.

FILTERING



It can also be achieved by dropping a channel in a stereo transmission.
Scaling methods can work at different granularities for video

1. TEMPORAL SCALING reduces the resolution of the video stream in the time domain by decreasing the number of video frames transmitted within an interval. Temporal scaling is best suited for video streams in which individual frames are selfcontained and can be accessed independently. Delta compression techniques are more difficult to handle as not all frames can be easily dropped. Hence, temporal scaling is more suitable for Motion JPEG than for MPEG streams.
2. SPATIAL SCALING reduces the number of pixels of each image in a video stream. For spatial scaling, hierarchical arrangement is ideal because the compressed video is immediately available in various resolutions. Therefore, the video can be transferred over the network using different resolutions without recoding each picture before finally transmitting it. JPEG and MPEG-2 support different spatial resolutions of images and are well-suited for this kind of scaling.
3. FREQUENCY SCALING modifies the compression algorithm applied to an image. This result in some loss of quality, but in a typical picture, compression can be increased significantly before a reduction of image quality becomes visible.
4. AMPLITUDINAL SCALING reduces the colour depths for each image pixel. This scaling method is, in fact, used in H.261 encodings to arrive at a constant throughput although image content varies.

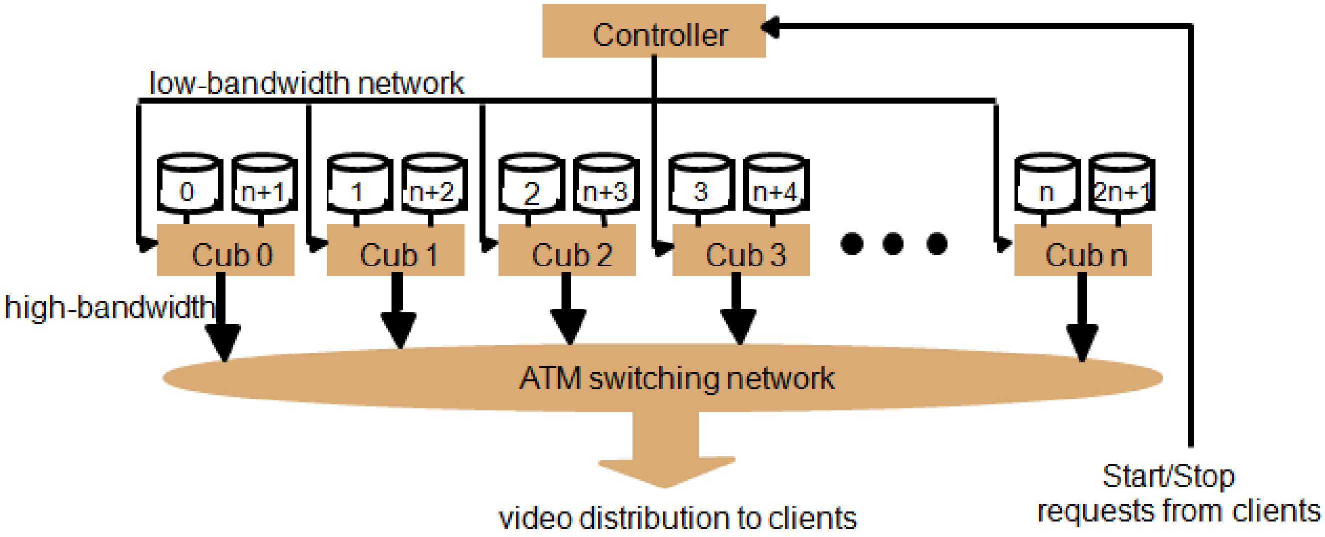
5. COLOUR SPACE SCALING reduces the number of entries in the colour space. One way to realize colour space scaling is to switch from colour to greyscale presentation. Obviously, combinations of these scaling methods are possible. A system to perform scaling consists of a monitor process at the target side and a scaler process at the source. The monitor keeps track of the arrival times of messages. When messages get delayed, it is an indication of some bottleneck in the system. The monitor then sends a Scale-Down message to the source and it reduces the bandwidth of the stream. After some period of time, the source scales the stream up again. Should the bottleneck still exist, the monitor will again detect a delay and scale the stream down 1993]. The fundamental problem of the scaling approach is to find good heuristics to avoid unnecessary Scale-Up operations and to prevent the system from oscillating.

FILTERING

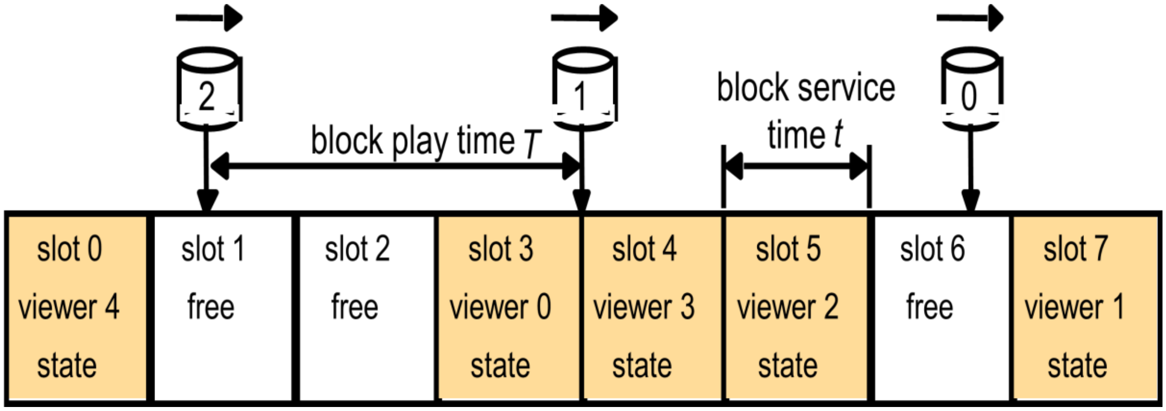
As scaling modifies a stream at the source, it is not always suitable for applications that involve several receivers: when a bottleneck occurs on the route to one target, this target sends a Scale-Down message to the source and all targets receive the degraded quality although some would have no problem in handling the original stream.

Filtering is a method that provides the best possible QoS to each target by applying scaling at each relevant node on the path from the source to the target (Figure 15.9). RSVP is an example of a QoS negotiation protocol that supports filtering. Filtering requires that a stream can be partitioned into a set of hierarchical substreams, each adding a higher level of quality. The capacity of nodes on a path determines the number of sub-streams a target receives. All other sub-streams are filtered out as close to the source as possible (perhaps even at the source) to avoid transfer of data that is later thrown away. A sub-stream is not filtered at an intermediate node if somewhere downstream a path exists that can carry the entire sub-stream.

TIGER VIDEO FILE SERVER HARDWARE CONFIGURATION



TIGER SCHEDULE


2
1
0
block play time T
block service time t
slot 0 viewer 4 state
slot 1 free
slot 2 free
slot 3 viewer 0 state
slot 4 viewer 3 state
slot 5 viewer 2 state
slot 6 free
slot 7 viewer 1 state