

Sistemas Multimídia Distribuidos

Baseado no cap. 17 do livro do Coulouris[3]

6 de setembro de 2011

Panorama I

- Fluxos contínuos de dados em tempo real
- Grandes quantidades de áudio, vídeo e outros elementos, respeitando o critério temporal
- Elementos de dados distribuídos com atrasos geralmente são eliminados
- Especificação: em termos de taxa de passagem de dados (largura de banda), atraso da distribuição de cada elemento (latência) e taxa de eliminação/perca de pacotes
- Latência: especialmente importante em aplicativos interativos
- Perca de pacotes é aceitável quando é possível re-sincronizar após o ponto de perda

Panorama II

- Alocação de recursos é referida como qualidade de serviços: alocação de processamento, largura de banda da rede e memória (para buffer)

Introdução I

- Fluxos de dados contínuos (streams) baseados no tempo: telefonia pela Internet, vídeoconferência, etc
- A qualidade geral é ruim; é imprópria para: TV digital/interativa, supervisão com vídeo
- Sistemas multimídia são sistemas em tempo real: precisam executar tarefas e apresentar resultados de acordo com um escalonamento determinado externamente
- O grau de sucesso desse fornecimento é o QoS (Quality of Service), usufruída pelo aplicativo
- Diferenças entre os sistemas de tempo real de aviação, processo de fabricação, etc:

Introdução II

- estes possuem volumes de dados pequenos e prazos finais rígidos; o não cumprimento pode ter consequências desastrosas, por isso superestima-se recursos e trabalha-se com atendimento no pior caso
- os sistemas multimídia:
- operam dentro de um ambiente geral, competindo com recursos e banda de rede com outros aplicativos distribuídos
- os requisitos são dinâmicos: mais participantes, mais recursos necessários; ou uma simulação pode requerer mais processamento

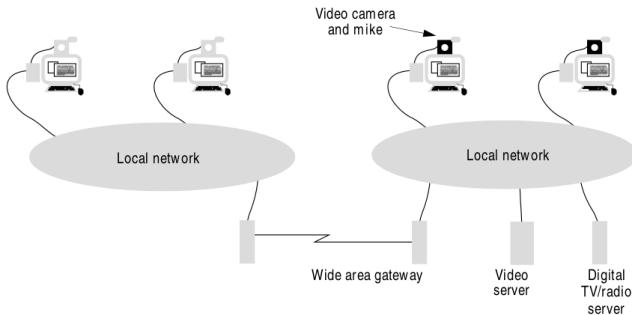
Introdução III

- operação de sistemas multimídia em conjunto com outras aplicações: edição de textos, conversa de voz separada, mensagens instantâneas, em meio a uma vídeo-conferência
- vídeoconferência em desktop
- acesso a sequência de vídeo
- transmissão de TV e rádio digital

Recursos para o gerenciamento da qualidade de serviço: largura de banda da rede, ciclos do processador e capacidade de memória

Introdução

Figura: Sistema Multimídia Distribuído



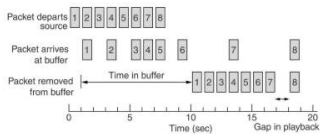
Introdução

Aplicativos multimídia que têm sido implantados:

- Multimídia baseada na web: permite acesso aos fluxos de áudio e vídeo na Web; buffers podem fornecer exibição contínua e suave mas com atraso da origem para o destino (segundos)
- Telefone de rede e áudio-conferência: aplicações de natureza interativa com baixos atrasos de RTT¹
- Vídeo sob demanda: bem sucedidos onde há largura de banda, servidor de vídeo e estações, todos dedicados; alto uso de buffers no destino

¹round-trip time, tempo de ida e volta

Figura: Buffers



Introdução

Aplicativos muito interativos: problemas...

- telefonia na Internet - VOIP
- vídeoconferência: restrições de largura de banda e latência
- ensaio de execução musical distribuída: severas restrições de sincronização

Introdução

Exigências das aplicações super-interativas

- comunicação com baixa latência:
tela+controle+vídeo+rede+captação ; RTT de 100 a 300 ms
- estado distribuído síncrono: se um usuário interrompe um vídeo, todos devem ver a interrupção no mesmo quadro
- sincronismo de mídia: o exemplo da execução musical distribuída; Konstantas et al. [1997] aponta até 50ms; fluxos separados de áudio e vídeo devem manter sincronismo *labial*²
- sincronização externa: aplicações cooperativas diversas devem parecer sincronizadas³ com os fluxos multimídia baseados no tempo (exemplo: animações de computador, dados CAD, quadros-negros eletrônicos).

²exemplo: sessão de karaokê distribuída

³isso é perceptível quando filmamos a televisão

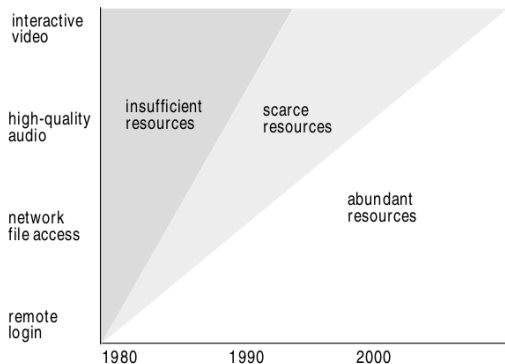
Introdução

Janela de escassez

- Sistemas atuais tem capacidade para manipular dados multimídia
- As limitações estão nos recursos necessários, especialmente na quantidade e qualidade de fornecimento de fluxos
- É necessário alocar e escalonar os recursos
- *Antes que a janela de escassez seja alcançada, um sistema tem recursos insuficientes para executar as aplicações relevantes*

Introdução

Figura: Janela de Escassez



Características

Algumas definições

- média contínua: sequência de valores discretos que substituem-se uns aos outros com o passar do tempo; ex: uma imagem é amostrada 25 vezes/seg. para dar impressão de movimento com qualidade de TV; um sinal sonoro é amostrado 8000 vezes/seg para transmitir fala com a qualidade de um telefone
- fluxos multimídia são baseados no tempo⁴: os tempos nos quais os valores são reproduzidos ou gravados afetam a validade dos dados, definem a *semântica* ou conteúdo do fluxo

⁴ ou isocrônicos

Características

Figura: Taxas e Amostras

	<i>Data rate (approximate)</i>	<i>Sample or frame size</i>	<i>frequency</i>
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 640×480 pixels \times 16 bits	24/sec
Standard TV video (MPEG-1 compressed)	1.5 Mbps	variable	24/sec
HDTV video (uncompressed)	1000–3000 Mbps	up to 1920×1080 pixels \times 24 bits	24–60/sec
HDTV video (MPEG-2 compressed)	10–30 Mbps	variable	24–60/sec

Características

- dados multimídia são volumosos: precisam de maior desempenho de entrada/saída que os sistemas convencionais
- utiliza-se compactação, embora transformações, como mistura de vídeo, sejam difíceis de realizar com fluxos compactados.
- compactação pode reduzir requisitos de largura de banda, mas não requisitos de temporização de dados contínuos
- codecs em hardware podem realizar a compactação; codecs em software oferecem maior flexibilidade

Figura: Pacotes multimídia

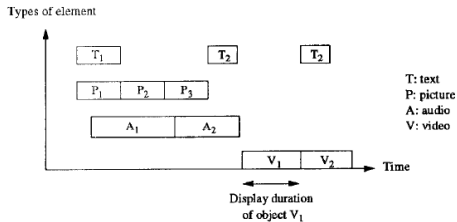


Fig. 33. The temporal relationships between objects in a multimedia scenario.

Características

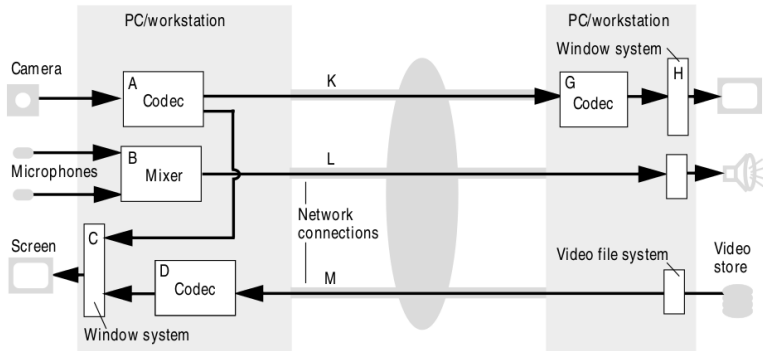
- método MPEG é assimétrico⁵, com algoritmo de compactação complexo e descompactação simples
- isso ajuda em conferências no desktop: compactação feita por codec em hardware, descompactação via software

⁵o codificador é diferente do decodificador

Gerenciamento da Qualidade de Serviço

- aplicações multimídia executadas em redes de PC's competem por recursos: ciclos de processador, barramento, capacidade de buffer)
- redes são projetadas pra que mensagens de diferentes origens sejam intercaladas, permitindo a existência de muitos canais de comunicação virtuais nos mesmos canais físicos
- Ethernet: gerencia um meio de transmissão compartilhado na base do melhor esforço (não-confiável)
- enfim: rodízio, tempos aleatórios, etc, podem não satisfazer as necessidades das aplicações multimídia: distribuição atrasada não tem valor.

Figura: arquitetura abstrata com fluxos de mídia de dados gerados continuamente



→ : multimedia stream

White boxes represent media processing components, many of which are implemented in software, including:

codec: coding/decoding filter
 mixer: sound-mixing component

QoS: Negociação

A aplicação especifica requisitos de qualidade através de 3 parâmetros:

- largura de banda: a taxa na qual os dados fluem pelo fluxo
- latência: tempo exigido para um elemento de dados individual se mover em um fluxo, da origem até o destino; a variação dessa latência é denominada *jitter*
- taxa de perda: quadros de vídeo ou amostras de áudio eliminados; até 1%; para aplicações críticas, bem menos!

Figura: requisitos de recurso para a arquitetura abstrata

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

Figura: Tarefas do negociador de QoS

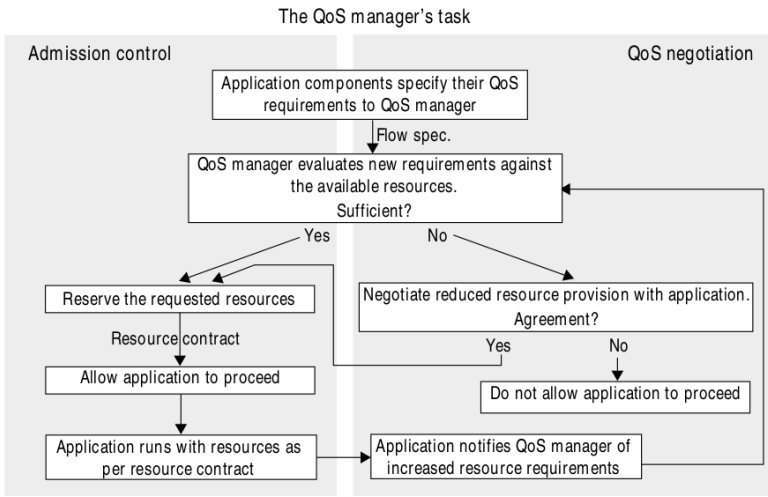
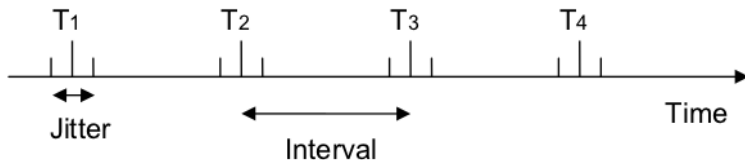


Figura: Jitter[6]



T_i : arrival time of video frame i

Figura:

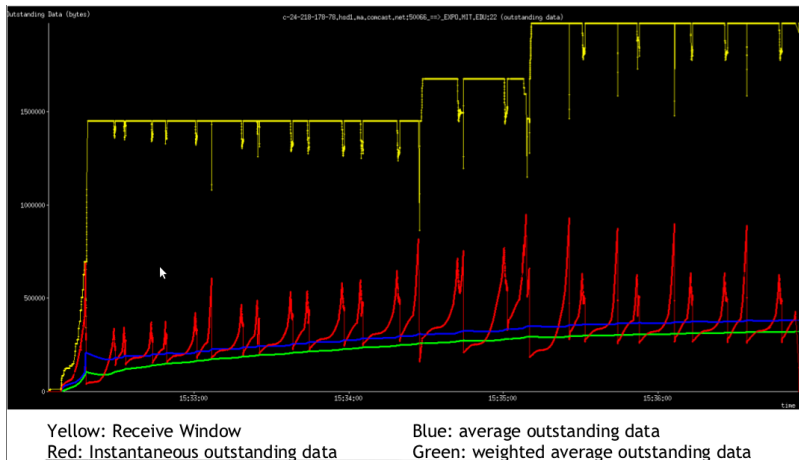
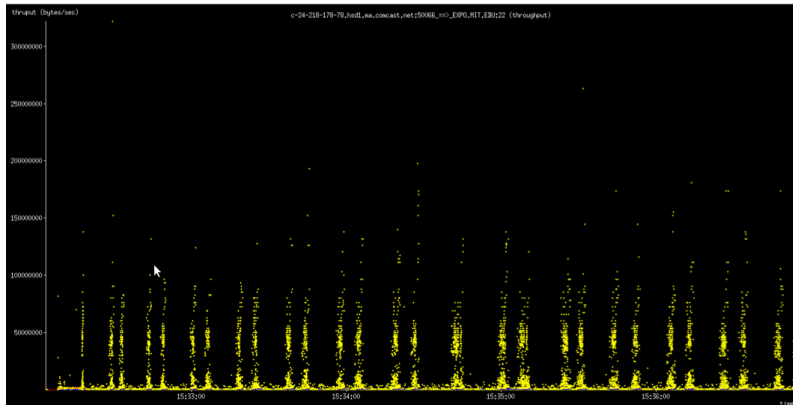


Figura:



Throughput graph

QoS: Negociação

Exemplos de descrição dos parâmetros:

- descrevendo as características de um fluxo: numa aplicação de conferência, é preciso largura de banda média de 1,5Mbps; atraso máximo de 150ms para evitar hiatos na palestra; o algoritmo de descompactação no destino pode produzir imagens aceitáveis com uma perda de 1 quadro em 100
- descrevendo capacidade de fluxo: uma rede pode fornecer conexões de largura de banda de 64kpbs; os algoritmos de enfileiramento garantem atrasos de menos de 10ms; o sistema de transmissão garante uma taxa de perda menor que 1 em 10^6 .

Exemplo de parâmetros: h.264

- codec/decoder de vídeo
- 1080p @ 3.7Mbps, 720p @ 2.2 Mbps, 480p @ 1.25 Mbps :
- perda de pacote aceitável até 25%⁶

⁶[urlhttp://x264dev.multimedia.cx/archives/249](http://x264dev.multimedia.cx/archives/249)

Exemplo de parâmetros: CELT

- codec/decoder de áudio
- 200 ms aceitavel, 20ms p/ música
- atraso de audio ; 20ms = 8 metros p/ som no ar
- latência: 5ms
- bitrate: 32kpbs @ 48kHz stereo

QoS: Negociação

Os parâmetros são interdependentes. Exemplos:

- taxa de perda depende de estouro do buffer e dados dependentes do tempo chegando atrasados; logo, quando maior a largura de banda e o atraso, menor será a taxa de perda
- quanto menor a largura de banda, em relação à carga, mais mensagens serão armazenadas e mais buffers serão necessários; quanto maior o buffer, mais provável que mensagens esperem outras que estão na frente para serem atendidas, e assim, maior será o atraso.

QoS I

Especificando parâmetros de QoS: largura de banda

- para MPEG, a compactação média está entre 1:50 e 1:100, a depender do conteúdo; logo, parâmetros de qualidade são citados como valores mínimo, médio e máximo.
- taxa de rajada (*burst*): considere 3 fluxos de 1Mbps: o primeiro transfere um único quadro de 1Mbit/s; o segundo é um fluxo assíncrono de elementos de animação com largura de banda média de 1Mbps; o terceiro envia amostra de som de 100bits a cada microssegundo. Os 3 fluxos exigem a mesma largura de banda, porém seus padrões de tráfego são diferentes. O parâmetro de rajada define o número máximo de

QoS II

elementos de mídia que podem chegar cedo, isto é, antes do que devam chegar, de acordo com a taxa normal.

- Outro exemplo sobre o conceito de rajada é: uma CPU transfere dados via canais ou barramento de duas maneiras, byte a byte ou então em blocos de bytes de cada vez; é esta modalidade de transferência em blocos é chamada de 'modo rajada'. [1]

QoS

- Método LBAP (linear-bounded arrival processes): define o número máximo de mensagens em um fluxo durante qualquer intervalo t como $Rt+B$, onde R é a taxa e B é o tamanho máximo da rajada.
- reflete bem dados multimídia: dados multimídia lidos do disco são geralmente distribuídos em blocos grandes, e recebidos das redes em pacotes pequenos; o parâmetro de rajada define a quantidade de espaço exigida no buffer para evitar perda.

Especificando Latência

- requisitos do fluxo: um quadro não deve permanecer em buffer, em média, por mais do que $1/R$, onde R é a velocidade de taxa de quadros de um fluxo, senão ocorrerá acúmulo de trabalho.
- requisitos da aplicação: ex. conferência, atrasos de ponta a ponta absolutos de não mais do que 150ms; reprodução de vídeo, resposta aos comandos Pausa e Parar teve ter latência máxima da ordem de 500ms
- jitter⁷: o uso de buffers resolve, mas há um limite.

⁷variação no período entre a distribuição de dois quadros subjacentes

QoS

Taxa de perda

- é o parâmetro mais difícil de especificar; resultam do cálculo de probabilidade estouro em buffers e mensagens atrasadas, baseados em pior caso ou distribuição padrão.
- taxas de perda dadas para período de tempo infinito são inúteis: qualquer perda em um tempo curto pode ultrapassar a taxa de longo prazo.

QoS

Moldagem de tráfego

- Termo que descreve o uso de buffers de saída para suavizar o fluxo de dados.
- Fluxos podem ser regulados pela inserção de um buffer na origem e um método de consumo.

Figura: QoS

Traffic shaping algorithms

(a) Leaky bucket



(b) Token bucket

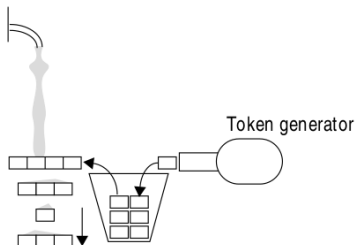


Figura: QoS

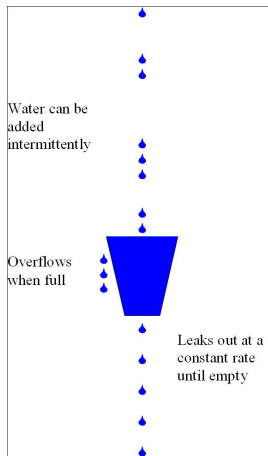
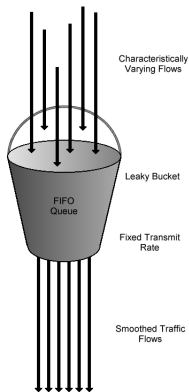


Figura: QoS



QoS - Especificações de fluxo

Um conjunto de parâmetros de qualidade de serviço são conhecidos como especificação de fluxo; há vários exemplos de especificações de fluxo e são todos semelhantes, compostos pelos itens:

- unidade de transmissão máxima e taxa de transmissão máxima determinam largura de banda máxima exigida pelo fluxo
- tamanho do balde com tokens e a taxa determinam a taxa de rajada do fluxo
- atraso mínimo que uma aplicação pode observar e o jitter máximo que ela pode aceitar
- número total aceitável de perdas em certo intervalo e número máximo de perdas consecutivas

Figura: Agrupamento dos parâmetros, especificação de fluxo RFC REC 1363

The 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

QoS - procedimentos de negociação

- os gerenciadores de qualidade de serviço estão nos nós
- estratégia simples de negociação: envio da especificação de fluxo para o gerenciador da QoS total; verificação no banco de dados local; encaminhamento para o próximo nó, até o destino final; no retorno da mensagem obtem-se a informação sobre a viabilidade do fornecimento de serviço
- aplicações têm requisitos de qualidade de serviço fixos
- é mais apropriado o sistema determinar qual tipo de QoS pode fornecer, e a aplicação decide se é aceitável
- é comum especificar um valor desejado e um pior valor
- exemplo: aplicação deseja largura de banda 1,5Mbps, mas pode manipular 1Mbps; o atraso deve ser 200ms, mas 300ms seria o pior caso aceitável.

QoS - procedimentos de negociação

- fluxo com vários destinos: a negociação bifurca de acordo com o fluxo de dados; são produzidos valores de pior caso para cada parâmetro da QoS; exemplos: protocolos SRP, ST-II e RCAP.
- situações com destinos heterogêneos: negociação iniciada pelo destino: RSVP; origens comunicam a existência dos fluxos; destinos se conectam aos nó mais próximo através do qual o fluxo passa a extrair dados de lá.

QoS - controle de admissão

- regula o acesso aos recursos para evitar sobrecarga e proteger pedidos que não possam ser por eles atendidos
- baseado em conhecimento geral da capacidade global do sistema e carga gerada por cada aplicação
- recursos com alocador único: simples para controlar a admissão
- recursos com pontos de acesso distribuídos: algoritmo de controle de admissão distribuído, para tratar concorrências e conflitos.

QoS - reserva de largura de banda

- para fornecer uma QoS garantida: reserva para sua largura de banda máxima
- aplicações que se tornam inúteis se houver quedas de qualidade
- exemplos: raio X (sintoma aparecer bem no momento de descarte de quadros); armazenamento de vídeos (falhas serão visíveis sempre quando reproduzir o vídeo).
- cálculo simples: em rede de largura de banda B podem ser admitidos s fluxos multimídia de largura de banda b_s , desde que $\sum b_s \leq B$
- uma rede token ring com 16Mb/s pode suportar até 10 fluxos de vídeo digital de 1,5Mb/s.

QoS - reserva de largura de banda

- alocar largura de banda de CPU exige conhecimento do perfil de execução do processo aplicativo, que depende do processador usado; não podem ser calculados com precisão.
- tempos de execução não são amplamente usados; têm margens de erro grandes e portabilidade limitada.
- há desperdício quando usa-se reserva baseada nos requisitos máximos.

QoS - multiplexação estática

- suposição: para grande número de fluxos, a largura de banda total permanece constante
- quando um fluxo enviar muitos dados, outro fluxo enviará poucos
- isso só ocorrerá em fluxos não-relacionados
- experiências mostram que o tráfego em ambientes típicos contradiz essa hipótese: o tráfego agregado mostra semelhança com os fluxos individuais dos quais é composto

Gerenciamento de recursos

o recurso precisa ter capacidade suficiente (desempenho) e estar disponível (escalonamento) quando a aplicação precisar

Ger. de recursos - escalonamento

- processos tem prioridades, determinada por critérios
- uso intenso de E/S recebem maior prioridade
- em multimídia, prazos finais para distribuição dos dados alteram o escalonamento
- exemplo típico: fluxo multimídia é recuperado do disco, enviado pela rede para uma estação destino, sincronizado com um fluxo proveniente de outra origem e exibido: escalonamento de vários recursos

Ger. de recursos - escalonamento

Escalonamento imparcial (fairness)

- fluxos concorrentes pelo mesmo recurso: imparcialidade para evitar que fluxos mal comportados ocupem largura de banda em demasia
- estratégia simples: escalonamento em rodízio em fluxos da mesma classe
- trabalhos que realizam esse escalonamento pacote por pacote, e bit por bit⁸.
- é possível estabelecer que, para certos fluxos, um número maior de bits seja transmitido por ciclo: enfileiramento imparcial ponderado (priorização no rodízio).

⁸pacotes não podem ser enviados bit a bit, mas dada uma taxa de quadros é possível calcular, para cada pacote, quando ele será enviado completamente

Ger. de recursos - escalonamento

Escalonamento em tempo real

- escalonamento EDF (Earliest-deadline-first): método tradicional que se adapta bem aos fluxos multimídia contínuos regulares
- prazos finais são associados aos itens de trabalho; itens com prazos finais mais adiantados são processos primeiro.
- escalonamento RM (Rate-monotonic): baseado nos elementos que existem por um tempo maior
- quanto maior a velocidade dos itens de um fluxo, maior é a prioridade do fluxo

Adaptação de FLuxo

- quando uma QoS não pode ser garantida, a aplicação precisa de adaptar
- forma mais simples: eliminar informações
- facilmente realizável em áudio, onde as amostras são independentes, mas notado imediatamente pelo ouvinte⁹
- exclusões de quadros em vídeos Motion JPEG são toleráveis: os quadros têm posições livres
- MPEG: interpretação de um quadro depende dos valores de quadros adjacentes: omissão pode amplificar erros

⁹Demonstra-se que as pessoas são menos tolerantes a erros de áudio do que erros de vídeo; assim, em competições pode ser atribuído ao áudio maior prioridade.[4]

Motion JPEG x MPEG¹⁰

With the Motion JPEG format, the three images in the above sequence are coded and sent as separate unique images (I-frames) with no dependencies on each other.



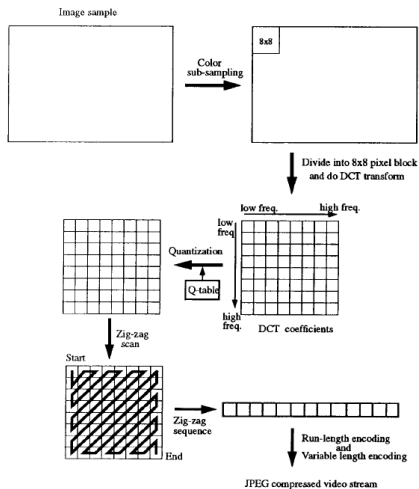
With difference coding, only the first image (I-frame) is coded in its entirety. In the two following images (P-frames), references are made to the first picture for the static elements, i.e. the house. Only the moving parts, i.e. the running man, are coded using motion vectors, thus reducing the amount of information that is sent and stored.

— Transmitted - - - Not transmitted



¹⁰[http:](http://www.axis.com/products/video/about_networkvideo/compression.htm)

Compressão JPEG



Adaptação de FLuxo

- largura de banda insuficiente e dados não eliminados: atraso de fluxo aumentará
- aceitável em aplicações não-interativas, apesar de possibilitar estouro de buffer
- aplicações interativas: se um fluxo estiver atrasado em seu tempo de término designado, sua taxa de término deverá ser aumentada até que ele esteja novamente no tempo certo: os quadros devem aparecer na saída assim que estiverem disponíveis.

Adaptação de FLuxo: mudanças na escala

- adaptação realizada apenas no destino em um fluxo: sobrecarga persistirá
- adaptar um fluxo para a largura de banda disponível: mudança de escala
- operação fundamental: fazer uma nova amostragem do sinal
- exemplos: áudio, eliminação de um canal estéreo; mudança de taxa de amostragem

Adaptação de FLuxo: mudanças na escala

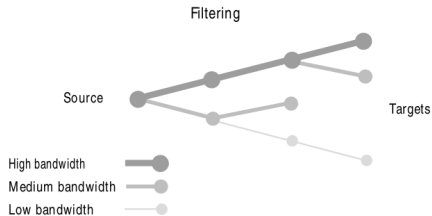
- vídeo:
 - temporal: diminui o número de quadros transmitidos dentro de um intervalo; apropriado para quadros individuais independentes
 - espacial: reduz o número de pixels em cada imagem (diferentes resoluções);
 - frequência: modifica o algoritmo de compactação aplicado às imagens
 - amplitude: reduz intensidades de cores de cada pixel da imagem;
 - espaço de cores: reduz o número de entradas no espaço de cores; cores para escala de tons;
- Podem ser usadas combinações destas mudanças

Adaptação de FLuxo: mudanças na escala

- um monitor controla tempos de chegada de mensagens
- mensagens atrasadas indicam gargalos
- uma mensagem para *diminuir escala* é enviada, e a largura do fluxo é reduzida; após algum período de tempo, a origem aumenta novamente a escala

Adaptação de FLuxo: filtragem

- mudança de escala modifica o fluxo na origem; nem todos os destinos teriam problemas para manipular o fluxo original
- a filtragem fornece a melhor QoS possível para cada destino



Filtragem II [6]

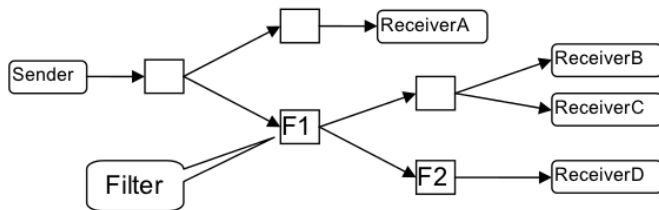


Figure 2. Multicast communication

Sender sends video at 30 frames per second (full color)

ReceiverA receives video at 30 frames per second (full color)

ReceiverB and ReceiverC receives video at 10 frames per second (full color)

ReceiverD receives video at 10 frames per second (grey-scale)

Figura: Distribuição em camadas[7]

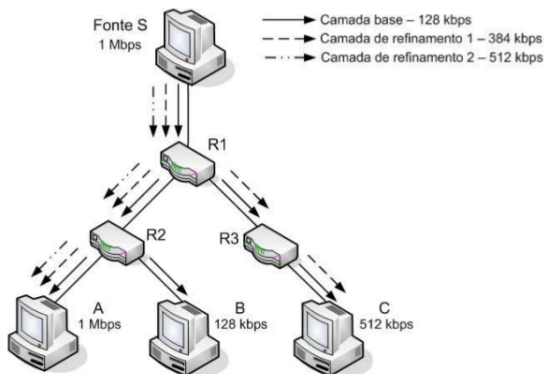


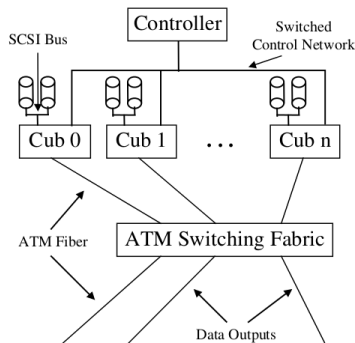
Figura 3.4. Um exemplo da codificação em camadas.

Servidor Tiger: objetivos do projeto

- vídeo sob demanda para grande número de usuários; exemplo: fornecimento de filmes para clientes pagantes; execução de operações de pausa, retrocesso e avanço rápido; poucos filmes populares, resultado em várias reproduções concorrentes deslocadas no tempo
- QoS: fornecimento com uma velocidade constante, flutuação (jitter) máxima (pequena) definida pela quantidade de buffers disponíveis nos clientes e taxa de perda mínima
- sistema com mudança de escala e distribuído: arquitetura extensível (+ computadores) para suportar até 10.000 clientes
- hardware de baixo custo: pc's comerciais com unidades de disco padrão
- tolerante a falhas: funcionamento sem degradação notável após falha de computadores servidores ou discos

Servidor Tiger: arquitetura

todos os componentes são produtos de prateleira; o controlador é:
ponto de contato com os clientes; servidor de hora local; outras
pequenas tarefas[2]



Servidor Tiger: armazenamento

- filmes são armazenados de forma segmentada (strips) entre todos os discos
- a perda de um disco resulta em uma lacuna em cada filme
- tratamento: espelhamento de armazenamento e outros descritos a seguir

Servidor Tiger: segmentação (stripping)

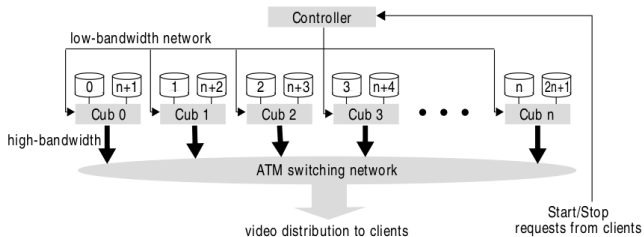
- filme é dividido em blocos, em torno de 1s, ocupando 0,5 Mb;
- blocos que formam um filme (7000, para 2hs) são armazenados em sequência de discos; um filme pode começar em qualquer disco
- ao atingir o disco mais alto, o filme *circula* para o disco 0 e o processo continua

Servidor Tiger: espelhamento

- um bloco é dividido em partes, chamadas secundários
- quando um cub falhar, a carga cai sobre outros cubs, não sobre um apenas
- número de secundários por bloco: fator de desagrupamento d (tipicamente entre 4 e 8)
- secundários de um bloco i são armazenados nos cubs $i+1$ a $i+d$
- desde que existam mais do que d cubs, nenhum dos discos é ligado no mesmo cub do disco i

Servidor Tiger: armazenamento

Figure 15.10 Tiger video file server hardware configuration



Servidor Tiger: escalonamento distribuído

- organização por repartições (slots): cada repartição representa o trabalho a ser feito para reproduzir um bloco de um filme (lê-lo do disco relevante e transferí-lo para a rede ATM)
- existe uma repartição para cada cliente em potencial receber um filme (visualizador); cada repartição ocupada representa um visualizador recebendo um fluxo de dados de vídeo em tempo real

Servidor Tiger: escalonamento distribuído

O estado do visualizador é representado no escalonamento pelo(a):

- endereço do computador cliente
- ID do arquivo sendo reproduzido
- posição do visualizador no arquivo (próximo bloco a ser distribuído no fluxo)
- número de sequência no visualizador (a partir do qual calcula-se tempo de distribuição para o próximo bloco)
- algumas informações de contabilidade

Servidor Tiger: tolerância a falhas

- recuperação de dados das cópias secundárias espelhadas, quando um bloco primário torna-se indisponível

Servidor Tiger: suporte de rede

- blocos de cada filme são passados para a rede ATM pelo cub que o contém
- rede ATM possui QoS para entregar o bloco ao cliente
- cliente apenas verifica o número de sequência do bloco recebido
- secundários sendo servidos: os d cubs distribuem na rede (em sequência) e o cliente reúne-os e monta-os em seu buffer

Outras funções

- implementação inicial: escalonamento e distribuição eram manipulados pelo controlador
- problema: único ponto de falha e gargalo de desempenho em potencial
- solução: escalonamento refeito, onde os cubs executam um algoritmo distribuído, em tempo não programado, em resposta aos comandos do controlador
- alucinação coerente: termo usado para a implementação distribuída de um objeto compartilhado
- nenhum cub tem uma cópia de todo o agendamento, mas age como se houvesse um estado global coerente

Servidor Tiger: desempenho e escalabilidade

- 1994: 5 PC's pentim 133MHz, 48Mb RAM, 3 discos SCSI de 2Gb, 1 controlador ATM, Windows NT
- 68 clientes de filmes, sem falhas, distribuição perfeita
- 1 cub danificado (3 discos), perda de dados de 0,02%

Adicional: aplicações[5]

- operações cirúrgicas remotas
- controle remoto exploratório de robôs

Adicional: definições[5]

- A estrutura da Internet não é suficiente para fornecer a qualidade, confiabilidade e interatividade necessária para conteúdo multimídia
- Os sistemas multimídia distribuídos adicionam arquiteturas e protocolos ao topo da Internet (LAN's) para garantir níveis de qualidade
- Clientes adicionais não precisam degradar o sistema: eles podem ser recusados se os recursos são escassos

Introdução

Características dos dados multimídia

Gerenciamento de qualidade de serviço

Gerenciamento de recursos

Adaptação de fluxo

Estudo de caso: servidor de arquivos de vídeo Tiger

Referências

Adicionais

FIM

FIM



http://www.proz.com/kudoz/english_to_portuguese/computers%3A_hardware/1560052-burst_transfer_rate.html, 2011.



William J. Bolosky, Robert P. Fitzgerald, and John R. Douceur.

Distributed schedule management in the tiger video fileserver.
SIGOPS Oper. Syst. Rev., 31:212–223, October 1997.



George Coulouris, Jean Dollimore, and Tim Kindberg.
Sistemas Distribuídos: Conceitos e Projetos.
Bookman, Porto Alegre, 4 edition, 2007.



Victor O. K. Li, Wanjiun Liao, and Student Member.
Distributed multimedia systems.
In *Proceedings of the IEEE*, pages 1063–1108, 1997.



James Maxlow.

Distributed Multimedia Systems, 2003.



Gu Mingyang.

Issues for distributed multimedia systems.



Igor Monteiro Moraes.

Distribuição de Vídeo sobre Redes Par-a-Par.

PhD thesis, COPPE/PEE/UFRJ, 2009.