

SSC-144

Redes de Alto Desempenho
2010

Nas aulas passadas:

- ❑ Sinais
- ❑ Áudio
- ❑ Vídeo
- ❑ Formatos
- ❑ Padrões
- ❑ Caracterização para especificação de demanda de armazenamento e transmissão em redes

Nesta aula:

- ❑ Protocolos de comunicação

Multimídia em Redes de Computadores

Luciana A. F. Martimiano

Paulo Sérgio Lopes de Souza

* os slides desta aula foram adaptados do material desenvolvido pelo *Prof. Dr. José Augusto Suruagy Monteiro*, disponível em: <http://www.nuperc.unifacs.br/suruagy/redes/index.html>

Multimídia em redes

Características fundamentais:

- sensível a atrasos e *jitter*
- tolerante a perdas → técnicas de recuperação de pacotes
- transdutor minimizam esses problemas (*RealPlayer*, *Windows Media Player*, *Quick Time*)

Classes de aplicações multimídia:

- fluxo contínuo armazenado (*streaming*)
- fluxo contínuo de áudio/vídeo ao vivo → tempo real unidirecional (rádio ou um jornal pela Internet)
- vídeo/áudio interativos em tempo real (VoIP, videoconferência)

Multimídia em redes

Protocolos

- fluxo contínuo
 - RTSP (*Real Time Streaming Protocol*)
- aplicações interativas em tempo real
 - RTP (*Real Time Protocol*) e RTCP (*Real Time Control Protocol*)
 - H.323
 - SIP (*Session Initiation Protocol*)

Multimídia em redes: classes de aplicações

Fluxo MM armazenado

- ❑ clientes solicitam arquivos de áudio/vídeo de servidores e efetuam o *pipeline* da recepção através da rede e apresentação
- ❑ **fluxo contínuo**
- ❑ interatividade: usuário pode controlar a operação
 - semelhante ao VCR
 - pausa, retomada, avanço rápido, retorno
- ❑ atraso: pode haver um atraso desde o pedido do cliente até que a apresentação tenha início

Multimídia em redes: classes de aplicações

Tempo real unidirecional:

- semelhante às estações de TV e rádio convencionais, mas com distribuição através da Internet
- fluxo contínuo
- não interativa
 - objetivos: escutar e ver

Tempo real interativo:

- Videoconferência e VoIP
- requisitos de atraso mais exigentes
 - vídeo: < 150 mseg é aceitável
 - áudio: < 150 mseg é bom, <400 mseg é aceitável
- muito sensível ao *jitter* e menos sensível a perdas

Recuperação da perda de pacotes

❑ O que é perda?

pacote nunca chega ou chega mais tarde do que o tempo previsto para sua reprodução

❑ Técnicas:

- FEC (*Forward Error Correction*): duas implementações
- Intercalação

Recuperação da perda de pacotes

1º esquema FEC – simples

- adicionar informações redundantes ao fluxo original
- para cada grupo de n pedaços, criar um pedaço redundante efetuando XOR dos n pedaços originais
- pode reconstruir os n pedaços originais se houver **no máximo um pedaço perdido**

stream original



XOR dos pacotes anteriores

Recuperação da perda de pacotes

1º esquema FEC – simples

- se n pequeno, grande parte dos pacotes perdidos podem ser recuperados
 - aumenta taxa de transmissão ($1/n$)
 - $n = 3$, taxa de 33%
 - $n = 4$, taxa de 25%
 - atraso da reprodução, pois receptor deve esperar grupo de pacotes inteiro chegar antes de começar a reproduzir
 - atraso de reprodução é menor
- se n grande
 - menos pacotes perdidos podem ser recuperados
 - atraso de reprodução é maior
 - menos desperdício de banda
 - $n = 10$, 10%

Recuperação da perda de pacotes

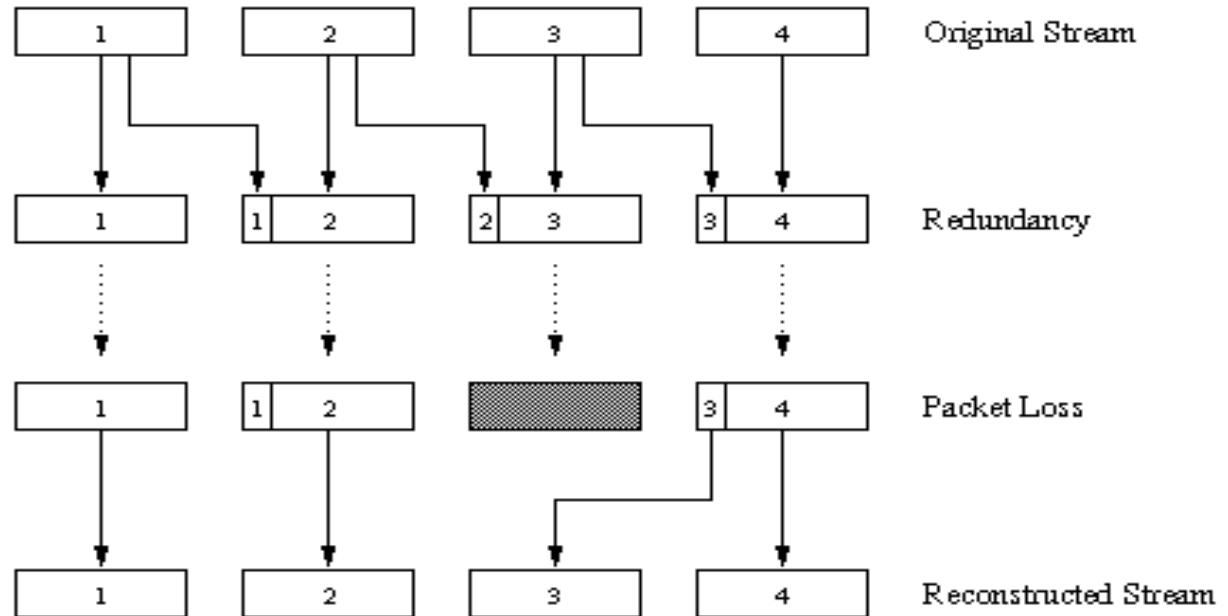
2º esquema FEC: Redundância

- transmite um fluxo de menor qualidade de “carona” → informação redundante

PCM de 64 kbps
(nominal)

GSM de 13 kbps
(redundante)

- transmissor cria pacote com o **n-ésimo** pedaço do fluxo nominal e adicionando a ele o **(n-1)-ésimo** pedaço do fluxo redundante

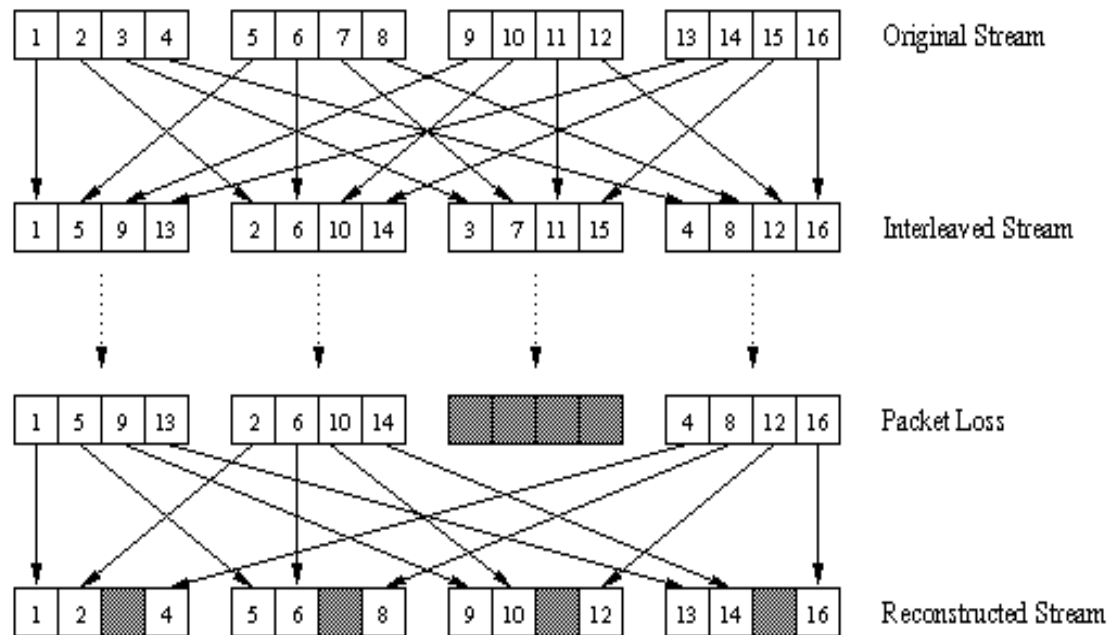


- perdas não consecutivas podem ser recuperadas pelo receptor
- podem ser adicionados o **(n-1)-ésimo** e o **(n-2)-ésimo** pedaços de baixa taxa de transmissão

Recuperação da perda de pacotes

Intercalação

- ❑ pedaços são quebrados em unidades menores
- ❑ ex: 4 unidades de 5 milissegundos por pedaço
- ❑ intercala os pedaços como mostrado no diagrama
- ❑ pacote agora contém pequenas unidades de pedaços diferentes
- ❑ remonta pedaços no receptor
- ❑ **desvantagem**: aumenta latência
- ❑ **vantagens**: não requer mais largura de banda; se pacote é perdido, ainda há muito de cada pedaço



RTSP – Real Time Streaming Protocol (fluxo contínuo)

RTSP (RFC 2326)

- ❑ protocolo cliente-servidor da camada de aplicação
- ❑ usuário pode controlar a apresentação:
 - retorno, avanço rápido, pausa, retomada, reposicionamento, etc.

O que o RTSP faz:

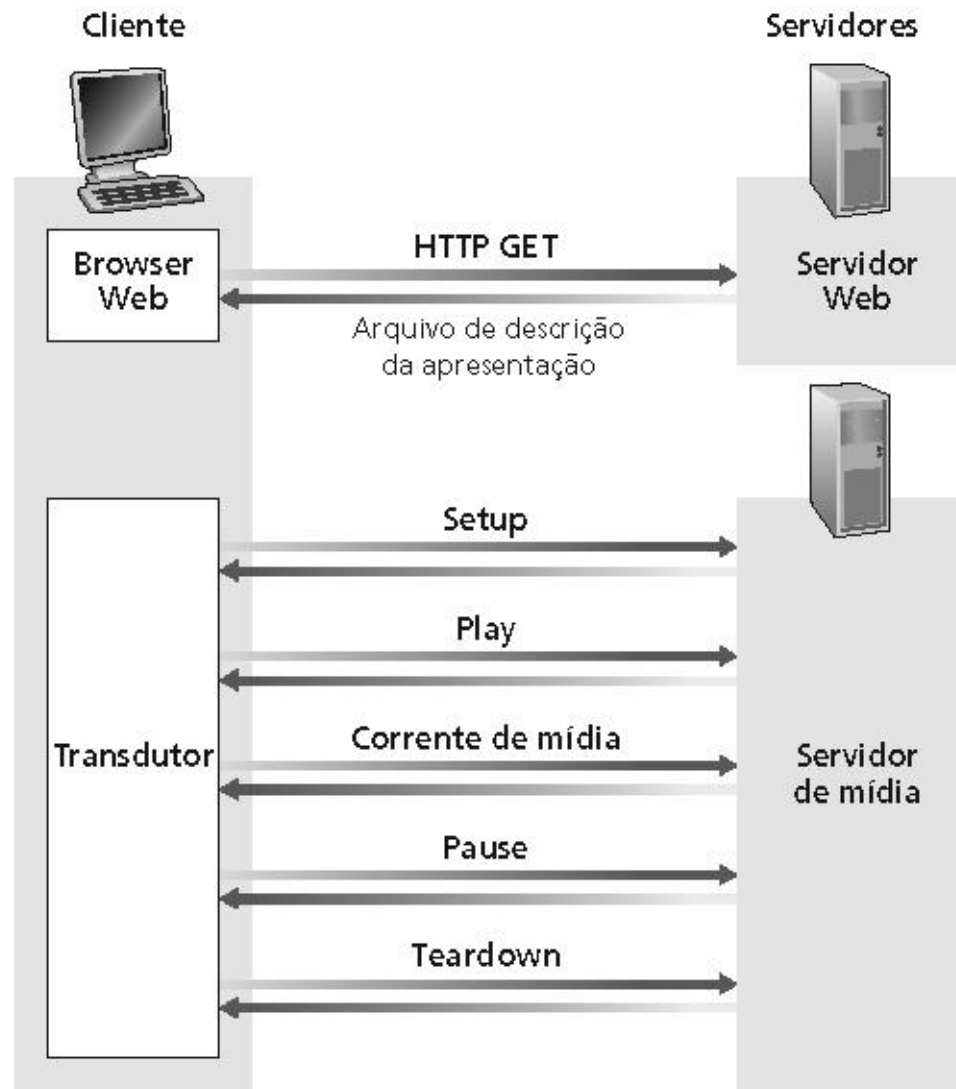
- permite o cliente controlar a transmissão de um fluxo contínuo → pausa/reinício, reposicionamento, ...
- controla o estado do cliente
- conhecido como **protocolo fora de banda**
 - mensagens de controle RTSP usam números de portas diferentes do fluxo da mídia, e são, portanto, enviadas fora da banda (porta 554)
 - fluxo de mídia, cuja estrutura de pacote não é definida pelo RTSP, é considerado “dentro da banda” (UDP)

RTSP – Real Time Streaming Protocol

O que o RTSP não faz:

- não define como áudio e vídeo são encapsulados
- não restringe como a mídia tipo *stream* é transportada (UDP/TCP)
- não especifica como o transdutor da mídia armazena áudio e vídeo

O RTSP inicia e controla a entrega



O RTSP inicia e controla a entrega

- ❑ *browser* cliente solicita e obtém uma descrição da apresentação multimídia, que pode consistir de diversos fluxos de mídia
- ❑ *browser* chama apresentador da mídia (aplicação suporte) baseado no tipo do conteúdo da descrição da apresentação
- ❑ descrição da apresentação inclui referências aos fluxos de mídia, usando o método URL `rtsp://`
- ❑ transdutor envia um pedido RTSP SETUP; o servidor envia uma resposta RTSP SETUP
- ❑ transdutor envia um pedido RTSP PLAY; o servidor envia uma resposta RTSP PLAY
- ❑ servidor de mídia “bombeia” o fluxo da mídia
- ❑ transdutor pode enviar um pedido RTSP PAUSE na seqüência; o servidor envia uma resposta RTSP PAUSE
- ❑ transdutor envia um pedido RTSP TEARDOWN; o servidor envia uma resposta RTSP TEARDOWN

Exemplos de mensagens RTSP

```
C: SETUP rtsp://audio.example.com/twister/audio RTSP/1.0
Cseq: 1
Transport: rtp/udp; compression; port=3056; mode=PLAY
```

```
S: RTSP/1.0 200 1 OK
Cseq: 1
Session 4231
```

```
C: PLAY rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
Cseq: 2
Session: 4231
Range: npt=0-
```

```
S: RTSP/1.0 200 Ok
Cseq: 2
Session: 4231
```

npt: normal play time (tempo no qual a transmissão irá começar: de 0 segundos até o final; pode ser um intervalo de tempo, ou um horário (absolute time - utc))

Exemplos de mensagens RTSP

```
C: PAUSE rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
  Cseq: 3
  Session: 4231
  Range: npt=37

S: RTSP/1.0 200 Ok
  Cseq: 3
  Session: 4231

C: TEARDOWN rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
  Cseq: 4
  Session: 4231

S: RTSP/1.0 200 Ok
  Cseq: 4
  Session: 4231
```

RTP – *Real Time Protocol*

- ❑ RTP especifica uma estrutura para encapsular e transportar dados de áudio e de vídeo: RFC 1889
- ❑ pacote RTP fornece:
 - identificação do tipo da carga
 - numeração da seqüência de pacotes
 - carimbo de tempo
- ❑ Áudio e vídeo são encapsulados em pacotes RTP
- ❑ pacotes RTP são encapsulados em segmentos UDP
 - pode ser visto como uma subcamada da camada de transporte
 - pode ser visto como um protocolo da camada de aplicação
 - RTP é integrado à aplicação
- ❑ interoperabilidade, pois é independente da aplicação
- ❑ baseado em sessões

RTP e QoS

- ❑ RTP **não** provê
 - nenhum mecanismo para garantir a entrega em tempo dos dados
 - nenhuma outra garantia de qualidade de serviço
- ❑ Para fornecer QoS à uma aplicação, a Internet deve permitir que a aplicação **reserve recursos da rede** ou permitir **prioridades de classes de serviços**
- ❑ Encapsulamento RTP é visto apenas nos sistemas terminais – não é visto por roteadores intermediários
 - roteadores provendo o serviço tradicional
 - Internet de melhor esforço não faz nenhum esforço adicional para garantir que os pacotes RTP cheguem ao destino em tempo

Fluxos RTP

- permite que cada fonte tenha o seu próprio fluxo de pacotes RTP
 - ex: câmera/microfone em uma videoconferência entre dois participantes
 - 4 fluxos RTP podem existir:
 - 2 p/ áudio e 2 p/ vídeo
 - » um em cada direção
 - no entanto, técnicas de codificação (ex: MPEG1 e MPEG2) podem unir áudio e vídeo num único fluxo
 - então apenas um fluxo RTP é gerado em cada direção
- comunicação não precisa ser *unicast*
 - uma sessão RTP pode ser *multicast* (muitos-para-muitos)
 - todos os transmissores e fontes tipicamente enviam seus fluxos RTP com o mesmo endereço *multicast*

Cabeçalho RTP



RTP Header

12 bytes

tipo da carga (7 bits)

campo usado para indicar o tipo de codificação

carga útil pode variar no meio de uma conferência

áudio - tipo de carga 0: PCM lei μ , 64 Kbps

áudio - tipo de carga 3, GSM, 13 Kbps

vídeo - tipo de carga 31. H.261

vídeo - tipo de carga 33, vídeo MPEG2

número de seqüência (16 bits): número de seqüência é incrementado para cada pacote RTP enviado; útil também para detectar perda de pacotes e restaurar seqüência

campo de SSRC (32 bits): identifica a fonte de um fluxo RTP. Cada fluxo numa sessão RTP deve possuir um SSRC distinto

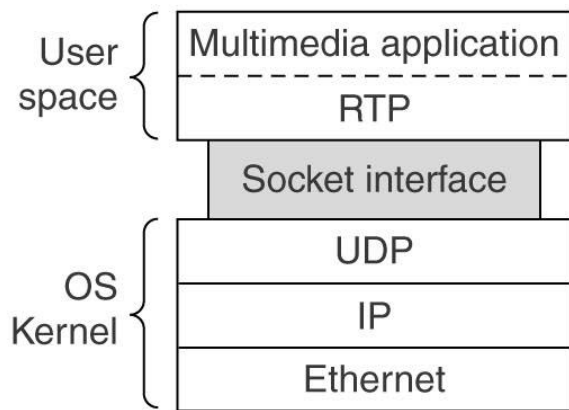
Cabeçalho RTP



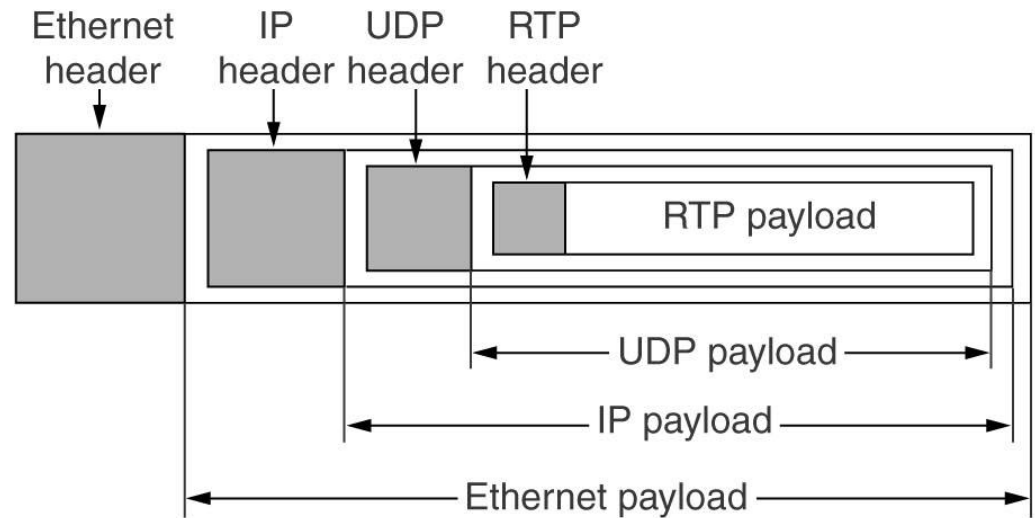
RTP Header

- ❑ **campo de carimbo de tempo (32 bits)**
 - é o instante de amostragem do primeiro byte no pacote de dados RTP. Receptor pode usar esse campo para remover o *jitter* dos pacotes e prover uma reprodução síncrona. Carimbo de tempo é derivado a partir de um relógio de amostragem no transmissor.
- ❑ **outros campos:**
 - **versão**

Empacotamento do pacote RTP



(a)



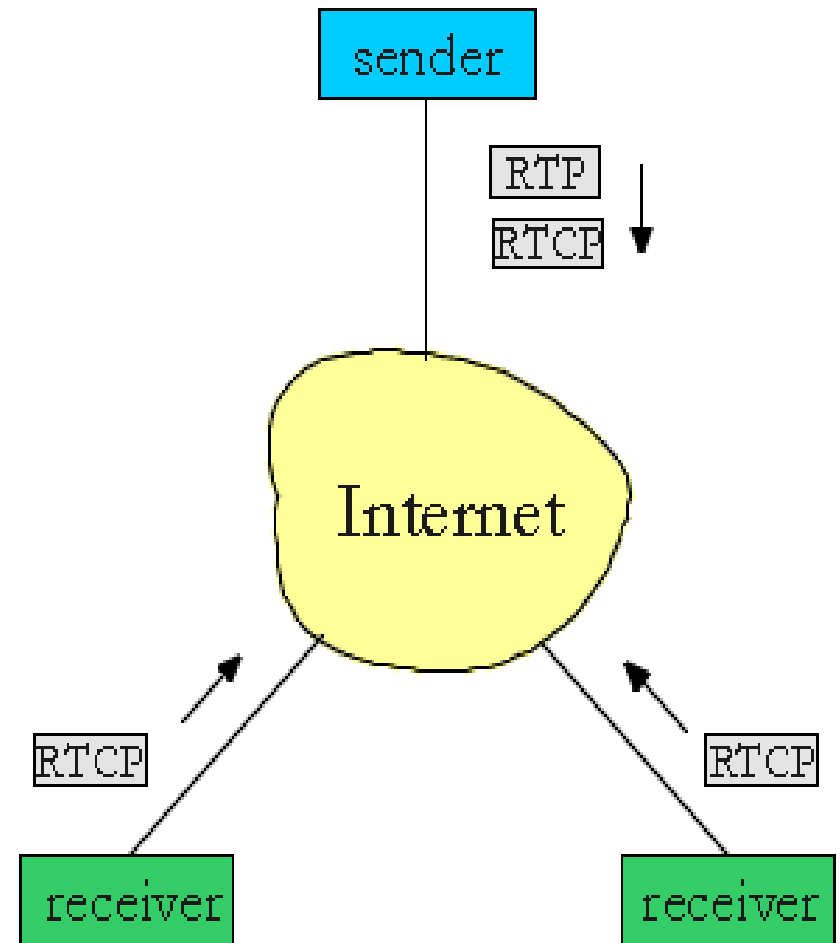
(b)

RTCP – *Real Time Control Protocol*

- ❑ trabalha em conjunto com o RTP
- ❑ cada participante numa sessão RTP periodicamente transmite pacotes de controle RTCP para todos os demais participantes (*multicast*)
- ❑ não contém áudio ou vídeo
- ❑ contém relatos do transmissor e/ou receptor sobre **estatísticas úteis para as aplicações**
 - número de pacotes enviados,
 - número de pacotes perdidos,
 - *jitter* entre chegadas, ...
- ❑ realimentação de informação para as aplicações pode ser usada para controlar o desempenho e para finalidades de diagnóstico
 - transmissor pode modificar as suas transmissões baseado na realimentação

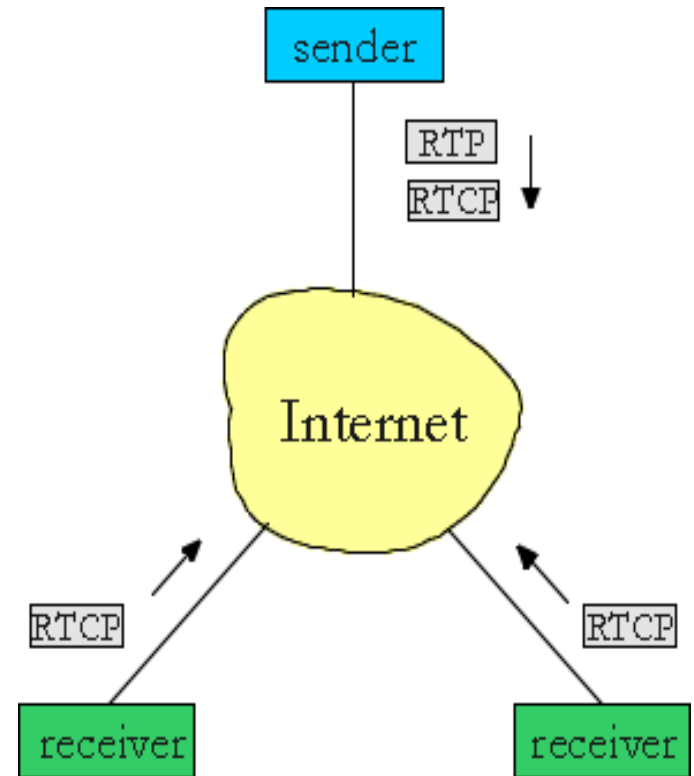
RTCP

- para uma sessão RTP há tipicamente um único endereço *multicast*
 - todos os pacotes RTP e RTCP pertencentes à uma sessão usam o endereço *multicast*
- pacotes RTP e RTCP são diferenciados uns dos outros através do uso de números de portas distintos (RTP = porta n , RTCP= porta $n+1$)
- tráfego RTCP pode ser reduzido à medida que cresce a demanda RTP
 - limite padrão é de 5% da largura de banda da sessão



RTCP

- 5% da largura de banda da sessão, sendo 75% para os receptores e 25% para o remetente
- **Exemplo:**
 - Tráfego total de 2 Mbps (4 receptores com 500 kbps cada um)
 - 5% → 100 kbps
 - 75% → 75 kbps
 - 25% → 25 kbps
 - Tráfego individual (500 kbps)
 - 5% → 25 kbps
 - 75% → 18,75 kbps
 - 25% → 6,25 kbps



Pacotes RTCP

Pacotes do receptor:

- ❑ fração de pacotes perdidos, último número de seqüência, *jitter* médio entre chegadas

Pacotes do transmissor:

- ❑ SSRC (fonte) do fluxo RTP, tempo atual, número de pacotes enviados e número de bytes enviados

Pacotes de descrição do transmissor:

- endereço de e-mail do transmissor, nome do transmissor, o SSRC do fluxo RTP associado
- pacotes fornecem mapeamento entre o SSRC e o nome do usuário/*host*

Pacotes RTCP

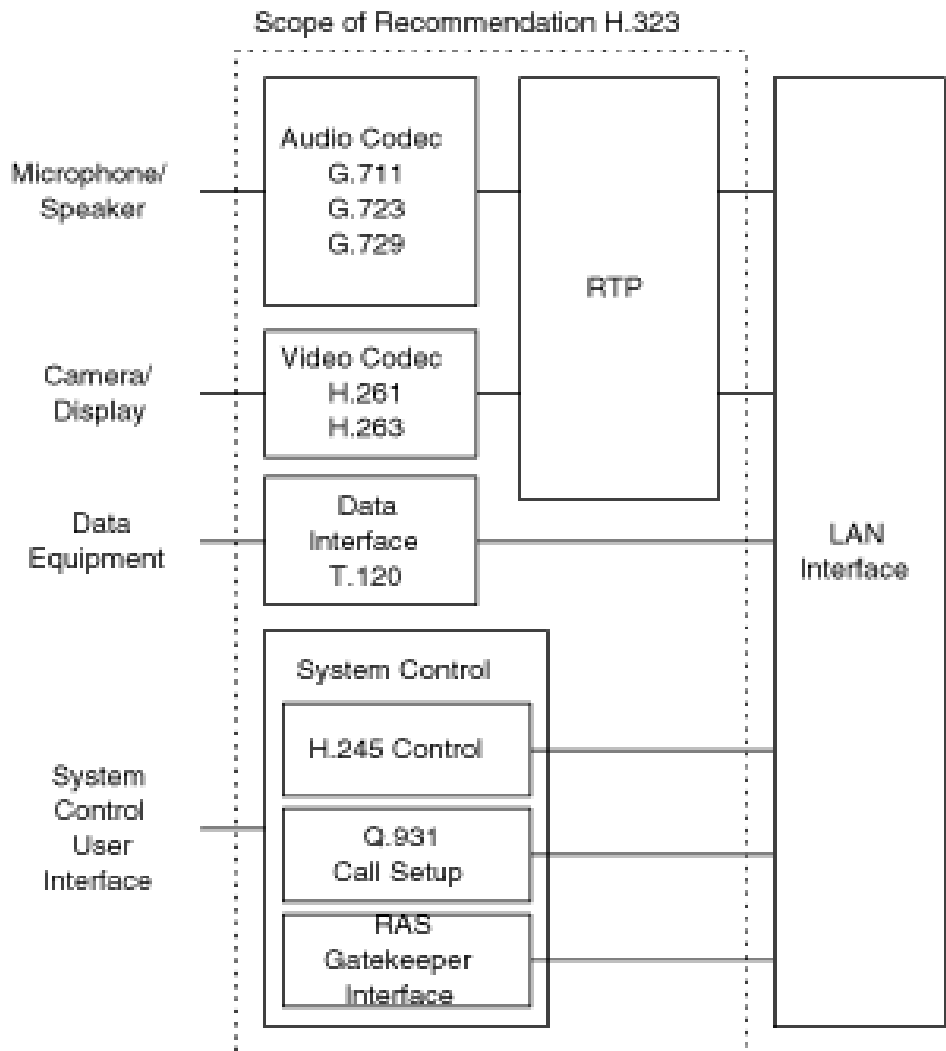
- ❑ RTCP pode ajudar a sincronizar diferentes fluxos (vídeo e áudio)
- ❑ Pacotes RTCP também são encapsulados em segmentos UDP

H.323 – uma visão geral

- ❑ padrão para videoconferências entre sistemas finais
- ❑ recomendação ITU (*International Telecommunication Union*)
- ❑ H.323 especifica como:
 - equipamentos fazem e recebem chamadas
 - equipamentos negociam codificações comuns de áudio/vídeo
 - blocos de áudio e vídeo são encapsulados e enviados através da rede via RTP
 - áudio e vídeo são sincronizados
 - equipamentos se comunicam com seus respectivos *gatekeepers*
 - telefones IP e os telefones convencionais se comunicam pelos *gateways*

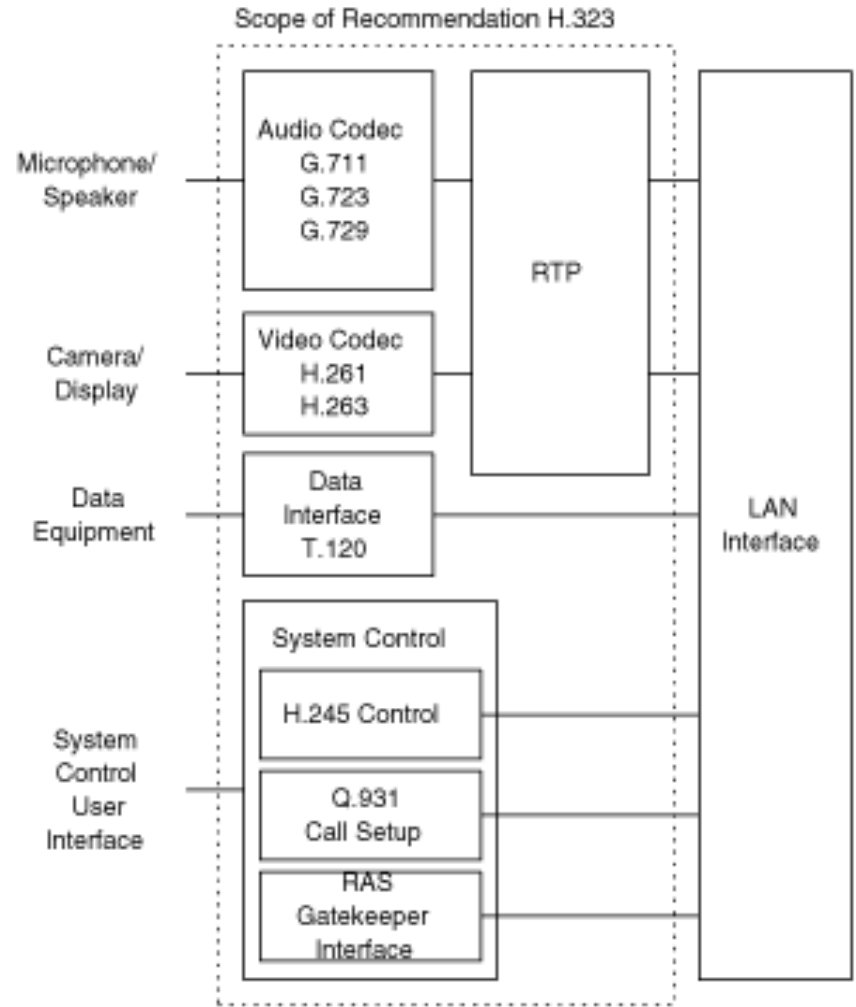
Padrões utilizados pelo H.323

- ❑ **G.711** – padrão do ITU para compressão de voz
- ❑ **H.261** – padrão de vídeo
- ❑ **RTP** - protocolo para encapsular blocos de dados de mídia em pacotes



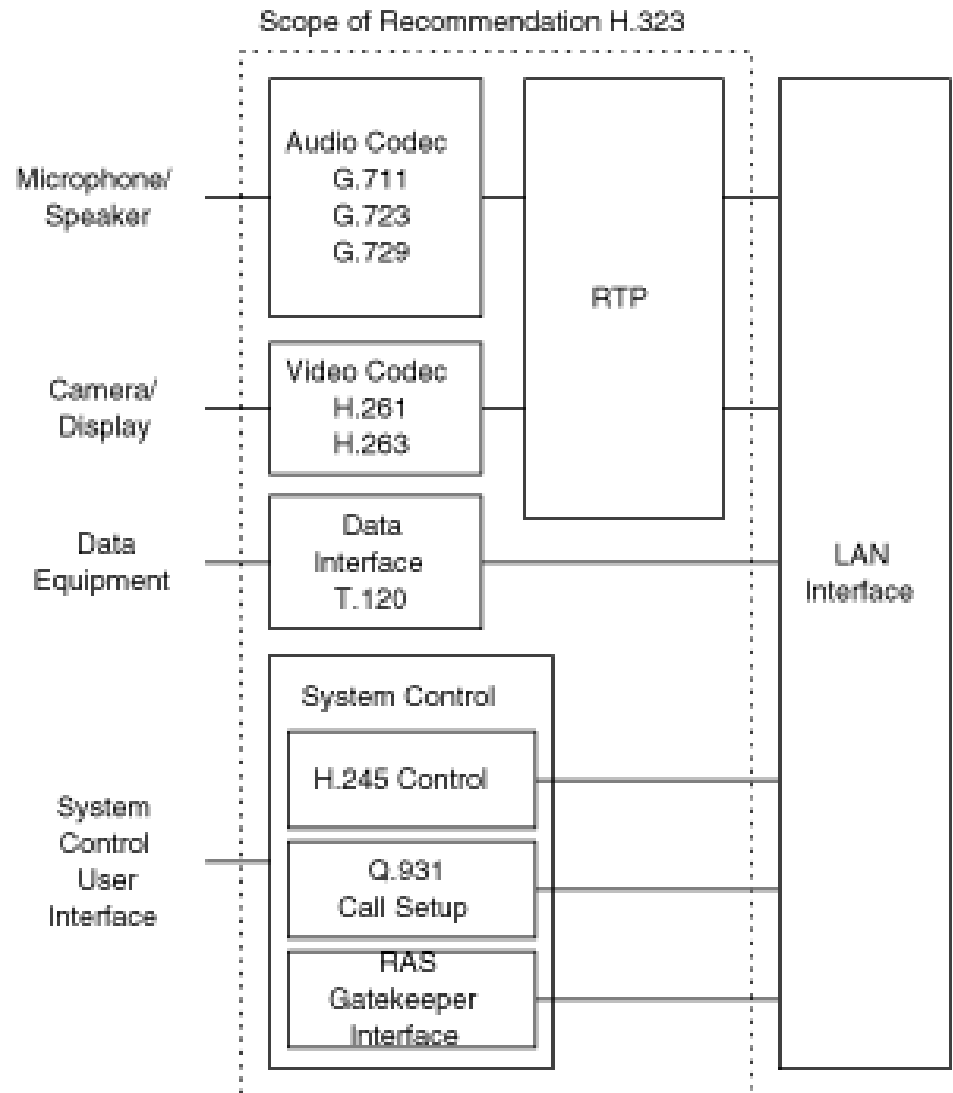
Padrões utilizados no H 323

- ❑ **T.120** – *suite* de protocolos que permite criar produtos e serviços para aplicações em tempo real que sejam **compatíveis**, assim usuários podem participar de uma videoconferência, por exemplo, utilizando redes e conexões diferentes
- ❑ Permite estabelecer conexões, transmitir e receber dados
- ❑ Pode ser utilizado para **empacotar** e **enviar** dados (Microsoft usa no Netmeeting)



Padrões utilizados pelo H.323

- ❑ **H.245** – protocolo de controle “fora da banda”:
 - permite negociação do padrão de compressão a ser utilizado
- ❑ **Q.931** – protocolo de sinalização para estabelecer e encerrar chamadas (TCP)
 - fornece as funcionalidades da telefonia tradicional
- ❑ **RAS** (Registro/Admissão/Status)
 - protocolo para a comunicação com um *gatekeeper* (caso esteja presente)



Canal de controle H.245 de chamada

- ❑ fluxo H.323 pode conter múltiplos canais para tipos diferentes de mídia RTP
- ❑ um canal de controle de chamadas (H.245) por sessão H.323
- ❑ o canal de controle H.245 é um canal confiável (TCP)
- ❑ tarefas do H.245:
 - abrir e fechar canais de mídia
 - trocar informações para acordo sobre o algoritmo de codificação antes de transmitir
- ❑ nota: o uso do H.245 para aplicações multimídia interativas em tempo real é **análogo ao RTSP para mídia contínua armazenada**

Pilha do H.323

Speech	Control			
G.7xx	RTCP	H.225 (RAS)	Q.931 (Call signaling)	H.245 (Call control)
RTP				
UDP			TCP	
IP				
Data link protocol				
Physical layer protocol				

SIP (*Session Initiation Protocol*) (RFC 3261)

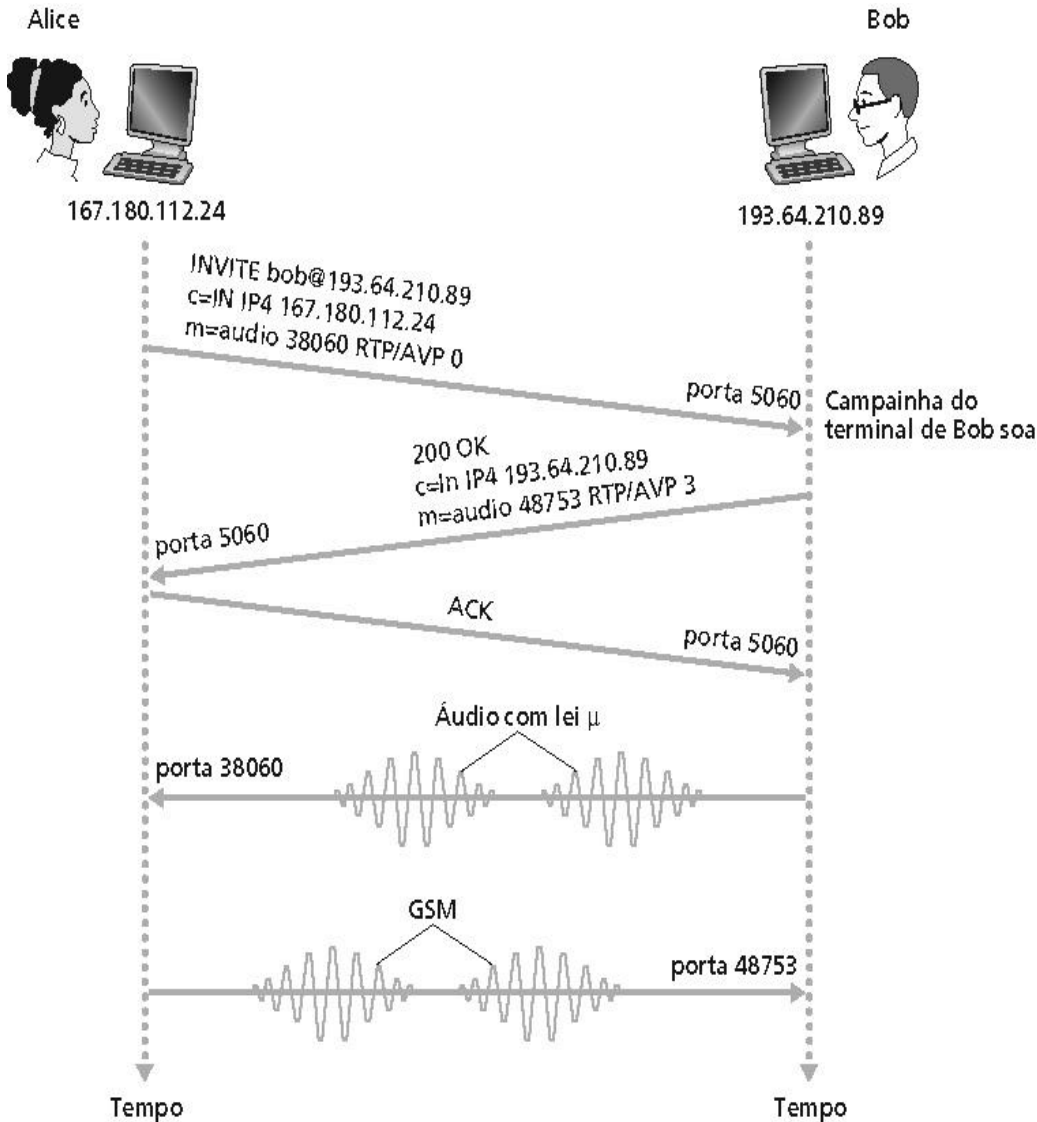
❑ Características:

- prover mecanismos para estabelecer chamadas em rede IP
- prover mecanismos para determinar endereço IP de quem é chamado
- prover mecanismos para gerenciar chamadas
- inicializar e finalizar chamadas de voz, videoconferências, texto
- permitir chamadas de IP para telefones comuns
- criptografia e autenticação
- também é um **protocolo “fora de banda”**

❑ Ações:

- estabelecer chamadas
- endereços SIP
- mensagens SIP
- tradução de nome e localização de usuário

SIP - estabelecer chamadas



A mensagem `INVITE` do SIP de Alice indica seu número de porta e endereço IP. Indica a codificação que Alice prefere receber (PCM lei- μ)

Bob envia a mensagem `200 OK` e indica seu número de porta, endereço IP e codificação preferida (GSM)

O número de porta padrão do SIP é 5060

SIP - estabelecer chamadas

- ❑ Negociação do Codec:
 - Suponha que Bob não tenha o codificador do PCM para lei μ
 - Bob responderá então com 606 Not Acceptable Reply e listará os codificadores que ele pode usar
 - Alice pode então enviar uma nova mensagem INVITE, anunciando um codificador apropriado
- ❑ Rejeitando a chamada
 - Bob pode rejeitar com respostas “ocupado,” “ausente,” “pagamento exigido,” “proibido”
- ❑ A mídia pode ser enviada sobre RTP ou algum outro protocolo
- ❑ Mensagens SIP podem ser enviadas sobre TCP ou UDP

SIP - mensagens

```
INVITE sip:bob@domain.com SIP/2.0
Via: SIP/2.0/UDP 167.180.112.24
From: sip:alice@hereway.com
To: sip:bob@domain.com
Call-ID: a2e3a@pigeon.hereway.com
Content-Type: application/sdp
Content-Length: 885

c-IN IP4 167.180.112.24
m=audio 38060 RTP/AVP 0
```

SDP - *Session Description Protocol*

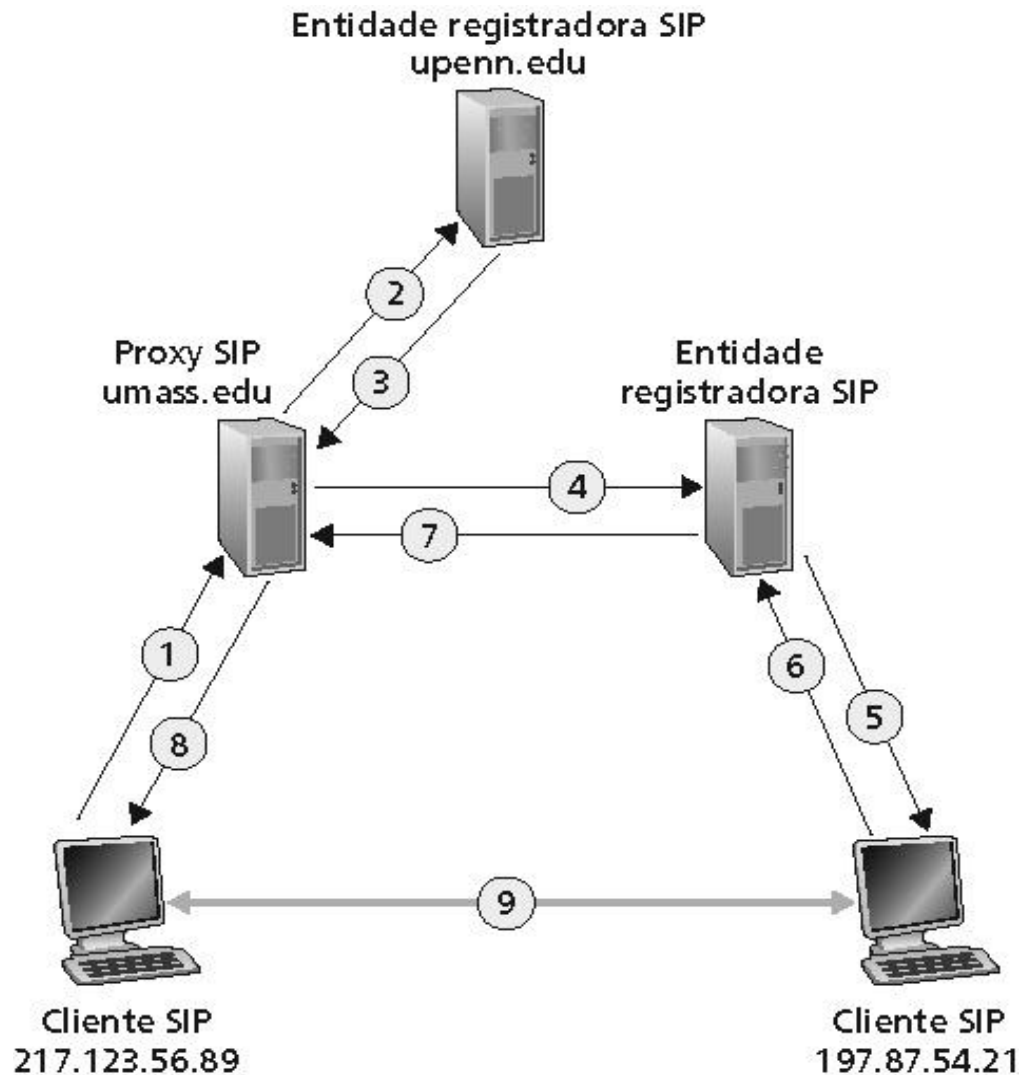
SIP - tradução de nome e localização de usuário

- Cada usuário SIP tem uma entidade registradora
 - Descobrir endereço IP caso esteja sendo utilizado DHCP ou vários IPs
 - Monitora o endereço IP corrente
 - Usuário envia mensagens de tempos em tempos para manter endereço
 - Traduz identificadores de pessoas para endereços IP dinâmicos
- Usuário pode descobrir o endereço IP por meio de uma mensagem `INVITE`
 - Servidores *proxies*

SIP - tradução de nome e localização de usuário

```
REGISTER sip:domain.com SIP/2.0  
Via: SIP/2.0/UDP 193.64.210.89  
From: sip:bob@domain.com  
To: sip:bob@domain.com  
Expires: 3600 (segundos)
```

SIP - tradução de nome e localização de usuário



H.323 x SIP

Item	H.323	SIP
Designed by	ITU	IETF
Compatibility with PSTN	Yes	Largely
Compatibility with Internet	No	Yes
Architecture	Monolithic	Modular
Completeness	Full protocol stack	SIP just handles setup
Parameter negotiation	Yes	Yes
Call signaling	Q.931 over TCP	SIP over TCP or UDP
Message format	Binary	ASCII
Media transport	RTP/RTCP	RTP/RTCP
Multiparty calls	Yes	Yes
Multimedia conferences	Yes	No
Addressing	Host or telephone number	URL
Call termination	Explicit or TCP release	Explicit or timeout
Instant messaging	No	Yes
Encryption	Yes	Yes
Size of standards	1400 pages	250 pages
Implementation	Large and complex	Moderate
Status	Widely deployed	Up and coming

PSTN: Public Switched Telephone Network

Leitura mínima recomendada

Kurose, Ross

Redes de Computadores e a Internet: uma nova abordagem

Cap. 07 – seções 7.1 a 7.5

Tanenbaum

Redes de Computadores, 4a. ed.

Cap. 06 – seção 6.4.3 (RTP e RTCP)

Cap. 07 – seção 7.4.5 (H.323 e SIP)

H.323

<http://www.microsoft.com/windows/NetMeeting/Corp/reskit/Chapter11/default.asp>