

Proposta para Serviços de Monitoração e Controle no GTGV

Daniel da Costa Uchôa
duchoa@larc.usp.br

- Cenário: **Vídeo sob Demanda;**
 - Quatro requisitos básicos:
 1. Disponibilidade do conteúdo;
Qualquer instante;
De qualquer lugar;
Grande número de usuários;
 2. Responder prontamente;
 3. Interação usuário/fluxo;
 4. Continuidade do fluxo;
 - Fatores limitantes:
 - Largura de banda da rede;
 - Necessidade de sincronismo;
 - Taxa de reprodução constante;
 - Largura de banda do servidor;
 - etc.

- Problema: como oferecer tal serviço com **boa qualidade** à um **grande número de usuários**, mantendo recursos de equipamento e rede em limites adequados?
- Uma possível solução: DVoD;
 - Distribui conteúdo de forma eficiente sobre o *backbone* da rede RNP;
 - *proxying and caching*: resolve 1 e 2;
- Como prevenir que ...
 - Pacotes sejam recebidos em uma taxa bastante variável;
 - Fonte gere pacotes em uma taxa muito maior do que a capacidade de processamento do receptor;
- Como otimizar o uso do canal de transmissão;
- Como escalar níveis de qualidade;
- Como garantir ordenação dos pacotes;
- Como detectar perdas;
- Como se adequar às características dinâmicas do tráfego na rede;
- Resumo: **Como fazer a melhor entrega possível?**

- Internet hoje:
 - **Não** há garantias sobre o atraso fim-a-fim e nem sobre a variação deste atraso (*jitter*):
 - **Inadequado para aplicações multimídia;**
 - **Não** há garantia de entrega nem de ordem;
- **Desafio:** Construir mecanismos robustos de entrega de mídia **em tempo real** sobre uma **camada de transporte não confiável**;
- Aplicações multimídia:
 - Devem utilizar técnicas para minimizar estes problemas;
 - Protocolos para transmissão em tempo real;
 - RTP (Real Time Protocol);
 - RTCP (Real Time Control Protocol);
 - RTSP (Real Time Streaming Protocol);

RTP – Real Time Protocol



- Provê funções de **transporte fim-a-fim** adequadas para aplicações em **tempo real**, sobre serviços de rede ***unicast* ou *multicast***:
 - Identificação de conteúdo;
 - Número de sequência;
 - Marca de tempo;
 - Monitoramento de entrega;
- Dividido em duas partes:
 - Transmissão de dados (RTP);
 - Protocolo de controle (RTCP);
- Definido na **RFC 3550**;

Novo Estilo de Protocolo



- Enquadramento à nível de aplicação;
 - É quem melhor conhece seus dados;
 - Implicação: exposição de detalhes de entrega;
 - Aplicação responde apropriadamente caso erros ocorram;
- Processamento de camada integrada:
 - Não é implementado como uma camada separada da aplicação;
 - Parceria aplicação/transporte: cooperam no intuito de alcançar uma entrega confiável;

Novo Estilo de Protocolo

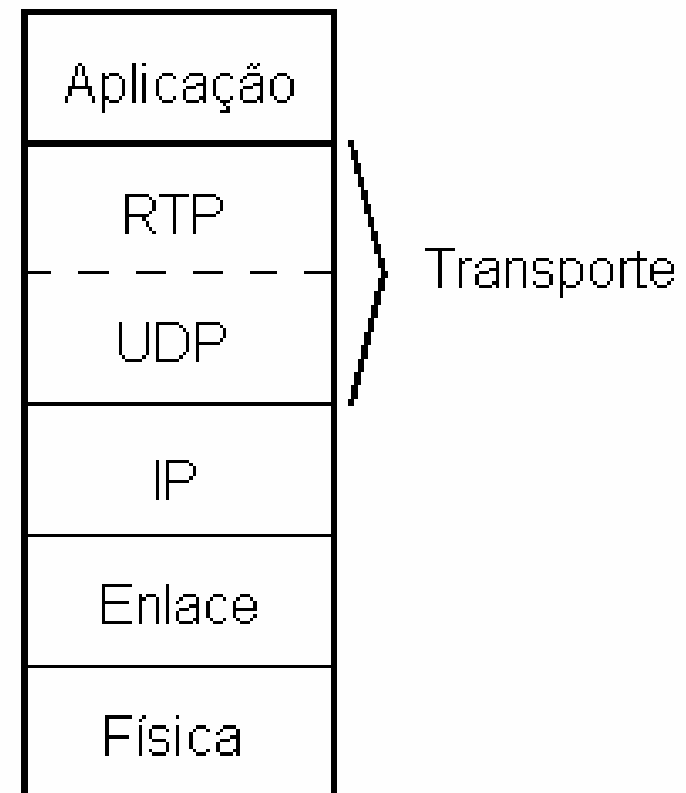


- Não pretende ser um protocolo completo, mas um **framework para construção de protocolos de aplicação**;
 - Sistemas baseados em RTP precisam de outros protocolos para negociar e estabelecer sessões (ex.: SIP – *Session Initiation Protocol*);

Sub-camada de Transporte



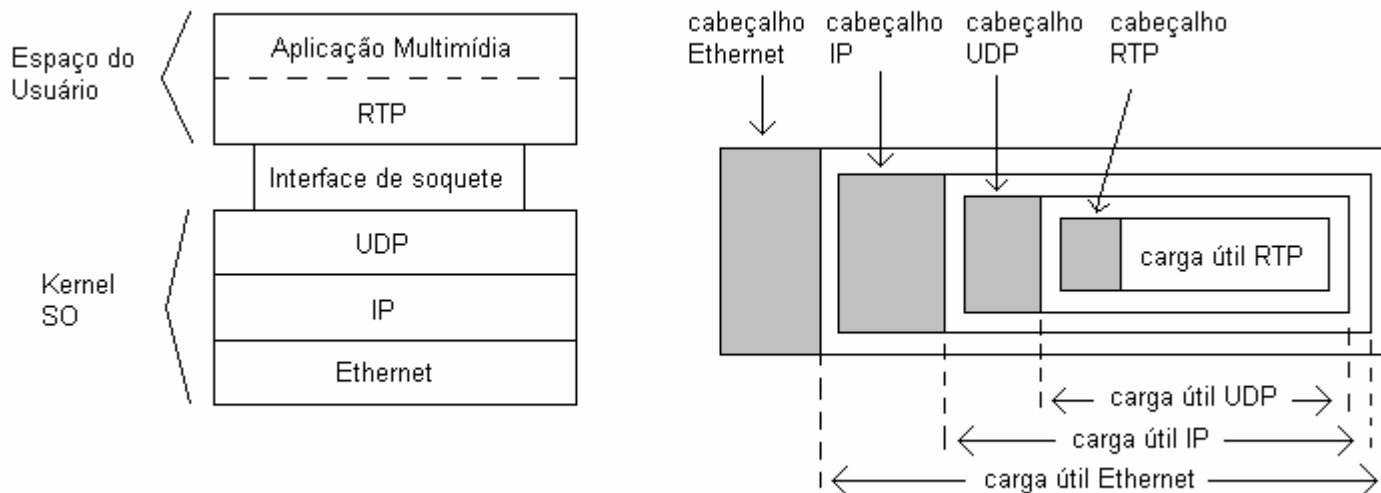
- Posição na pilha de protocolos:
 - Sub-camada da camada de transporte;
 - Implementado na camada de aplicação;
 - Livre escolha do protocolo de transporte;



Encapsulamento RTP/UDP/IP



- Função básica: **Multiplexador**;
 - encapsula diversos fluxos em tempo real sobre um único fluxo de pacotes UDP, através de diferentes redes;
- Geralmente implementado como função de biblioteca no espaço do usuário;



Não faz!



- Expande os serviços da camada de transporte UDP;
 - **Não** garante **entrega temporizada** ou provê qualquer **garantia de QoS** (embora preste serviços de baixo nível para isto);
 - **Não** garante **entrega** ou **ordenação**, não possui **controle de fluxo**, nenhuma **confirmação** e nenhum mecanismo de **retransmissão**;
 - **Não** há **reserva de recursos**;

- Facilita:
 - **Entrega;**
 - **Deteção de falta;**
 - **Segurança;**
 - **Monitoramento;**
 - **Reconstrução;**
 - **Mixagem;**
 - **E sincronização** fluxo de dados multimídia;
- Porém, não o faz propriamente!!!

- É usado para **monitorar a QoS** e levar informação sobre **participantes de uma sessão** formada;
- Opera em **sintonia** com o RTP, sendo definido na mesma do RTP (RFC3550);
- Baseia-se na transmissão periódica de pacotes de controle a todos os participantes da sessão;
- Implementado em 3 partes:
 - Formato de pacotes;
 - Regras de instante de envio (*timing rules*);
 - Base de dados dos participantes;

- Utiliza os mesmos mecanismos de entrega dos pacotes de dados RTP;
- Rede deve permitir **multiplexação** para que se faça **distinção** entre pacotes de **dados e controle**;
- Isto é implementado pelo UDP, por meio da **reserva de portas distintas** para pacotes RTP e RTCP;
 - Se RTP na porta n , RTCP na porta $n+1$;
- Embora RTCP seja considerado como um protocolo executado sobre o RTP, na verdade ele **é apenas um tipo especial de pacote RTP**;

- Fornece *feedback* sobre a qualidade de serviço da transmissão;
 - Usado para manter controle do codificador ativo;
- Identificação permanente de todos os participantes de uma sessão;
- Publica o número de participantes de sessão;
 - Usado para ajustar a taxa de transmissão do RTP;
- Controla a taxa com que os pacotes de controle são enviados de forma a conferir escalabilidade ao protocolo;

Tipos de Pacotes RTCP

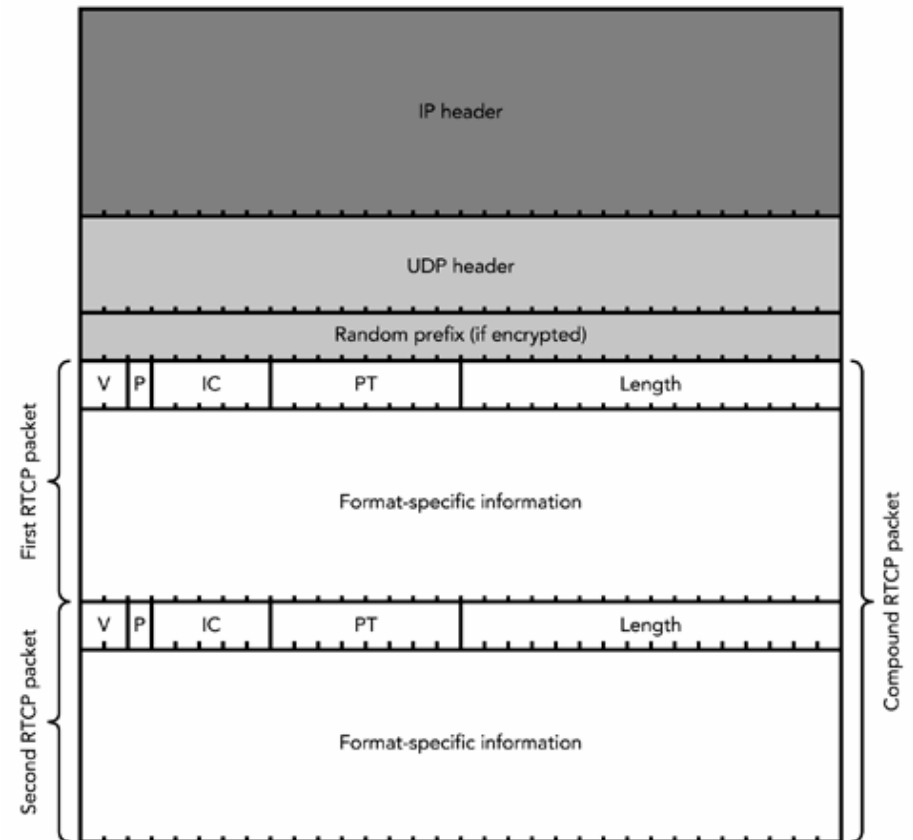


- Cada pacote RTCP possui uma estrutura padrão, comum a todos os pacotes, seguido de elementos estruturados que variam de acordo com o tipo de pacote.
- Cinco possíveis tipos diferentes:
 - **SR (Sender Report – PT=200)**: Estatísticas de transmissão e recepção dos participantes que são transmissores ativos;
 - **RR (Receiver Report – PT=201)**: Estatísticas de recepção dos participantes que não são transmissores ativos;
 - **SDES (Source Description Items – PT=202)**: Descrição de parâmetros da fonte, incluindo o CNAME (*Canonical Name*);
 - **BYE (PT=203)**: Indica fim da participação;
 - **APP (PT=204)**: Funções específicas de uma aplicação

Tipos de Pacotes RTCP



- Pacote RTCP nunca é transportado individualmente, sendo **sempre** agregado em um **pacote composto**;
- São enviados como um único pacote da camada inferior;
- Contém **ao menos um pacote de Report**, para maximizar a resolução das estatísticas;
- **E um pacote de SDES** contendo um *CNAME*, para que novos receptores identifiquem as fontes o mais rápido possível;
- Demais tipos de pacotes podem ou não estar contidos.



Estrutura Comum dos Pacotes RTCP

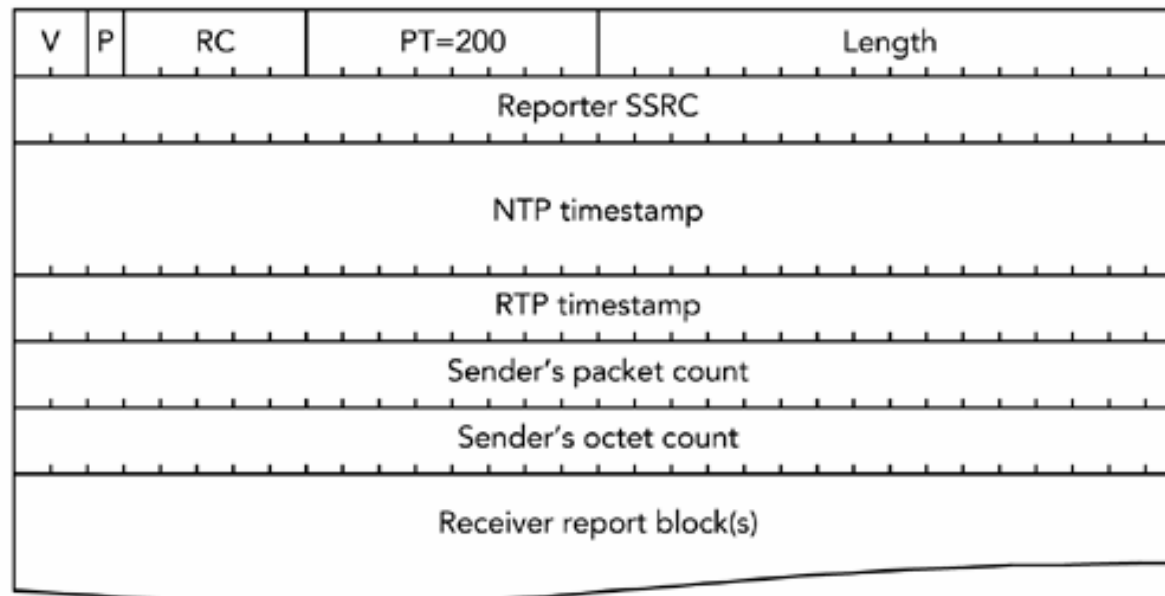


- **Versão (2 bits):** Identifica a versão do RTCP.
- **Padding (1 bit):** *Padding* é usado para **preencher o pacote** RTCP com bytes adicionais, que não fazem parte da informação de controle *payload*. Usado por algoritmos de codificação que exigem pacote com comprimento fixo.
- **Item Count (5 bits):** Contém a quantidade de alguns itens, variando com o tipo de pacote: num pacote SR, trata-se do número de blocos contendo *reports*, enquanto num pacote SDES, contém o número de transmissores especificados no pacote.
- **Packet Type (8 bits):** Contém uma constante que **identifica** os diversos tipos de pacote.
- **Length (16 bits):** Comprimento do pacote, quantificado em palavras de 32 bits, excluindo o cabeçalho e o padding.

SR (Sender Report)



- Enviados por todos os participantes que enviam pacotes RTP;
 - Sessão 1: Informações gerais;
 - Sessão 2: Provê informação sobre mídia enviada, principalmente para que receptores possam sincronizar múltiplos fluxos.
 - Sessão 3: Um *report block* para cada fonte que ele ouve;



SR (Sender Report)



- **Reporter SSRC (32 bits):** Identifica a fonte de sincronismo de quem gerou o pacote.
- **NTP Timestamp (64 bits):** Indica o horário em que o *report* foi gerado; é importante que se use a mesma referência ao gerar o *Receiver Report* para que a análise dos dados e aferição de parâmetros (como *round trip delay, jitter*) seja coerente.
- **RTP Timestamp (32 bits):** Corresponde ao mesmo valor do campo anterior, porém expresso nas mesmas unidades que este mesmo campo do pacote RTP de dados.
- **Sender's Packet Count (32 bits):** Número total de pacotes RTP de dados transmitidos desde o início da transmissão até o instante em que o *report* foi gerado.
- **Sender's Octet Count (32 bits):** Número total de octetos de *payload* (dados), excluindo o cabeçalho e *padding*, que foram enviados pelo transmissor desde o início da sessão até o instante em que o *report* foi gerado.

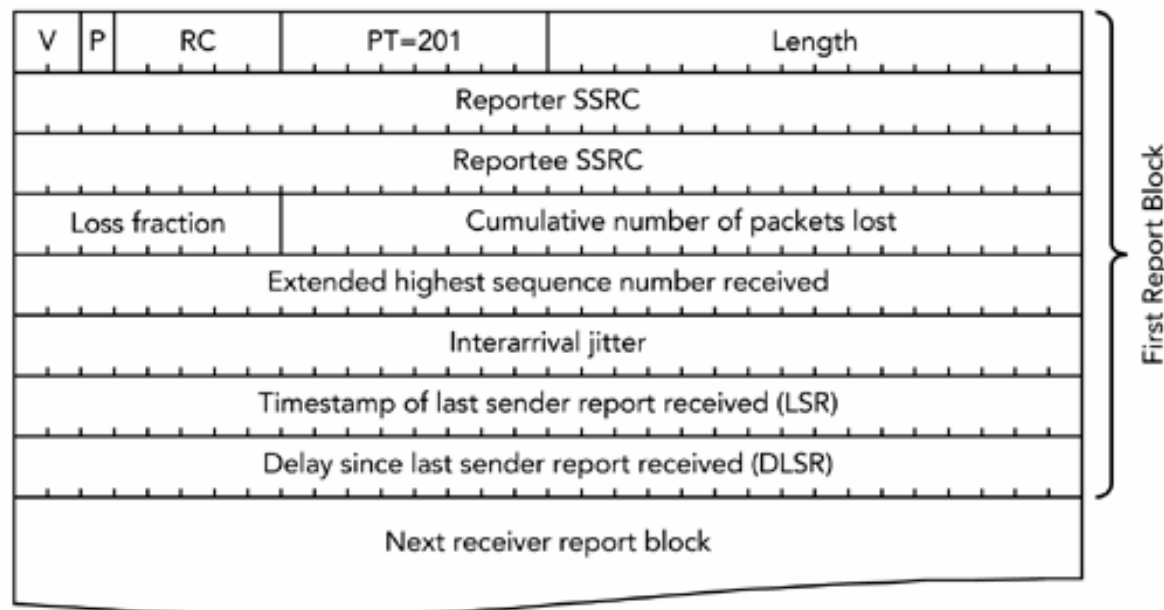
Interpretação dos dados SR

- Uma aplicação pode calcular a **vazão de dados** (*throughput*) aparente para aquele receptor:
 - A partir das informações SR, tem-se a **taxa média de dados da carga útil** (*payload*) e a **taxa média de pacotes**, sobre um intervalo, sem a necessidade de ter recebido os dados;
 - **Tamanho médio da carga útil**: razão dos dois últimos;
 - **Vazão** = número de pacotes recebidos X tamanho médio da carga útil
- Os *timestamps* são usados para gerar uma correspondência entre os *clocks* da mídia e uma referência externa conhecida:
 - Possibilidade de sincronização de lábio (lip synchronization)

RR (Receiver Report)



- Enviados por todos os participantes que recebem dados;
- Usado para monitorar qualidade de recepção;



V = version number
P = padding
RC = number of receiver report blocks
PT = packet type

RR (Receiver Report)



- **Reporter SSRC (32 bits):** Especifica a fonte de sincronismo de quem gerou o pacote.
- **Reportee SSRC (32 bits):** Identifica o receptor a quem as estatísticas de recepção contidas neste pacote se refere.
- **Loss fraction (8 bits):** Fração de pacotes RTP de dados que foram perdidos desde o envio do último *report* (perdidos/esperados).
- **Cumulative number of packets lost (24 bits):** Número total de pacotes relativos à fonte em questão que foram perdidos desde o início da transmissão. Pacotes atrasados não contam como perdidos e pacotes duplicados são contados como recebidos;
- **Extended highest sequence number received (32 bits):** Os 16 bits mais significativos contêm o número de ciclos de seqüência e os últimos 16 bits contêm o mais alto número de seqüência recebido em um pacote de dados RTP proveniente dessa fonte.

RR (Receiver Report)



- **Interarrival jitter (32 bits):** Uma estimativa da variância estatística de chegadas subseqüentes de pacotes RTP de dados, medida em unidades de *timestamp* e expressa em números inteiros. O *interarrival jitter* é definido como o desvio médio da diferença do espaçamento entre pacotes na recepção e no envio, para um determinado par de pacotes.
- **Last sender report received – LSR (32 bits):** Os 32 bits centrais dos 64 bits de campo *NTP Timestamp* recebidos no pacote RTCP SR mais recente.
- **Delay since last sender report received – DLSR (32 bits):** Atraso entre a recepção do último pacote SR da fonte e o envio do pacote RR atual.

Interpretação dos dados RR

- *Feedback* acerca da qualidade da recepção, que pode ser utilizada pelos demais participantes ou por terceiros:
 - Transmissores: adaptar parâmetros da transmissão;
 - Receptores: determinar se problemas detectados têm caráter local, regional ou global;
 - Administradores de rede: avaliar desempenho de suas redes, sem necessidade de recepção dos dados RTP, apenas pacotes RTCP;
- Campos LSR e DLSR: calcular o *round-trip delay* entre receptores;
 - alterar parâmetros de codificação da mídia;
- Dados sobre perda de pacotes (*cumulative counts*): diferença informa se uma perda tem caráter transiente ou de longa duração;
 - influenciar a escolha do formato de mídia ou codificação;
 - evitar congestionamentos e a conseqüente ocorrência de erros;
- Avaliação de *jitter*;

Natureza *peer-to-peer*



- Todos participantes enviam e recebem relatórios a todos os outros participantes (imAlive):
 - **Conhecimento de sessão** sobre todos os outros participantes:
 - Sua presença;
 - Qualidade de recepção;
 - Detalhes pessoais:
 - Nome;
 - Endereço de e-mail;
 - **Localização**;
 - Número de telefone;

Periodicidade RTCP



- Natureza *peer-to-peer*:
 - Grande número de participantes: problema de escalabilidade;
 - *Overhead* gerado pelo RTCP pode ser grande;
- Cada participante pode controlar a periodicidade com que enviam pacotes RTCP;
 - limitar a largura de banda ocupada pelo RTCP;

- **Periodicidade** no envio de pacotes RTCP é **proporcional ao número de participantes na sessão RTP**;
 - Tráfego RTCP não deve exceder 5% do tráfego RTP;
 - 25% dessa reserva é destinada aos transmissores;
 - Fornecem informações essenciais à tomada de decisão por parte dos receptores;
 - Sessões pequenas: um relatório RTCP a cada cinco segundos;
 - Taxa de envio aleatória por um fator de 0,5 a 1,5: evita sincronização indesejada entre relatórios;

Sincronização



- Diferentes fluxos podem utilizar *clocks* distintos, com granularidades e taxas de *jitter* diferentes;
- Terceiros devem controlar fluxos sem necessidade de recepção dos dados RTP;
- O RTCP pode ser utilizado para manter esses elementos sincronizados;