Z24: RTP, A/V Synchronization, Application-level adaptation

Mark Handley

UDP

- Connectionless, unreliable, unordered, datagram service
- No error control
- No flow control
- No congestion control
- Port numbers
- Must be used for real-time data:
  - TCP automatic congestion control and flow control behaviour is unsuitable

0 8 16 24 31

0–15	16–31
source port	destination port
length	checksum
data	

Multimedia over UDP

UDP provides demultiplexing and a checksum.

Realtime media also requires:

- *timestamps* for timing recovery and audio/video synchronization
- *sequence numbers* for loss detection
- *payload type indication* to allow codec switching
- *speaker identification* in multi-way sessions.

Solution: **Realtime Transport Protocol** (RTP)

- RTP is usually carried over UDP.

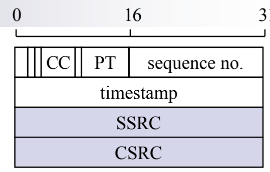
RTP: Realtime Transport Protocol

- RTP data packet header:

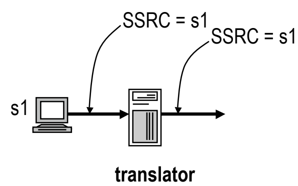
0				8		16	24
V=2	P	X	CC	M	PT	sequence number	
timestamp							
synchronization source (SSRC) identifier							
contributing source (CSRC) identifiers							

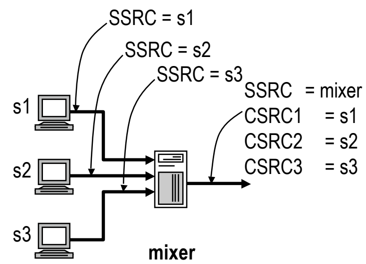
- RTP is a common packet header for realtime data, a set of conventions for how to use that header, and a companion "control" protocol RTCP.

RTP: Details

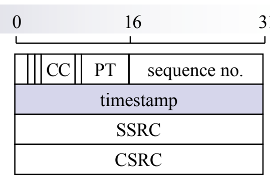


- SSRC identifies a unique RTP sender.
- The packet source does not identify the sender, which means users can move, and transcoders can be employed.
- If streams are mixed, CSRC identifies the source that were mixed together.



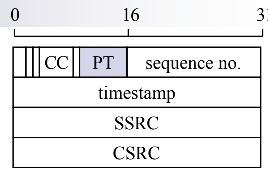


RTP: Details



- The RTP timestamp is in units which are meaningful for the media stream itself.
  - Eg, 8KHz sampled u-law audio would use an 8KHz sample-driven clock for timestamps.
  - MPEG video would use the MPEG 90KHz clock.
- This avoids having to deal with rounding when reconstructing the original timing within an audio stream.
- However, it means the clocks for audio and video streams from the same source use different timebases.
  - RTCP relates these different clocks to realtime so A/V synchronization can be performed.

RTP: Details



- The payload type field identifies the codec being used and the form of packetization.
  - Originally RTP allocated static payload types for codecs, but the space is only a 7-bit space and there are too many codecs!
  - Now, the field is mostly used with dynamic payload types. The field identifies the different codecs in a session, and an out-of-band mechanism (typically SDP) relates the PT field values to the actual codecs.
- The IANA MIME-types registry is used to register new codec names and their RTP packetization formats.

RTCP: Realtime Transport Control Protocol

Each RTP session is accompanied by an RTCP control session.

- RTCP SDES messages are sent by all session members.
  - The indicate who is in the session.
- RTCP SR messages are sent be all data senders
  - They provide the relationship between media timestamps and realtime.
- RTCP RR messages are sent by all receivers.
  - They provide feedback to the senders on reception quality.

- These messages can be combined into compound packets.
- Their rate depends on the membership
  - The more members, the less often each reports so that the control traffic is kept to a small fraction of the data bandwidth.

RTCP SDES packet

- **S**ource **DES**cription: all ASCII strings
- Information types from RFC1889:
  - CNAME: canonical identifier (mandatory)
  - NAME: name of user
  - EMAIL: address user
  - PHONE: number for user
  - LOC: location of user, application specific
  - TOOL: name of application/tool
  - NOTE: transient messages from user
  - PRIV: application-specific/experimental use

RTCP Sender Reports

- An RTCP Sender Report (SR) Message:

0			8	16	24
V=2	P	RC	PT=SR=200	length	
SSRC of RTCP packet sender					
NTP timestamp, most significnant word					
NTP timestamp, least significant word					
RTP timestamp					
sender's packet count					
sender's octet count					
receiver report blocks					

RTCP Receiver Reports

- An RTCP Receiver Report (RR) block:

0 8 16 24

SSRC of sender being reported	
fraction lost	cumulative number of packets lost
extended highest sequence number received	
interarrival jitter	
last SR	
delay since last SR	

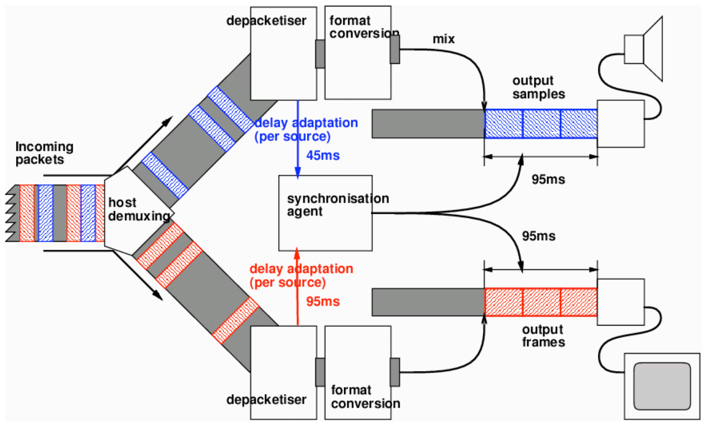
- RTCP RR's give:
  - loss feedback
  - jitter feedback
  - potential RTT feedback

RTCP Receiver Reports

- Receiver reports in RTCP are a little controversial
  - In small sessions they definitely provide useful feedback that helps problem diagnosis.
  - In large sessions they provide a statistical sampling of the network conditions. I believe this is very useful, but not everyone agrees.
  - They can be problematic when satellites with poor reverse path properties are used for multicast.
- In addition, it's been proposed the information be used for congestion control.
  - I don't believe RR's are too useful for this.

Inter-stream Synchronization

Once media timestamps are related to realtime, inter-stream synchronization can be performed by adjusting the playout delays.

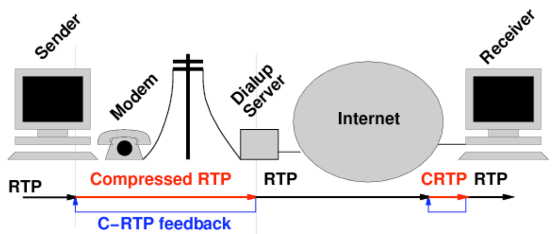


RTP Overhead

- The IP, UDP and RTP headers between them comprise 40 bytes.
- GSM audio at 14Kbps packetized in 80ms packets occupies 140 bytes per packet. Thus the headers add an extra ~30%.
  - There are lower bandwidth codecs than GSM for which this is even worse.
  - Also for better interactivity, perhaps 40ms packets would be desirable with some codecs?
- *On low bandwidth links, perhaps RTP is too expensive?*

IP/UDP/RTP Header Compression

- Most of these low bandwidth links are at the edges and are not highly multiplexed. Thus the same techniques that work for TCP header compression can be applied.



- Senders and receivers need the full IP/UDP/RTP data as do routers where there's a lot of multiplexing.
- On the slow dialup links, a predictor removes predictable fields and re-inserts them at the other end.

IP/UDP/RTP Header Compression

- RTP header compression compresses the IP, UDP and RTP headers down to a few bytes.
  - This possible because most of the headers either do not change between one packet and the next, or change in a predictable manner.
  - The link-sender removes the predictable state, and the link-receiver holds per-flow state and adds it back.
- Note: a gateway cannot tell whether UDP is carrying RTP
  - The scheme is designed to compress IP and UDP if RTP is not being used, and compress the RTP header too if it turns out to be predictable.

Application-level adaptation

Data rate

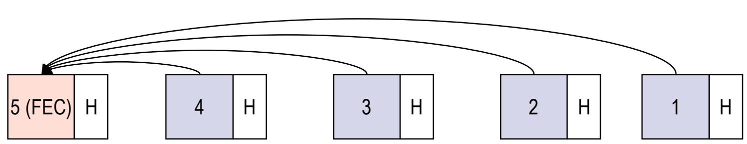
- Application specific changes to the media stream:
  - changes in encoding, use different codecs
  - reduce sampling rate
  - use compression schemes and other “clever” encodings
- Example – audio:
  - change sampling rate, audio bandwidth, encoding scheme
- Example – video:
  - change picture size, frames per second, colour depth

Error control and loss control

- Receiver-only techniques:
  - at receiver - do nothing!
  - media-specific “fill-in” (e.g. interpolation)
- Transmitter assisted techniques:
  - FEC
  - redundant encoding
  - (re-transmission possible in low-delay environments)

Error control and loss control [1]

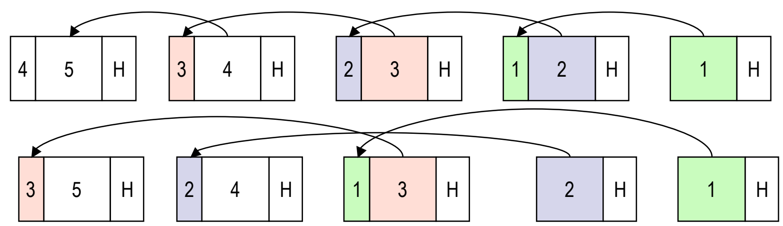
- Use packet-level forward error correction (FEC):



- Simplest FEC: packet 5 is the XOR of packets 1 to 4
  - Eg: if packet 3 is lost, it can be reconstructed from 1 2 4 5
- More powerful erasure codes can create n parity packets from m original data packets
  - Need to receive any n out of the $n+m$ packets sent.

Error control and loss control [2]

- Use example redundant encoding to cope with loss:



Redundant encoding is a “lower quality” (lower bit rate) version of the original packet

References

RFC2198, *RTP Payload for Redundant Audio Data,* C. Perkins, I. Kouvelas, O. Hodson, V. Hardman, M. Handley, J.C. Bolot, A. Vega-Garcia, S. Fosse-Parisis

RFC2508, *Compressing IP/UDP/RTP Headers for Low-Speed Serial Links,* S. Casner, V. Jacobson

RFC2733, *An RTP Payload Format for Generic Forward Error Correction*, J. Rosenberg, H. Schulzrinne

RFC2736, *Guidelines for Writers of RTP Payload Format Specifications,* M. Handley, C. Perkins.

RFC3242, *RObust Header Compression (ROHC): A Link-Layer Assisted Profile for IP/UDP/RTP*, L-E. Jonsson, G. Pelletier

RFC3550, *RTP: A Transport Protocol for Real-Time Applications,* H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson

RFC3551, *RTP Profile for Audio and Video Conferences with Minimal Control,* H. Schulzrinne, S. Casner.