EDA095
Audio and Video Streaming

Pierre Nugues

Lund University
http://cs.lth.se/pierre_nugues/

May 8, 2014

What is Streaming

Streaming corresponds to playing audio and video files from an Internet server.
This opposes to downloading the corresponding files.
As transmission over the Internet is not synchronous, streaming uses a buffer to store a part of the data.
This buffer dampens irregularities in the Internet transmission.
Streaming imposes constraints on the network speed:
Download speed on the average should be at least as fast as playback speed.

Applications of Streaming

Applications are numerous. In addition to data:

- Internet telephony and video conferences
- Digital radios and TV: ordinary broadcast but through Internet, no frontier, no distance
- Audio and video server: on demand movies and concerts.
- Games and virtual reality
- Interaction

Triple play: data (IP), audio (VoIP), video (TVoIP).
It is made possible because of the growing availability of ADSL and fast Internet
Companies in Sweden: NetInsight, PacketFront, Kreatel (now Motorola), Marratech (now Google), etc.

Problems with TCP/UDP

The original TCP/UDP protocols are based on packet transmission and have no quality of service.

Multimedia transmission has to tackle:

Delay. Must be less than 300 ms. (Perception threshold: 150 ms)

Jitter. Packets may use different transmission paths that results into time expansion and compression

Loss. Routers may drop packets when the network load is too high

Problems with TCP/UDP

Audio and video transmission needs buffering and synchronization, possibly error correction, for instance by repeating data

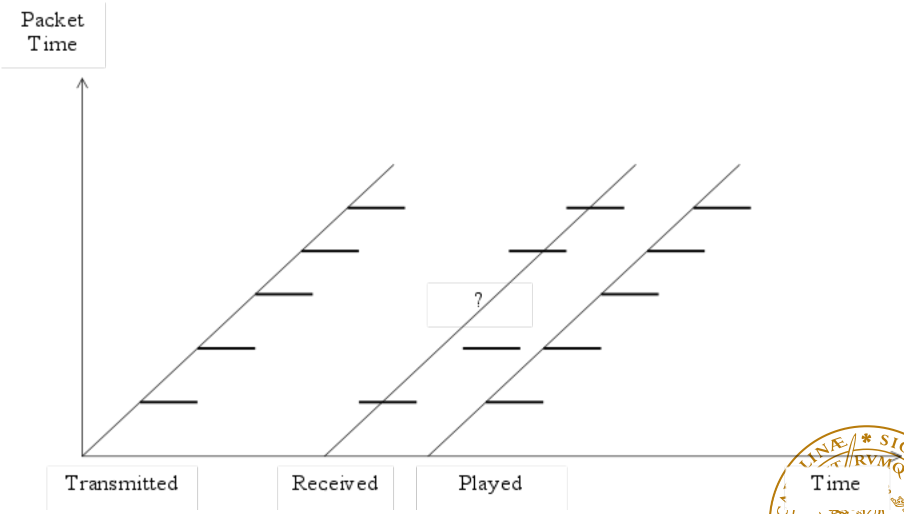
UDP is just a layer to address ports. It is compatible with the requirements.

However, there is no congestion control

The datagram congestion control protocol (DCCP) attempts to fill the gap, but is not widely adopted. See `http://tools.ietf.org/html/rfc4340`

TCP is aslo widely used in commercial video streaming, see `http://lass.cs.umass.edu/papers/pdf/TR03-TCP.pdf`

Packet Transmission



Real Time Transport Protocol

The Real Time Transport Protocol (RTP)

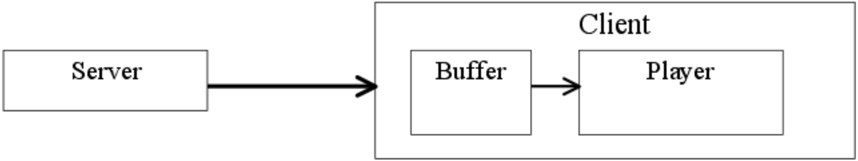
- Identifies the content
- Adds time stamps
- Adds sequence numbers

RTP is encapsulated inside UDP packets
RTP can be used with unicast and multicast transmission
RTP does not guarantee a real-time delivery.
RTP needs an application layer to

- Re-order packets
- Attenuate jitter
- Compensate packet loss

DASH, dynamic streaming over HTTP, is a modern alternative to RTP that uses HTTP instead of UDP.

Architecture



Real Time Transport Protocol (RTP)

RTP is on top of UDP. It uses even ports and port + 1 is for RTCP

Ethernet	IP	UDP	RTP Media content: MPEG, AIFF, and son on

The simplified RTP header structure is:

Payload type	Sequence Number	Timestamp	Sync. Source ID (SSRC)	Other fields

RTP RFCs are available here: `http://www.ietf.org/rfc/rfc3550.txt`

and `http://www.ietf.org/rfc/rfc3551.txt`
(Or through RFC Editor `http://www.rfc-editor.org/`)
Other reference: `http://csperkins.org/standards/rtp-book.html`
and `http://www.networksorcery.com/enp/protocol/rtp.htm`

RTP Header

- Version (2 bits)
- Padding (1 bit)
- Extension (1 bit)
- CSRC count (4 bits)
- Marker (1 bit)
- Payload type (7 bits) corresponds to the packet content: PCM = 0, DVI4=5, JPEG = 26, MPEG-2 = 33 (`http://www.iana.org/assignments/rtp-parameters`)
- Sequence number (16 bits) is incremented each time a packet is sent (Nothing guarantees the arrival order with UDP)
- The timestamp (32 bits) corresponds to the sampling instant of the first octet in the RTP data packet. (Clock of the sending machine)
- SSRC (32 bits) is the source of the stream. (A sending machine can have multiple sessions.)

Timestamp According to the RFC

The timestamp reflects the sampling instant of the first octet in the RTP data packet. The sampling instant must be derived from a clock that increments monotonically and linearly in time to allow synchronization and jitter calculations (see Section 6.3.1). The resolution of the clock must be sufficient for the desired synchronization accuracy and for measuring packet arrival jitter (one tick per video frame is typically not sufficient). The clock frequency is dependent on the format of data carried as payload [. . .]

As an example, for fixed-rate audio the timestamp clock would likely increment by one for each sampling period. If an audio application reads blocks covering 160 sampling periods from the input device, the timestamp would be increased by 160 for each such block, regardless of whether the block is transmitted in a packet or dropped as silent.

8 kHz $\rightarrow$ 125 μs, 160 samples $\rightarrow$ 20 ms

Real-Time Control Protocol (RTCP)

The real-time control protocol (RTCP) is part of the RTP protocol and defined in the same RFC.

It sends periodically control packets to all participants in the session and uses a different port, $N + 1$

It provides feedback on the quality of the data from the sender and the receiver: Statistics on packets sent, received, lost, jitter

Should be limited to 5% of the bandwidth.

Commands	Description
SR	Sender report
RR	Receiver report
SDES	Source description
BYE	Quit

Encoding Formats

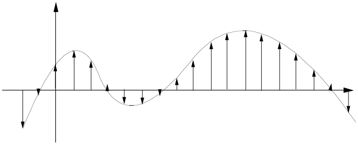
Telephone:	8 kHz, 1 octet (256 values), 64 kbit/s
CD:	44.1 kHz, 16 bits, stereo, 1.4 mbits/s
MP3,	compressed, 96, 128, 160 kbit/s
G.732,	(Internet telephony in H.323) 5.3 kbit/s or 6.4 kbit/s
MPEG-2	used in DVD, 3-6 mbit/s

Two important concepts in encoding methods:

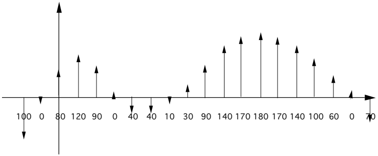
- Bit rate can be constant (CBR) or variable (VBR)
- From an original format, encoding compress data with or without loss. Lossy compression generally results in better rates but lower quality: sometimes not perceptible. Loss rate can be a parameter of the encoding method.

Pulse Code Modulation

Sampling



Digitization



Digitization can be linear or logarithmic: μ-law, A-law

JPEG

Compression standard for still pictures:

- Maps RGB images onto YUV coordinates (luminance and chrominance)
- Applies a discrete cosine transform (DCT)
- Quantizes, which results in a compression with an adjustable loss
- Run-length encoding

MPEG

Initially sequences of frames using JPEG (I Frames)

25 or 30 frames/s

Uses temporal redundancies between images: differences between frames (P and B Frames)

MPEG 2 has multiple possible resolutions: 720×480, 720×576, $1920 \times 1080\ldots$

Multimedia streams contain audio and video data that are synchronized in MPEG

(`http://en.wikipedia.org/wiki/MPEG`)

Codecs

Codecs encode and decode original data streams.

Depending on the media you are sending, you must have the corresponding codec.

Formats supported by the RTP implementation of the Java Media Framework: http://www.oracle.com/technetwork/java/javase/formats-138492.html#RTPFormats

Codecs can be found from many sources as: http://jffmpeg.sourceforge.net/

Real-Time Streaming Protocol (RTSP)

RTSP is a HTTP-like protocol to control streaming media.
It acts as a sort of remote control.
(`http://www.ietf.org/rfc/rfc2326.txt`)

Commands	Description
SETUP	Causes the server to allocate resources for a stream and start an RTSP session
PLAY	Tells the server to start sending data
RECORD	Records data
PAUSE	Temporarily halts a stream without freeing server resources
TEARDOWN	Frees resources associated with the stream. The RTSP session ceases to exist on the server

RTSP uses 544 as dedicated port.
RTSP servers typically use RTP for the audio/video transmission.

C→S SETUP rtsp://example.com/foo/bar/baz.rm RTSP/1.0
CSeq: 1
Transport: RTP/AVP;unicast;client_port=4588-4589

S→C RTSP/1.0 200 OK
CSeq: 1
Date: 23 Jan 1997 15:35:06 GMT
Session: 12345678
Transport: RTP/AVP;unicast;client_port=4588-4589;server_port=62

C→S PLAY rtsp://audio.example.com/audio RTSP/1.0
CSeq: 2
Session: 12345678
Range: npt=10-15 *npt: normal play time*

...

C→S PAUSE rtsp://example.com/audio RTSP/1.0
CSeq: 3
Session: 12345678

S→C RTSP/1.0 200 OK
CSeq: 3

RTSP State Machine (Client)

State	Message sent	Next state after response
Init	SETUP	Ready
	TEARDOWN	Init
Ready	PLAY	Playing
	RECORD	Recording
	TEARDOWN	Init
	SETUP	Ready
Playing	PAUSE	Ready
	TEARDOWN	Init
	PLAY	Playing
	SETUP	Playing (changed transport)
Recording	PAUSE	Ready
	TEARDOWN	Init
	RECORD	Recording
	SETUP	Recording (changed transport)

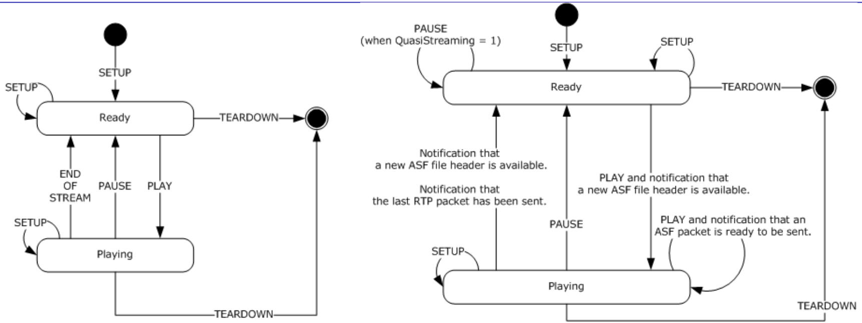
RTSP State Machine (Server)

State	Message received	Next state
Init	SETUP	Ready
	TEARDOWN	Init
Ready	PLAY	Playing
	SETUP	Ready
	TEARDOWN	Init
	RECORD	Recording
Playing	PLAY	Playing
	PAUSE	Ready
	TEARDOWN	Init
	SETUP	Playing
Recording	RECORD	Recording
	PAUSE	Ready
	TEARDOWN	Init
	SETUP	Recording

RTSP State Machines

RTSP state machines for the client and servers

Client	Server



From Microsoft MSDN documentation, http://msdn.microsoft.com/en-us/library/cc245238%28v=prot.10%29.aspx)

Session Initiation Protocol (SIP)

SIP is a protocol to establish a session with a remote host in UDP or RTP.

Defined by IETF in RFC: `http://www.ietf.org/rfc/rfc3261.txt`

SIP enables to set up a call, negotiate the parameters, manage, and close the session.

Borrows many ideas from HTTP and uses UDP or TCP.

Once the session is established on port 5060, the media transmission can use RTP or something else.

SIP is similar to RTSP.

A SIP Example (From the RFC)

Alice from Atlanta.com sends an INVITE request addressed to Bob's SIP URI at Biloxi.com.

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>;
Content-Type: application/sdp
Content-Length: 142
```

Fields

Via contains the address (pc33.atlanta.com) at which Alice is expecting to receive responses to this request

To contains a SIP URI (sip:bob@biloxi.com) towards which the request was originally directed.

From also contains a SIP URI (sip:alice@atlanta.com) that indicates the originator of the request.

Call-ID contains a globally unique identifier for this call.

CSeq or Command Sequence contains an integer, incremented for each new request within a dialogue

Contact contains a SIP URI that represents a direct route to contact Alice. While Via tells where to send the response, Contact tells where to send future requests.

A Content Example

The session description protocol, SDP, specifies details of the connection using name-value pairs. (http://www.ietf.org/rfc/rfc4566.txt)

```
v(ersion)=0
o(wner)=bell 53655765 2353687637 IN IP4 128.3.4.5
c(onnection)=IN IP4 135.180.144.94
m(edia)=audio 3456 RTP/AVP 0 3 4 5
```

A SIP Example (From the RFC)

Alice	Proxy	Proxy Bob
INVITE F1 →		
	INVITE F2 →	
← 100 TRYING F3		INVITE F4 →
	← 100 TRYING F5	
		← 180 RINGING F6
	←180 RINGING F7	←200 OK F9
←180 RINGING F8	←200 OK F10	
←200 OK F11		
	ACK F12 →	
	Media	
	← BYE F13	
	OK F14 200 →	

SIP Methods

Methods	Descriptions
INVITE	Invites a session
ACK	Acknowledges
OPTIONS	Server capabilities
BYE	Closes a session
CANCEL	Cancels a pending request
REGISTER	

SIP Registrar

When the SIP client starts, it registers its location.
Proxies can find people in different places using multiple devices.

```
REGISTER sip:registrar.biloxi.com SIP/2.0
Via: SIP/2.0/UDP bobspc.biloxi.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Bob <sip:bob@biloxi.com>;tag=456248
Call-ID: 843817637684230@998sdasdh09
CSeq: 1826 REGISTER
Contact: <sip:bob@192.0.2.4>
Expires: 7200
Content-Length: 0
```

SIP Registrar (II)

C→S:

```
REGISTER sip:bell-tel.com SIP/2.0
Via: SIP/2.0/UDP pluto.bell-tel.com
To: sip:watson@bell-tel.com
From: sip:jon.diligent@bell-tel.com
Call-ID: 17320@pluto.bell-tel.com
CSeq: 1 REGISTER
Contact: sip:tawatson@example.com
```

H.323

H.323 is a competitor to SIP.

It has been promoted by the ITU – the telephone companies

Complete and in the beginning more complex then SIP

Good integration with telephone systems

RTSP and SIP

From http://www.cs.columbia.edu/~hgs/rtsp/faq.html

RTSP and the Session Initiation Protocol (SIP) share many common characteristics.

RTSP is designed to control the media stream during delivery; SIP is not directly involved in controlling media streams.

Property	SIP	RTSP
Task	Inviting users to real-time conferences	Initiating and controlling media streams to unicast and multicast addresses
Data transport	Bi-directional between SIP caller and callee	One-directional; media server may either play or record data, with direction indicated at stream setup time
third-party delivery	not yet, but planned	The Transport header may contain any address, including an address differing from the one issuing the RTSP requests.
Caching	No notion of content caching, as conferences are real-time	Caching similar to HTTP, where end systems contact cache to obtain content. Like some HTTP caches such as

RTSP and SIP

Property	SIP	RTSP
Redirection	Location header; used for personal mobility and for bypassing proxies	Location header; used for load sharing between media servers
Session identification	Call-ID	Session
Session setup	INVITE Invites a user to one or more media sessions. Transport information is indicated in the session description included as the message body.	SETUP Invites a server to send data for a single media stream to the destination specified in the Transport header field. If left open by the client, the server may also select transport parameters and convey them to the client using the Transport response header.
Session teardown	BYE Terminates the whole call/session.	TEARDOWN Depending on URL, may terminate whole session or individual media stream.