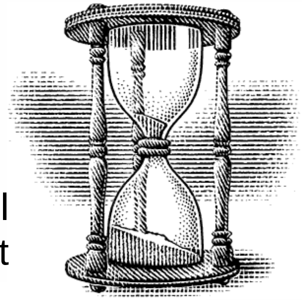
Digital Media Development
- Media Streaming -

Prof. Dr. Andreas Schrader
ISNM International School of New Media
University of Lübeck
Willy-Brandt-Allee 31a
23554 Lübeck
Germany
Schrader@isnm.de

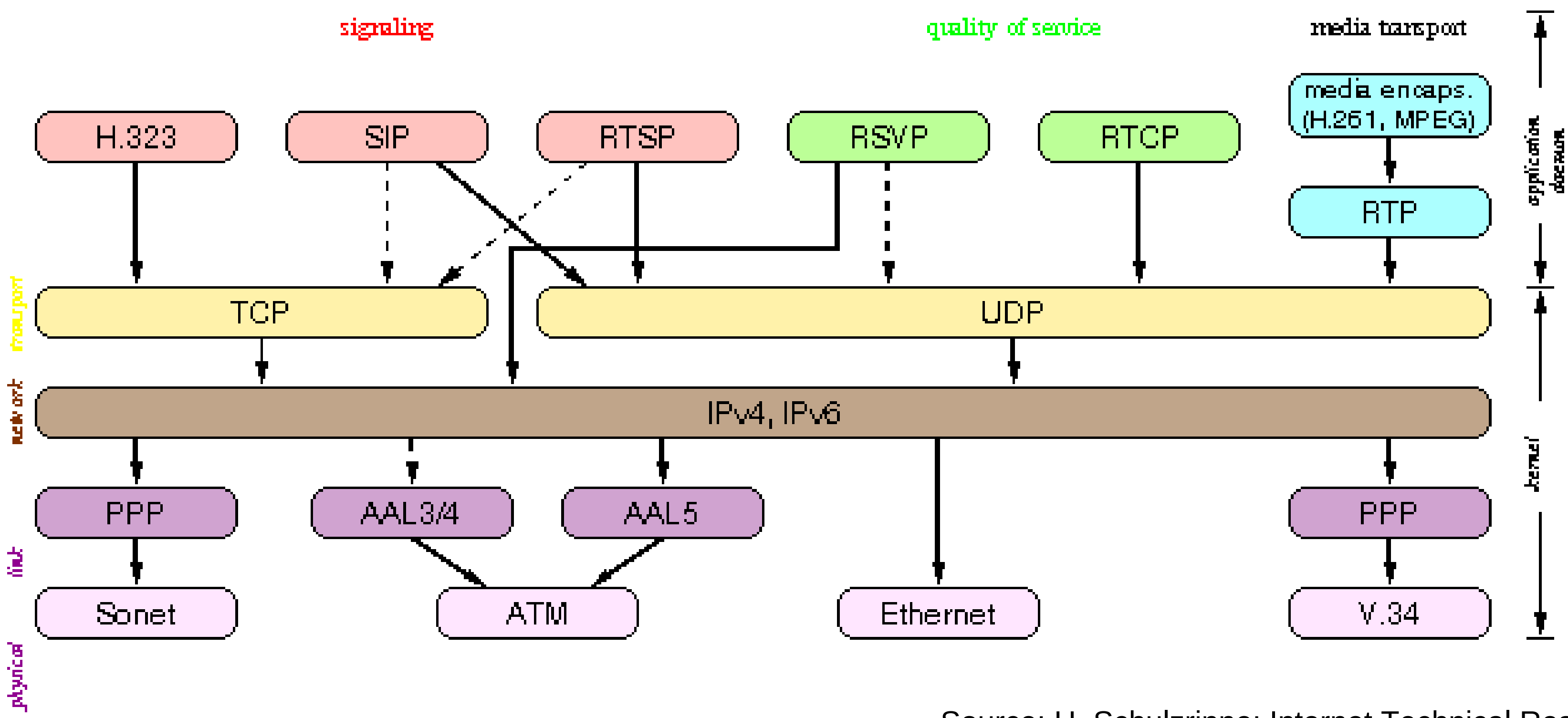
Realtime Transmission Protocols

Transmission Protocols

- Internet Protocols Overview



Hourglass model
of the Internet



Source: H. Schulzrinne: Internet Technical Resources

Transmission Protocols

- UDP – User Datagram Protocol
  - Defined in IETF RFC 768
  - ***Best-Effort*** datagram service (packet sequence):
    - messages may get ***lost***, ***duplicated*** or ***re-ordered***
    - messages may arrive faster than receiver can handle them (no flow control)
  - ***Connection-less*** (each packet goes it‘s own way)
  - ***Low complexity***, ***low overhead*** (in principle IP + small header of 8 Bytes)
  - Low share at all Internet traffic

    (but this is expected to increase due to realtime services)

Transmission Protocols

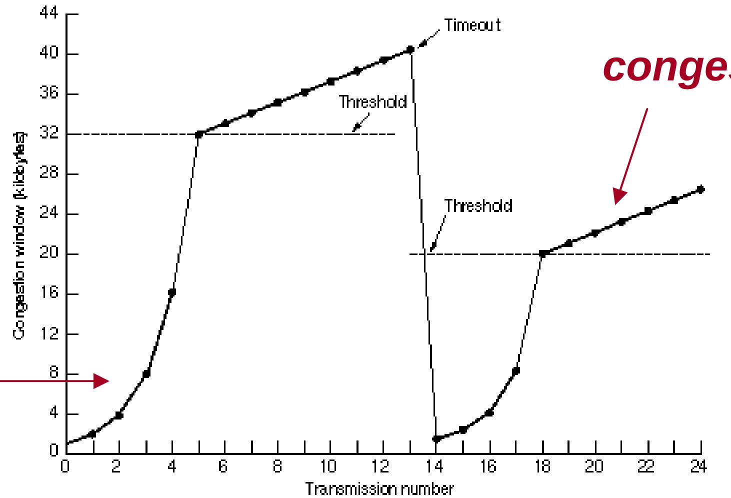
- TCP – Transmission Control Protocol
  - Defined in IETF RFC 793, Extensions and bug fixes in RFC 1122 and RFC 1323
  - ***Reliable*** stream delivery (byte sequence):
    - no messages get lost, duplicated, or re-ordered (handshaking, acknowledgements)
    - ***flow control*** mechanisms (sliding window)
  - ***Connection-oriented***
  - Full duplex communication
  - ***Complex***, ***some overhead*** (Header >= 20 Bytes)
  - High share at all Internet traffic (in particular due to WWW/HTTP)

Transmission Protocols

- TCP – Transmission Control Protocol
  - At start-up and after period of congestion: ***slow start*** recovery: increase congestionWindow exponentially each ACK
  - At half of original (pre-congestion) size: use ***congestion avoidance***: increase congestionWindow linearly for eack ACK
  - Flow Control: upon loss, reduce congestionWindow by half (exponential backoff, minimum is 1 segment)

congestion avoidance

slow start



Source: Tanenbaum: Computer Networks, 4th Edition, Pearson Education Inc., 2003, p. 550

Transmission Protocols

- TCP Extensions

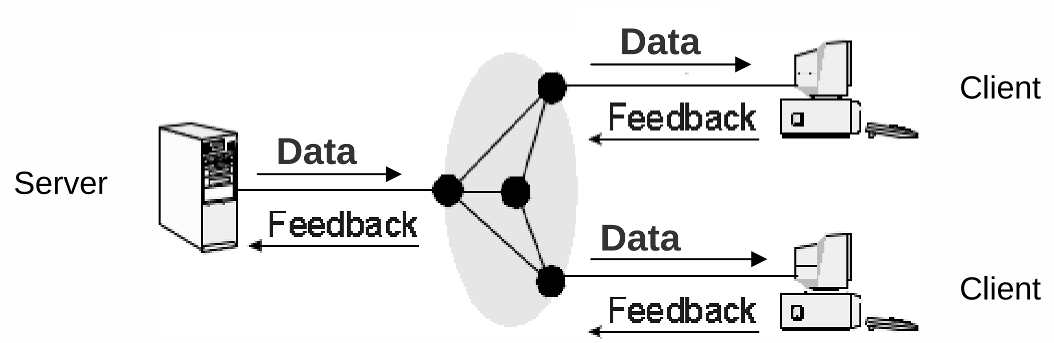
  - ***High Performance TCP***
    - Lines with high bandwidth (fat pipes), high delays or both
    - The capacity of the line is much higher than the max. window size of 65KByte
    - It takes too much time to wait for the ACK and the line is wasted

  - ***Wireless TCP***
    - TCP is optimized for wired links (loss only due to congestion)
    - In wireless links loss occurs without congestion
    - Sliding window mechanism: worse the problem (further reduces the datarate – better would be too try even harder!)
    - Possible solution: intermediate ACK‘s and retransmission (indirect TCP), but: violates the semantics of TCP
    - Alternative: automatic retransmission in wireless base stations

  - ***Header Compression***
    - Useful for slow links
    - Reduces combined 40 Byte TCP/IP header to three Byte

Transmission Protocols

- Real-time transport
  - Adaptive applications can change their behaviour dynamically according to the network transmission characteristics



  - The receiver needs to reply quality feedback information
  - This can be supported with real-time transport protocols
    - ***RTP/RTCP*** – Realtime Transport/Control Protocol
    - ***RTSP*** – Realtime Streaming Protocol

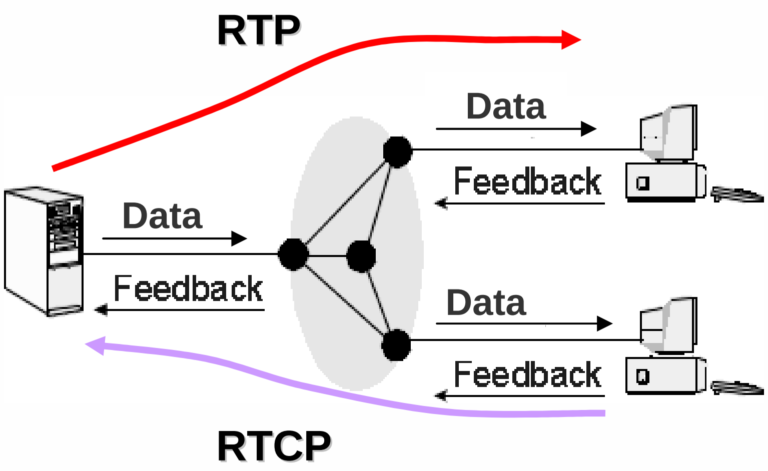
Transmission Protocols

- ***RTP – Realtime Transport Protocol***
  - Defined by IETF Audio/Video Transport Working Group
    - RTPv1 IETF RFC1889 (January 1996)
    - RTPv2 IETF RFC 3550 (July 2003) (Obsoletes RFC1889)
  - Support for real-time transmission across the Internet (audio and video)
  - Data part (RTP) and control part (RTCP)
  - RTP provides framing support, but no guarantees!
  - RTCP provides additional control functions, but no guarantees either!
  - Supported by
    - Netscape (Navigator), Microsoft (ActiveX, Netmeeting)
    - ITU (H.323) and many others

www.ietf.org

Transmission Protocols

- ***RTP – Realtime Transport Protocol***
  - Used for on-demand and interactive services
  - Determine sender and receivers
  - Determine data encoding
  - Synchronisation of data-streams
  - Error detection



Transmission Protocols

- RTP is adding an ***RTP header*** in front of the payload

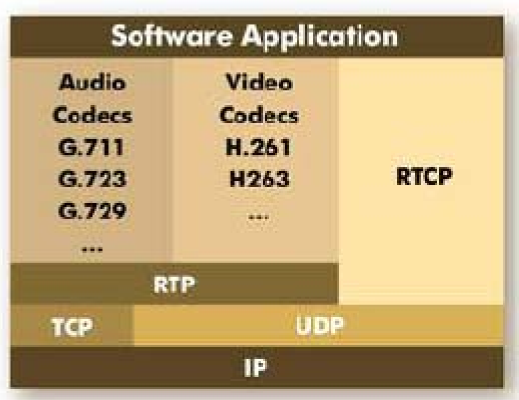
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|V=2|P|X|  CC   |M|     PT      |       sequence number         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           timestamp                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           synchronization source (SSRC) identifier            |
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
|            contributing source (CSRC) identifiers             |
|                             ....                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

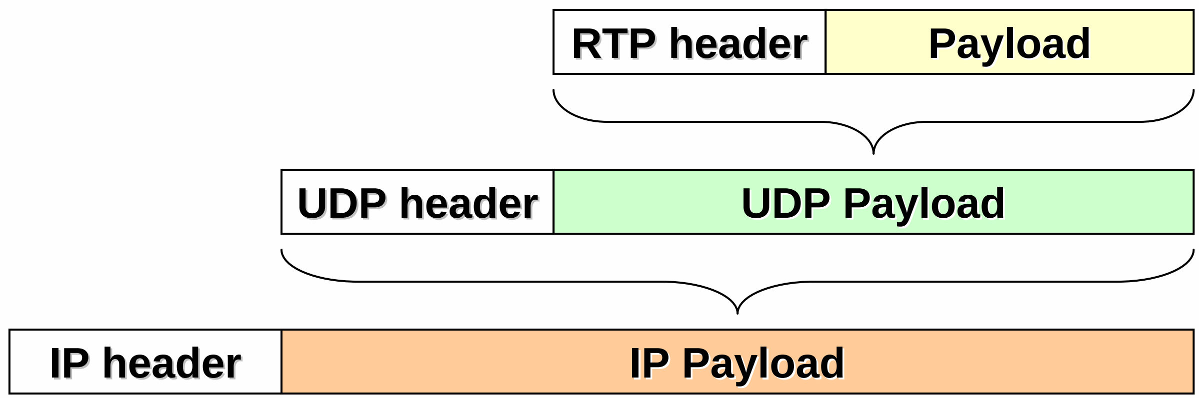
- v Version
- p Padding
- x Extension
- cc CSRC Counter
- m Marker

- pt Payload Type
- Sequence number
- Timestamp
- SSRC: synchronization source
- CSRC: contributing source(s)

Transmission Protocols

- The RTP packet including header can be transported over any transport layer protocol
- Most often UDP is used





- If the x-bit is set, payload type specific extensions follow the header:
  - 16 bit identifier
  - 16 bit length of the extension header
  - Variable length extension data

Image Source: http://www.ind.alcatel.com/library/e-briefing/eBrief_RTP.pdf

Transmission Protocols

- RTP ***profiles*** define the data transmission process in RTP packets

- Profiles are specified in IETF RFC 1890
  "*RTP Profile for Audio/Video conferences with minimal control*"
  - Sampling rate
  - Sampling precision
  - Payload types
  - Meaning of the timestamp field

- ***Multiplexing*** is provided by destination port transport address
  - Allows for payload type switching
  - Allows for selection of different network paths
  - Allows for selection of media subset (e.g. audio only)
  - Feedback reports have no payload type!

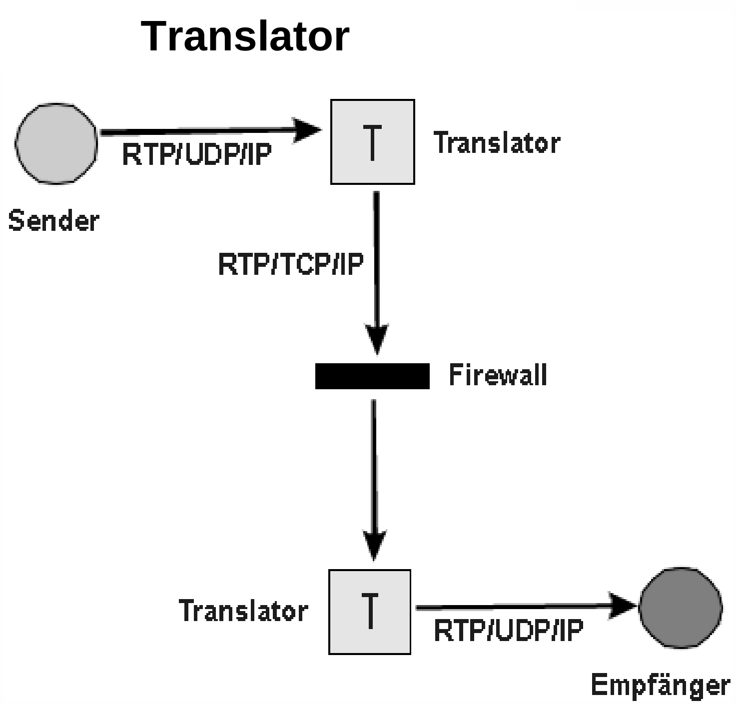
Transmission Protocols

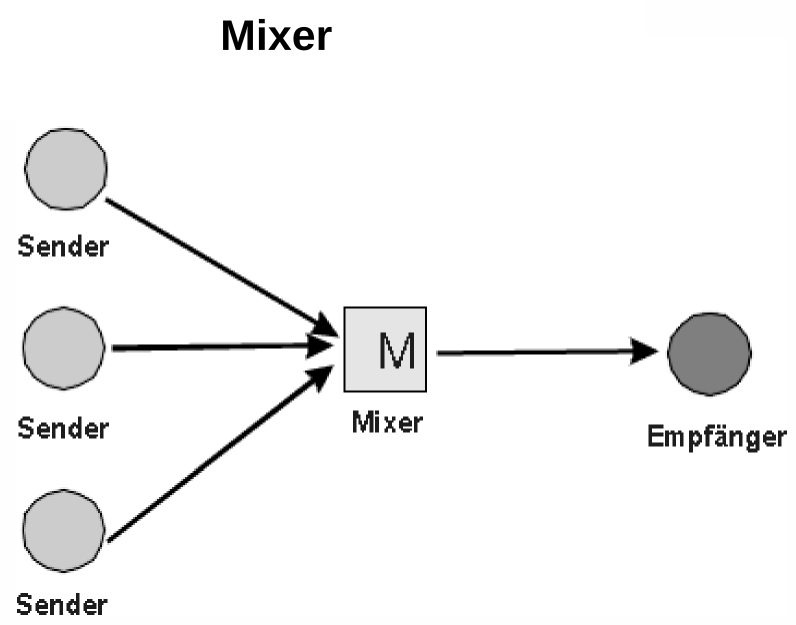
- Example RTP payload types defined in RFC 3550 & RFC 3551

Payload Type	Codec	Audio/Video	Clock Rate (Hz)	Channels (Audio)
0	PCM u-law	A	8000	1
2	G721	A	8000	1
15	G.728	A	8000	1
...				
25	CelB	V		
26	JPEG	V		
...				
72-76	Reserved	N/A	N/A	N/A
77-95	Unassigned	?		
96-127	Dynamic	?		

Transmission Protocols

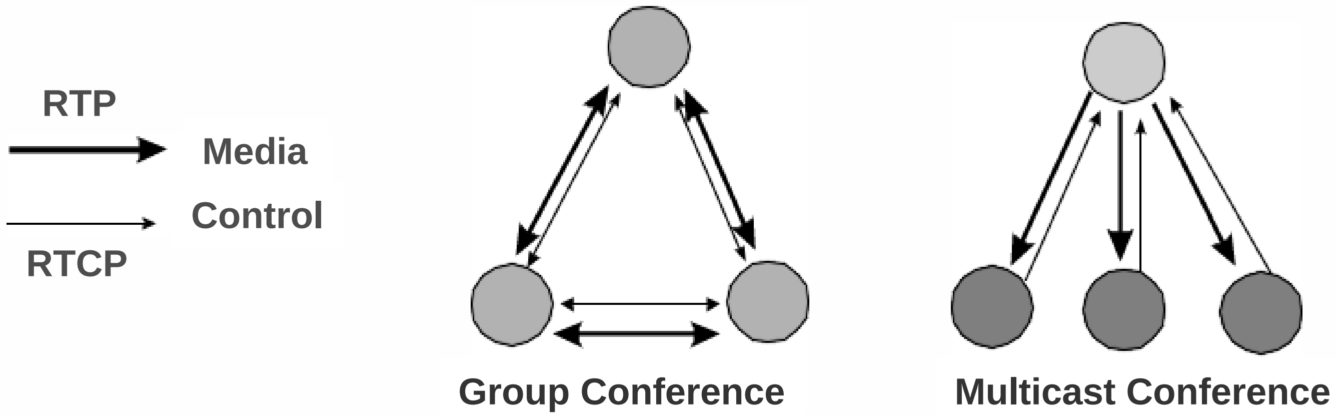
- RTP ***Translators*** can be used to change the underlying transport protocol
- RTP ***Mixer*** can be used to combine/interleave multiple streams





Transmission Protocols

- ***RTCP – Realtime Transport Control Protocol***
  - Defined in IETF RFC 1889 (obsoleted by RFC 3550)
  - QoS measurements and feedback on quality of the data distribution
  - Information about participants
  - Meta-Information (CNAME, Audio/Video)
  - Sent and received by all, allows for third party monitoring



Transmission Protocols

- Overall data rate must be controlled
  - All messages are multicasted to all participants
  - All participants monitor control data
  - 5% of session bandwidth for RTCP, 5 sec minimum interval, randomly distributed [0.5,1.5]
  - 25% of RTCP traffic for sender messages (adjustable)

- RTP uses even port number, RTCP the next higher odd port number

- Possible RTCP packet formats:
  - SR: Sender Report
  - RR: Receiver Report
  - SDES: Source Description, includes CNAME (canonical name)
  - BYE: Explicit leave
  - APP: Application specific data

Transmission Protocols

- RTCP Sender Report
  - v = version, 2 Bit = 2
  - p = padding, 1 Bit
  - RC = reception count, 5 Bit
  - PT = Type, 8 Bit = 200 (SR)
  - Packet length, 16 Bit
  - SSRC, 32 Bit
  - NTP time, 64 Bit
  - RTP time, 32 Bit
  - total number of sent packets
  - total number of sent bytes
  - Reception Reports

v	p	RC	PT	length
SSRC				
NTP time				
RTP time				
# packets				
# bytes				
SSRC_n				
Loss%	Loss total			
last packet				
jitter				
last sender report timestamp				
delay since last sender report				

per received data stream

Transmission Protocols

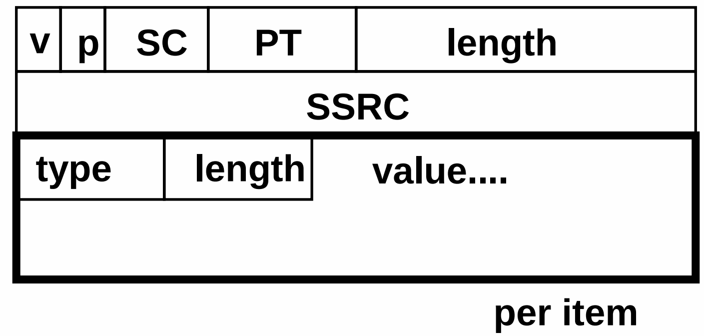
- RTCP Receiver Report
  - v = version, 2 bit = 2
  - p = padding, 1 bit
  - RC = reception count, 5 Bit
  - PT = packet type, 8 bit = 201 (RR)
  - packet length, 16 Bit
  - SSRC, 32 Bit
  - Reception Reports

v	p	RC	PT	length
SSRC				
SSRC_n				
Loss%	Loss total			
last packet				
jitter				
last sender report timestamp				
delay since last sender report				

per received stream

Transmission Protocols

- RTCP Source Description
  - v=version, 2 bit = 2
  - v = Version, 2 bit = 2
  - p = padding, 1 bit
  - SC = source count, 5 Bit
  - PT = packet type, 8 bit = 202 (SDES)
  - packet length, 16 Bit
  - Item
    - type, 8 Bit
    - length, 8 Bit
    - value
  - CNAME, Typ = 1: user@domain
  - NAME, Typ = 2
  - EMail, Typ = 3
  - phone number, location, application




```
schrader@cyberlounge.isnm.de
Andreas Schrader, ISNM
Andreas.Schrader@acm.org
```

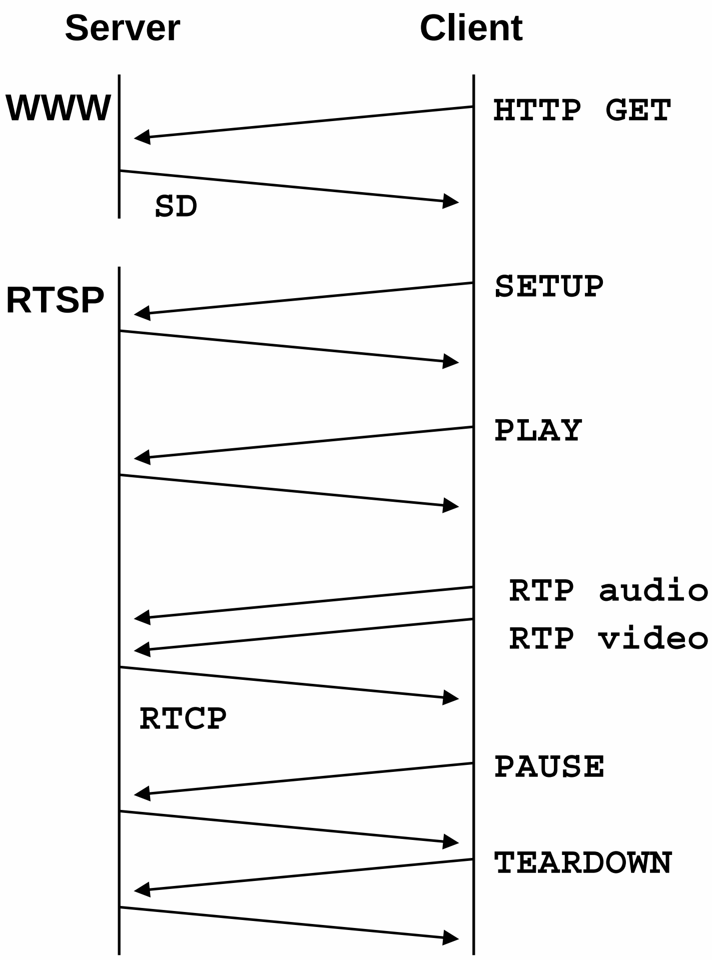
Transmission Protocols

- RTCP in Translators and Mixers
  - Translator that do not modify RTP data forward RTCP messages unchanged
  - Translators that do modify RTP data must adapt RTCP messages to provide correct information
  - Mixers do not forward RTCP messages, but create their own reports for both sides

- RTP/RTCP Summary
  - Provides helping mechanisms for applications
  - Does not guarantee anything
  - Is protocol neutral (UDP/IP, ATM, proprietary protocols)
  - Supports unicast and multicast
  - Scalable (supports large user groups)

Transmission Protocols

- ***RTSP – Realtime Streaming Protocol***
- Defined in RFC 2326
    - Stream Control
    - may use UDP, RTP, TCP, RSVP
    - as well as HTTP, SIP, SDP, IP Multicast
  - Data request from a media server (unicast)
  - Remote stream control (Internet VCR)



- RTSP Applications
  - Lectures, seminars
  - On-demand instruction
  - Entertainment
  - Remote Digital Editing
  - Voice Mail

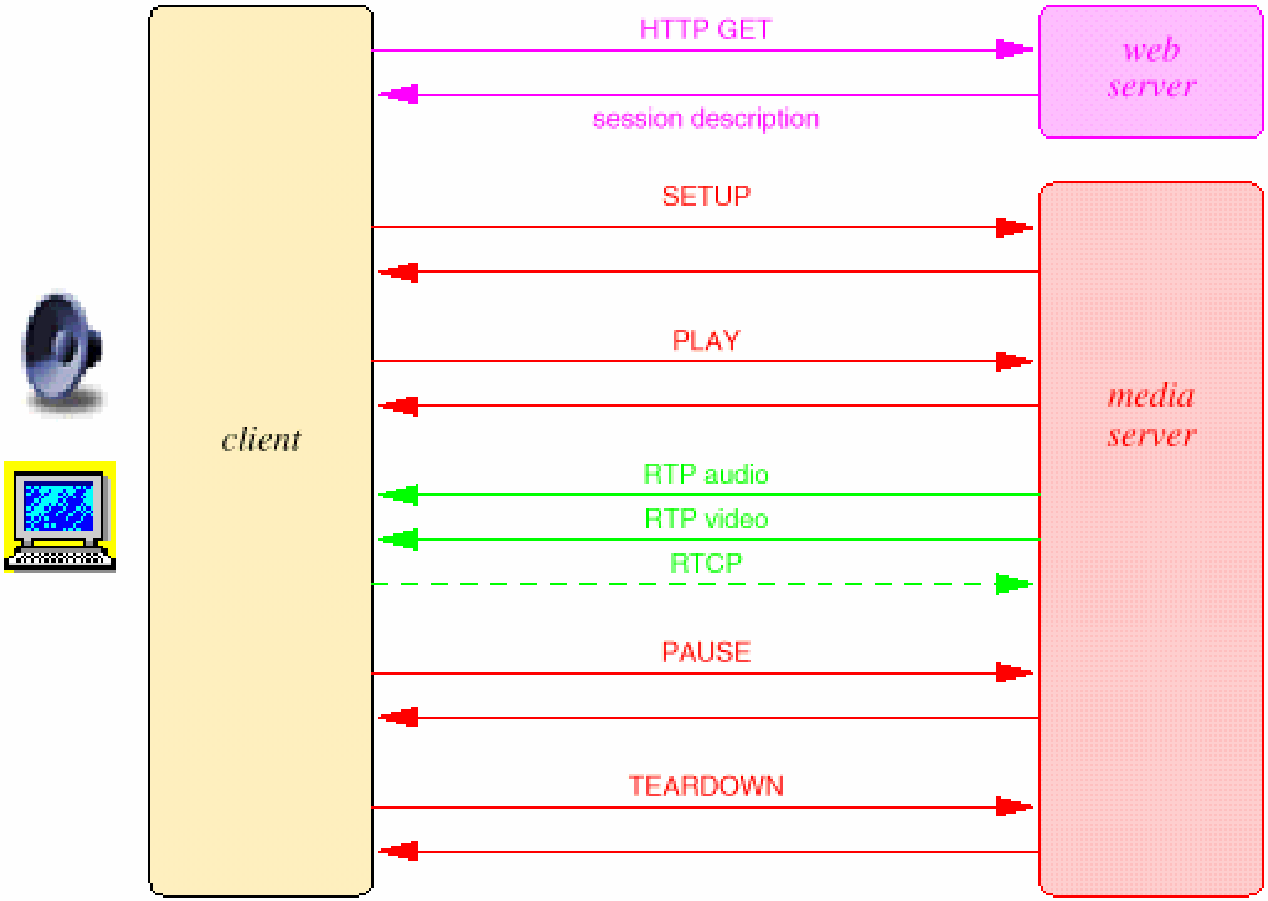
Transmission Protocols

- ***RTSP*** Methods

OPTIONS	get available methods
SETUP	establish transport
ANNOUNCE	change description of media object
DESCRIBE	get (low-level) description of media object
PLAY	start playback, reposition
RECORD	start recording
REDIRECT	redirect client to new server
PAUSE	halt delivery, but keep state
SET PARAMETER	device or encoding control
TEARDOWN	remove state

Transmission Protocols

- ***RTSP*** Session



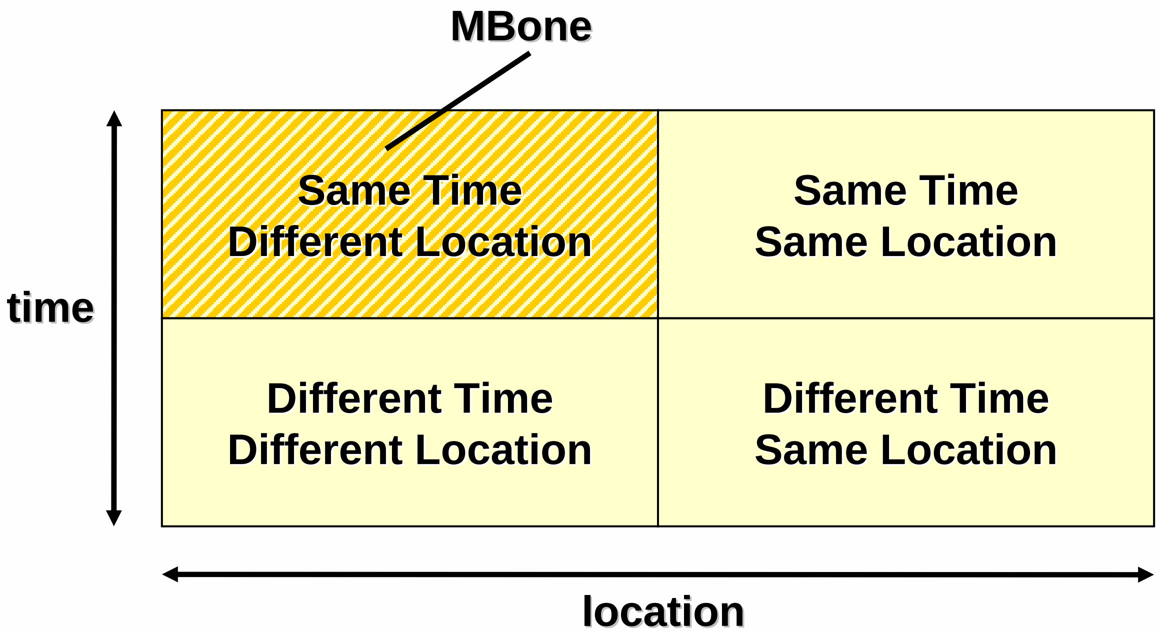
MBone
vic/vat

Transmission Protocols

- ***MBONE*** = Virtual ***M***ulticast ***B***ackbone ***O***n the Inter***NE***t
  - Uses class D IP addresses of the Internet
  - Enables ***group communication*** scenarios
  - Invented by Steve Deering (Stanford University), later developed by Xerox PARC. Further enhances by Van Jacobson (Berkeley) and Steve Casner
  - First adopted after the 1992 IETF meeting
  - The MBONE network consists of multicast capable routers

Transmission Protocols

- ***Groupware*** is a piece of technology that enables group work
- One possible categorization of groupware scenarios:

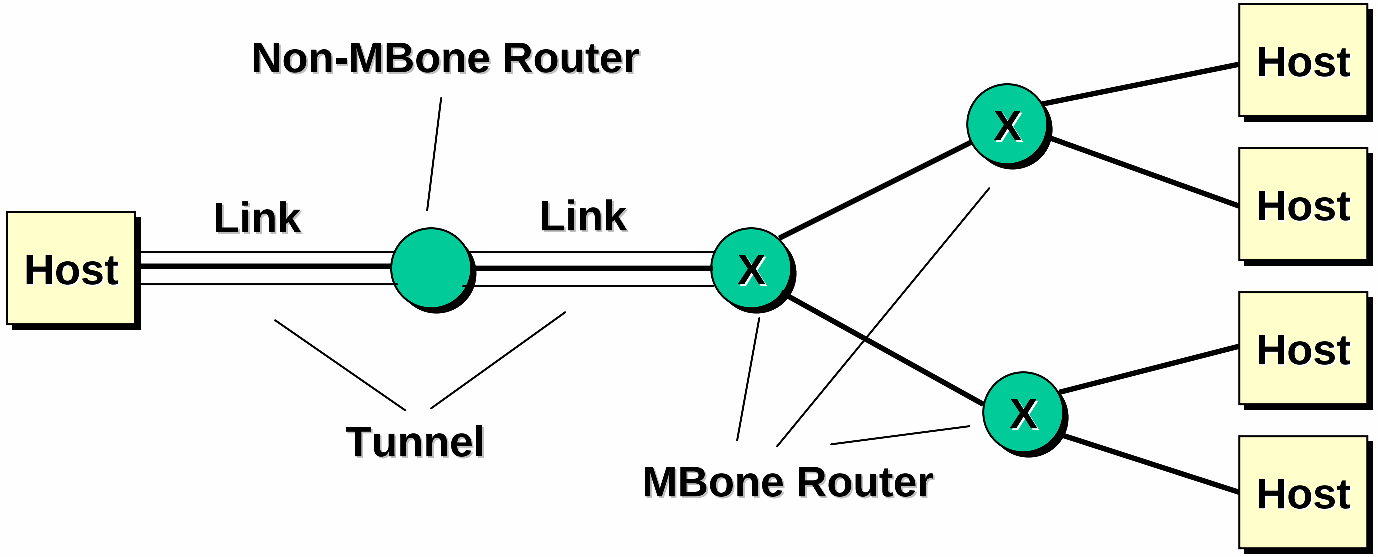


- MBone supports some scenarios for same time, different location groupware (also called real-time or synchronous collaboration)
- Examples: Video Conferences, Multiparty Phone calls, etc.

Transmission Protocols

❑ *MBONE Structure*

- Instead of duplication each packet for each receiver in a group scenario MBone routers deal with a virtual group address



- MBone routers support class D address space
- IANA (Internet Address Number Authority) defines range 224.2.*.* for MBone

Transmission Protocols

- ***MBONE Structure***
  - Host can join a multicast session by sending a IGMP join message
  - Only subscribed sites will receive packets of that group address
  - Every Internet host can send packets to any group
  - Routing is performed using
    - **DVMRP** – Distance Vector Multicast Routing Protocol [RFC 1075]
    - **MOSPF** – Multicast Open Shortest Path Factor
  - Each multicast address has a ***Time To Live*** (TTL) value, specifying how many hops are allowed to pass
  - ***Tunnels*** are used to forward multicast IP packets through Unicast (legacy) routers (packet encapsulation)
  - Since more and more router products support multicast, the MBone can be build with reduced number of tunnels

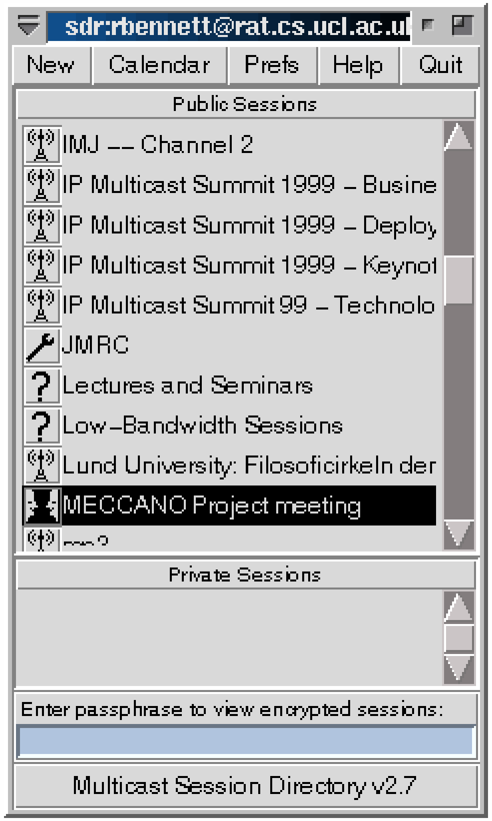
    (virtual backbone -> real backbone)

Transmission Protocols

- ***MBONE Software***
  - Video (nc, vic, mash-vic, RendezVou, CuSeeMe, etc.)
  - Whiteboards (wb, dlb, etc.)
  - Text-Tools (nte, etc.)
  - Session Announcement & Controls (srd, etc.)
  - Recording, Debugging, Multicast Routing, etc.
    - http://vcc.urz.tu-dresden.de/mbone/software.html
  - OS support:
    IRIX, HPUX, Linux, FreeBSD, Win95/98/NT4.0/2000/XP, Solaris

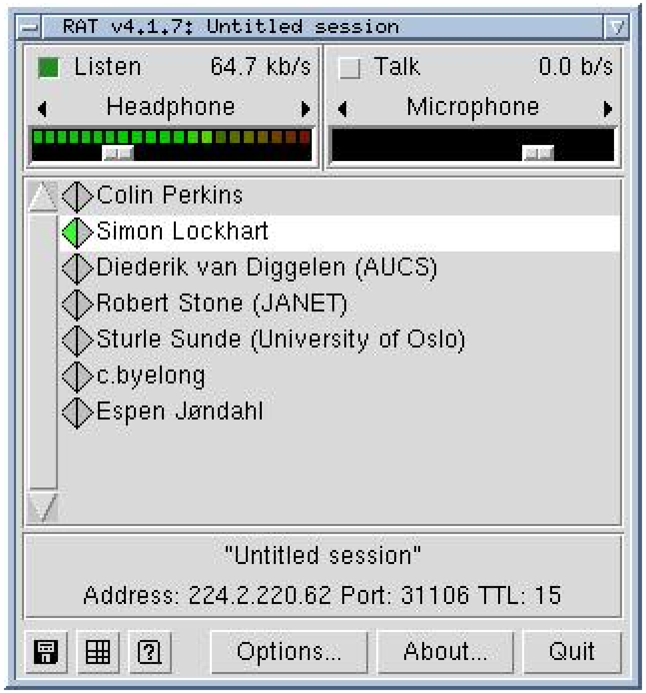
Transmission Protocols

- ***SDR*** – ***S***ession ***D***i***R***ectory Tool
  - SDR is a session directory tool designed to allow the advertisement and joining of multicast conferences on the MBone
  - Based on Tcl/Tk script language (http://www-mice.cs.ucl.ac.uk/multimedia/software/sdr/)
  - MBone/SDR User Directory Service in Germany can be found at (http://meta.rrzn.uni-hannover.de/mbone.html) Last Update was in 1999.
  - SDR announced sessions can be joined using audio or video conferencing tools



Transmission Protocols

- ***RAT*** – ***R***obust ***A***udio ***T***ool
  - RAT is an open-source audio conferencing and streaming application that allows users to participate in audio conferences over the Internet
  - RAT is just an audio application, it does not perform call services like user location, neither does it listen to session announcements
  - Feature:
    - Streaming with RTP
    - Monitoring using RTCP
    - Repair mechanism for damaged audio streams
    - Secure conferencing
    - Transcoder mode
    - Adaptive Playout mode
    - Several audio codecs



Source: http://www-mice.cs.ucl.ac.uk/multimedia/software/rat/

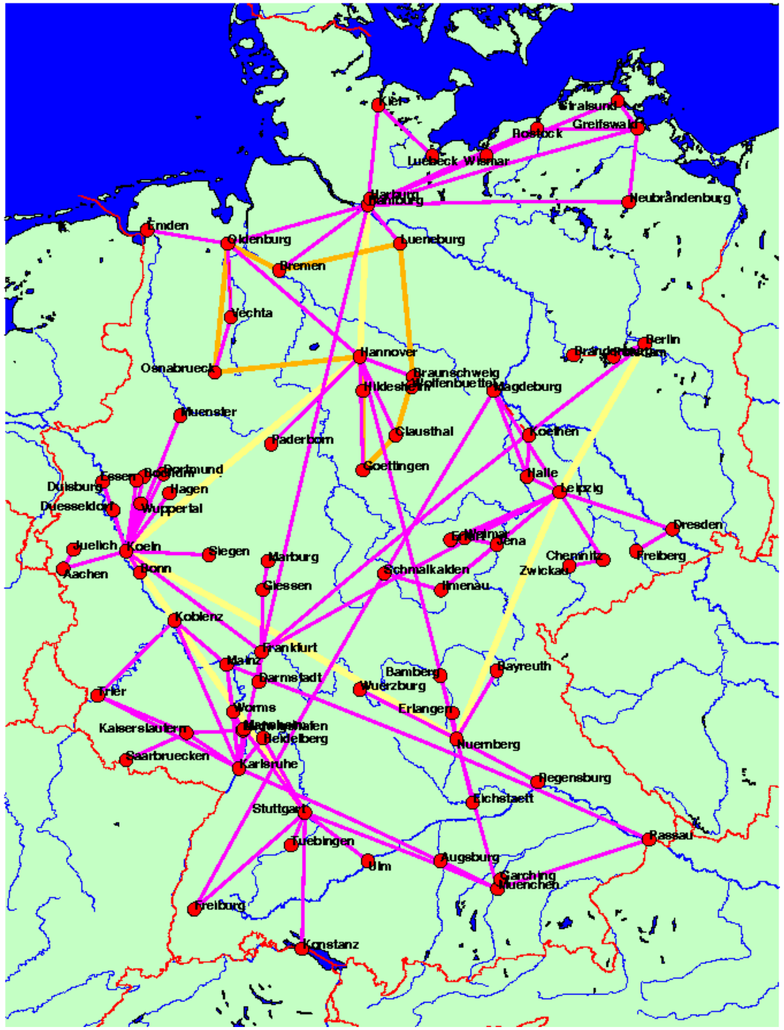
Transmission Protocols

- ***VIC*** – ***VI***deo ***C***onferencing Tool
  - VIC is a video conferencing application developed by the Network Research Group at the Lawrence Berkeley National Laboratory in collaboration with the University of California, Berkeley
  - Features:
    - Streaming with RTP
    - Monitoring with RTCP
    - JPEG, H.261 video encoder
    - Voice switched viewing windows
    - Multiple dithering algorithms
    - Interactive title generation
    - Frame rate, data rate and quality control

Source: http://www-nrg.ee.lbl.gov/vic/

Transmission Protocols

- **_German MBone_**
  - Since April 2000 every network node (~3000 routers) of the german research network(DFN) supports multicast.
  - Since then MBone is called ***DFN Multicast*** and is a standard service of the DFN backbone
  - July 2002
    - Yellow: MBone-DE Backbone
    - Magenta: MBone-DE leaf links
    - Red: MBone-DE down links



Source: http://www.mbone.de