Multimedia Protocols

Difference with classic applications

- Highly delay-sensitive
  - Packets are useless if they arrive too late
- Loss-tolerant (for the most part)
  - Packet loss can be concealed

Categories of Internet audio/video

- Streaming Stored Audio and Video
  - On-demand requests for compressed and stored audio/video files
- Streaming Live Audio and Video
  - Broadcasting of radio or TV programs over the Internet
- Real-Time Interactive Audio and Video
  - Internet telephony or Internet teleconferencing
- Others

Streaming Stored Audio and Video

- The multimedia content has been prerecorded and stored on a server
- User may pause, rewind, forward, etc…
- The time between the initial request and display start can be 1 to 10 seconds
- **Constraint:** after display start, the playout must be continuous

Streaming Live Audio and Video

- Similar to traditional broadcast TV/radio, but delivery on the Internet
- Non-interactive just view/listen
  - Can not pause or rewind
- Often combined with multicast
- The time between the initial request and display start can be up to 10 seconds
- **Constraint:** like stored streaming, after display start, the playout must be continuous

Real-Time Interactive Audio and Video

- Phone conversation/Video conferencing
- **Constraint**: delay between initial request and display start must be small
  - Video: <150 ms acceptable
  - Audio: <150 ms not perceived, <400 ms acceptable
- **Constraint**: after display start, the playout must be continuous

Others

- Multimedia sharing applications
  - Download-and-then-play applications
- Distance learning applications
  - Coordinate video, audio and data
  - Typically distributed on CDs

Challenges

- TCP/UDP/IP suite provides best-effort, no guarantees on expectation or variance of packet delay

- Performance deteriorate if links are congested (transoceanic)

- Most router implementations use only First-Come-First-Serve (FCFS) packet processing and transmission scheduling

Other Issues

- Limited bandwidth
  - Solution: Compression
- Packet Jitter
  - Solution: Fixed/adaptive playout delay for Audio (example: phone over IP)
- Packet loss
  - Solution: FEC, Interleaving

Packet Loss

- Packet never arrives or arrives later than its scheduled playout time.
- Since retransmission is inappropriate for Real Time applications, Forward Error Correction or Interleaving are used to reduce loss impact.

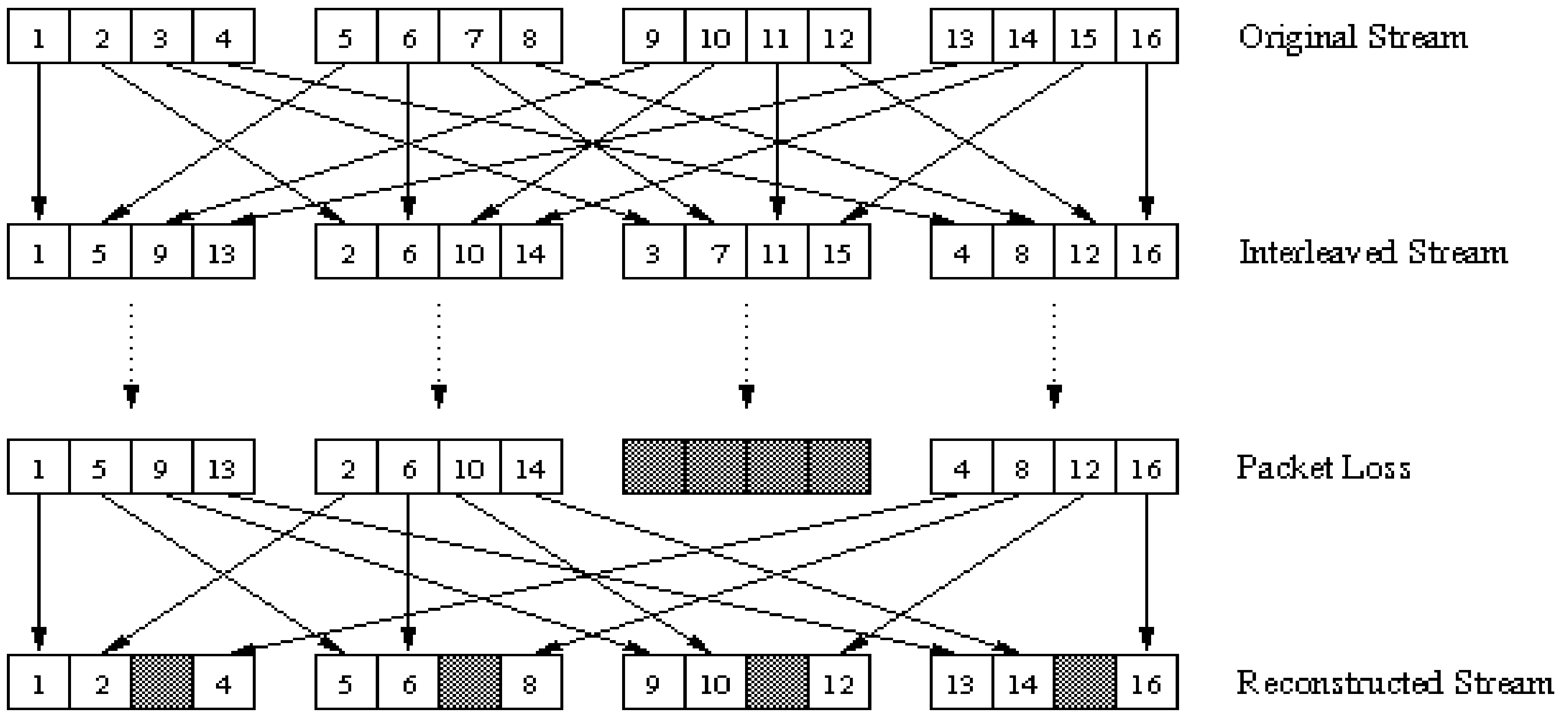
Forward Error Correction

- Send redundant encoded chunk every n chunks (XOR original n chunks)
  - If 1 packet in this group is lost, it can be reconstructed
  - If >1 packets lost, it cannot be recovered
- Disadvantages
  - The smaller the group size, the larger the overhead
  - Playout delay increases

Packet Loss

Interleaving

- Divide 20 msec of audio data into smaller units of 5 msec each and interleave
- Upon loss, have a set of partially filled chunks



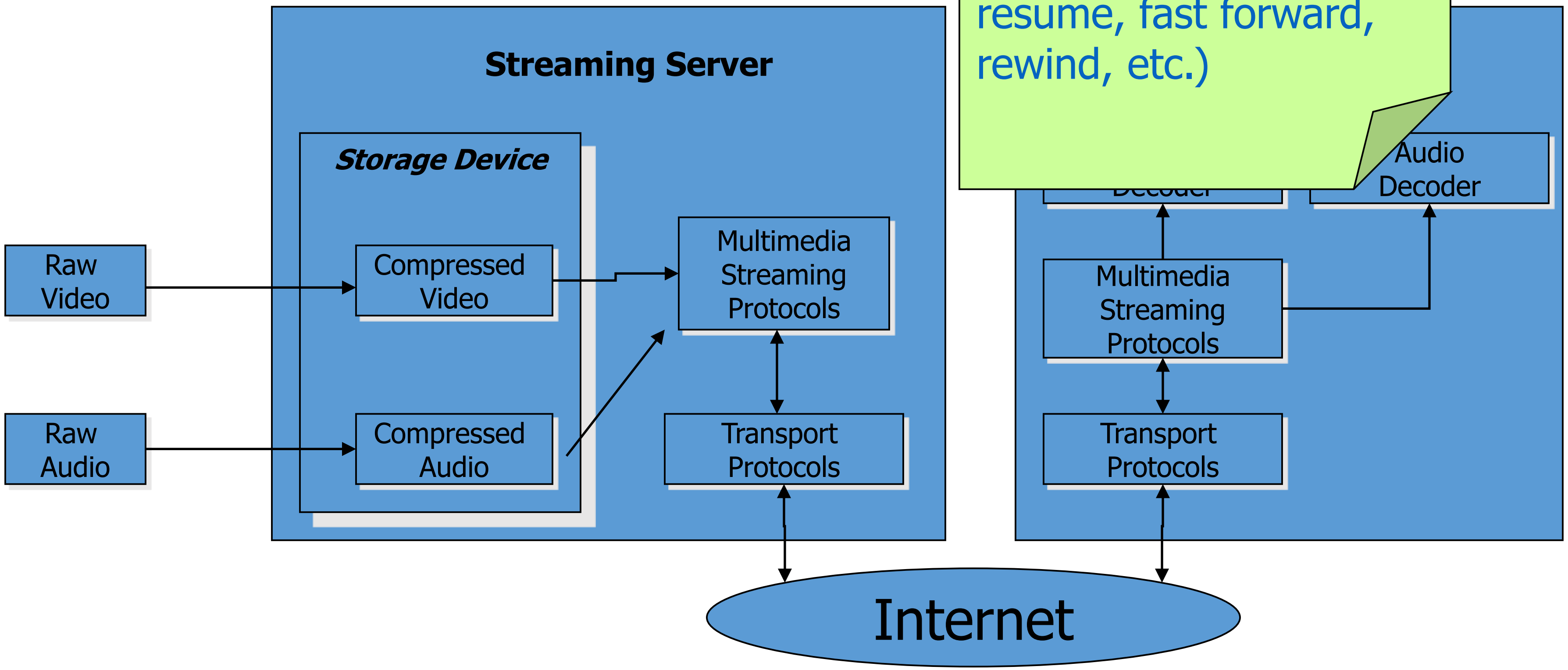
Recovering from packet loss
Receiver-based Repair

- The simplest form: Packet repetition
  - Replaces lost packets with copies of the packets that arrived immediately before the loss
- A more computationally intensive form: Interpolation
  - Uses Audio before and after the loss to interpolate a suitable packet to cover the loss

Streaming Stored Audio / Video

• Multimedia Streaming: Clients request audio/video files from servers and pipeline reception over the network and display

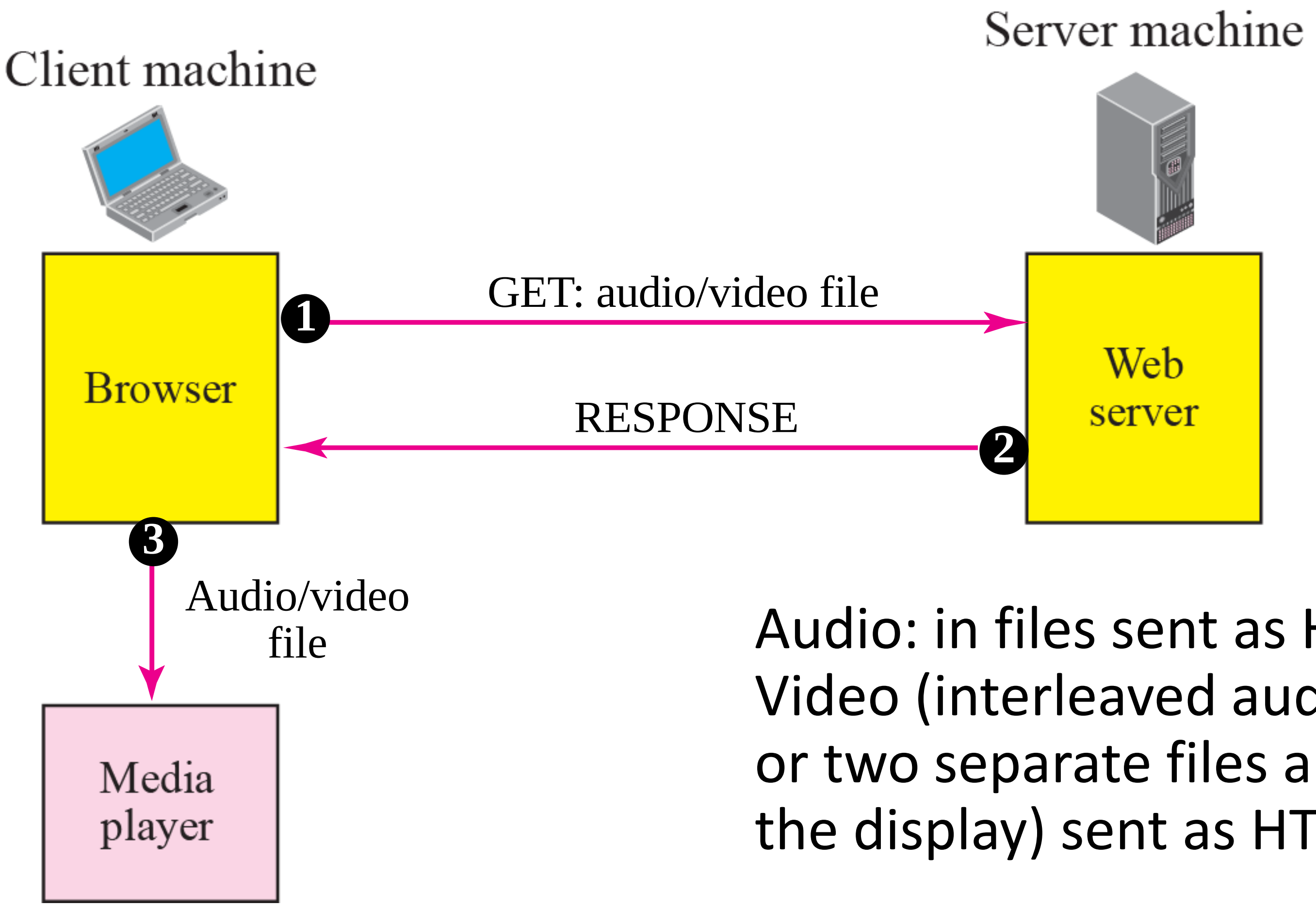
<u>User’s perspective:</u>
- Quick start without waiting for full download.
- Coming continuously without interruption.
- VCR operation (pause, resume, fast forward, rewind, etc.)



Streaming Stored Audio / Video

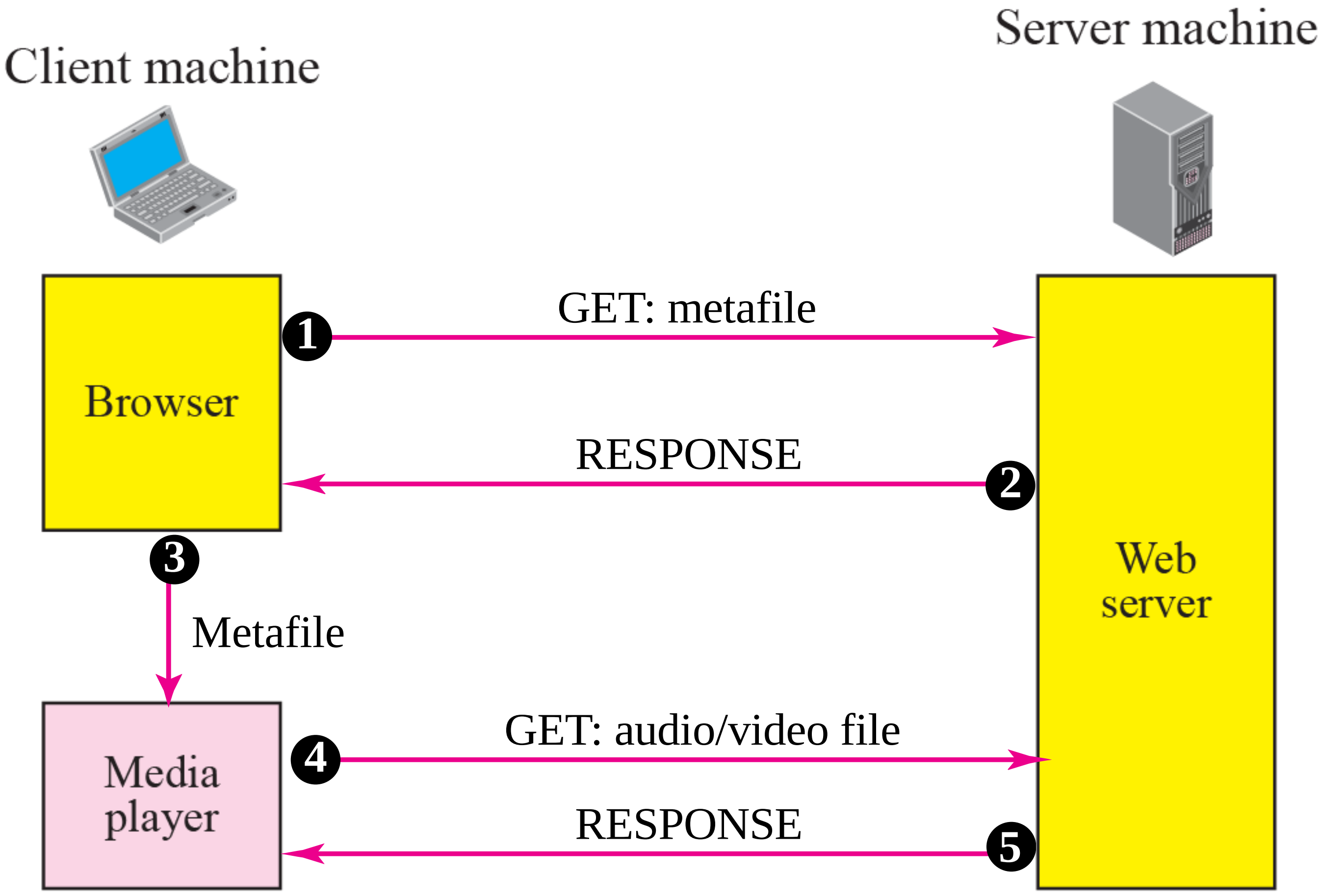
- First Approach: Using a Web Server
- Second Approach: Using a Web Server with Metafile
- Third Approach: Using a Media Server
- Fourth Approach: Using a Media Server and RTSP

Using a web server



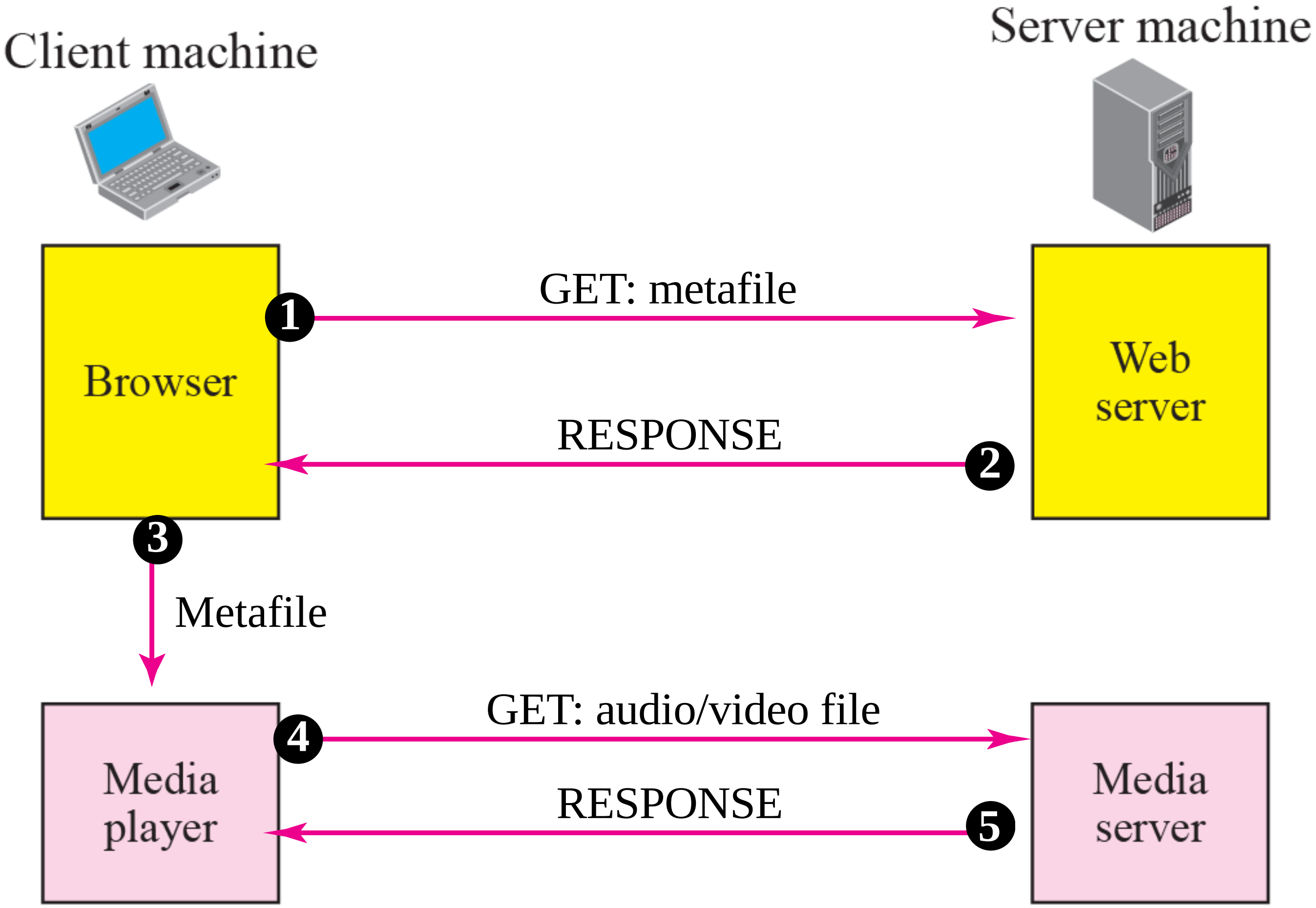
Audio: in files sent as HTTP objects
Video (interleaved audio and images in one file, or two separate files and client synchronizes the display) sent as HTTP object(s)

Using a Web Server with Metafile



Web browser requests and receives a **Meta File** (a file describing the object) instead of receiving the file itself;
Browser launches the appropriate Player and passes it the Meta File;
Player sets up a TCP connection with a streaming server and downloads the file

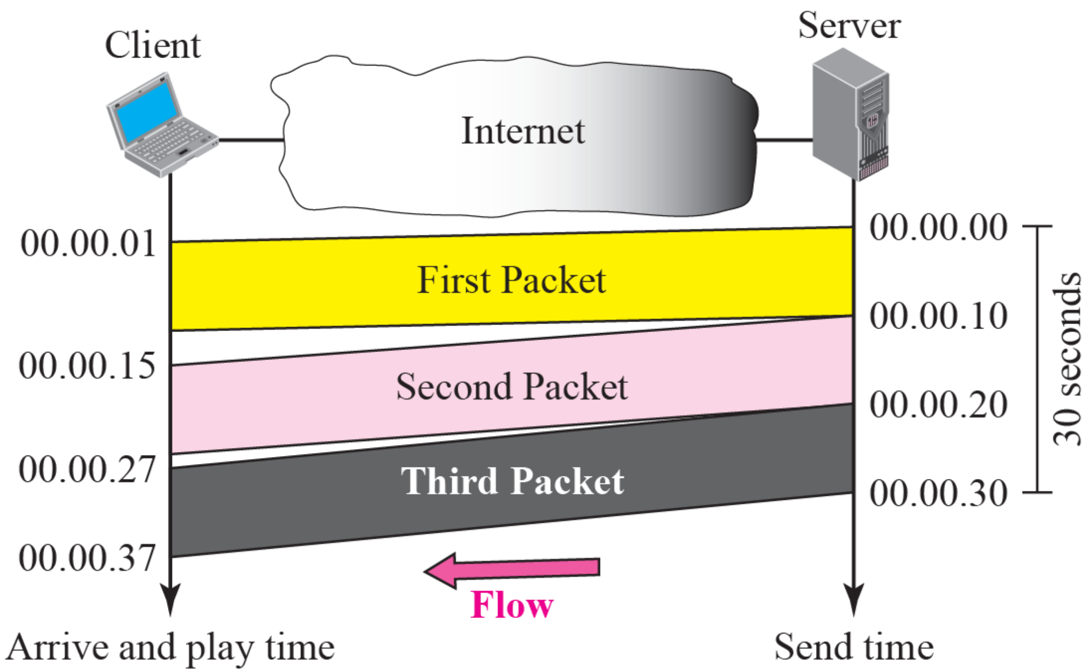
Using a Media Server



Streaming Live Audio Video

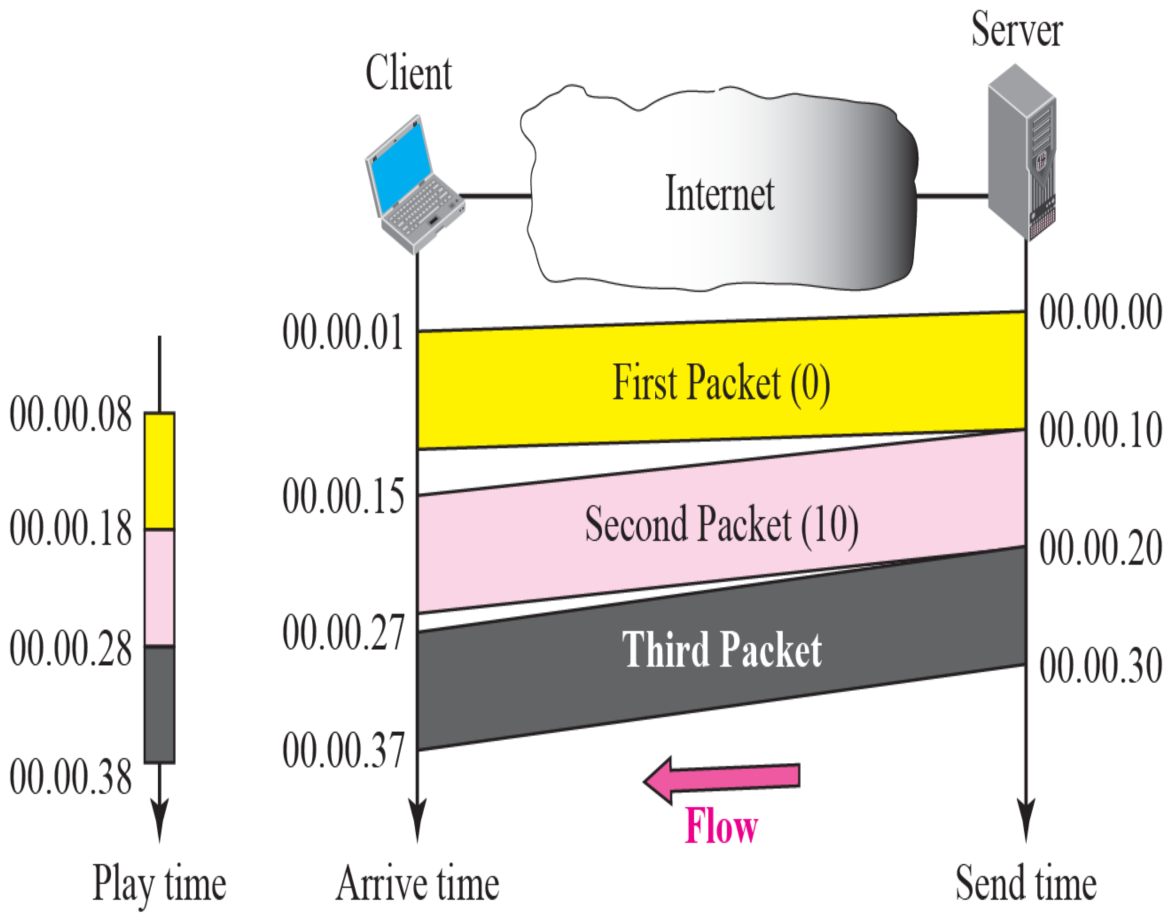
- Similar to streaming stored audio/video.
- However, in the first application, the communication is unicast and on-demand. In the second, the communication is multicast and live.

Real-time Audio Video



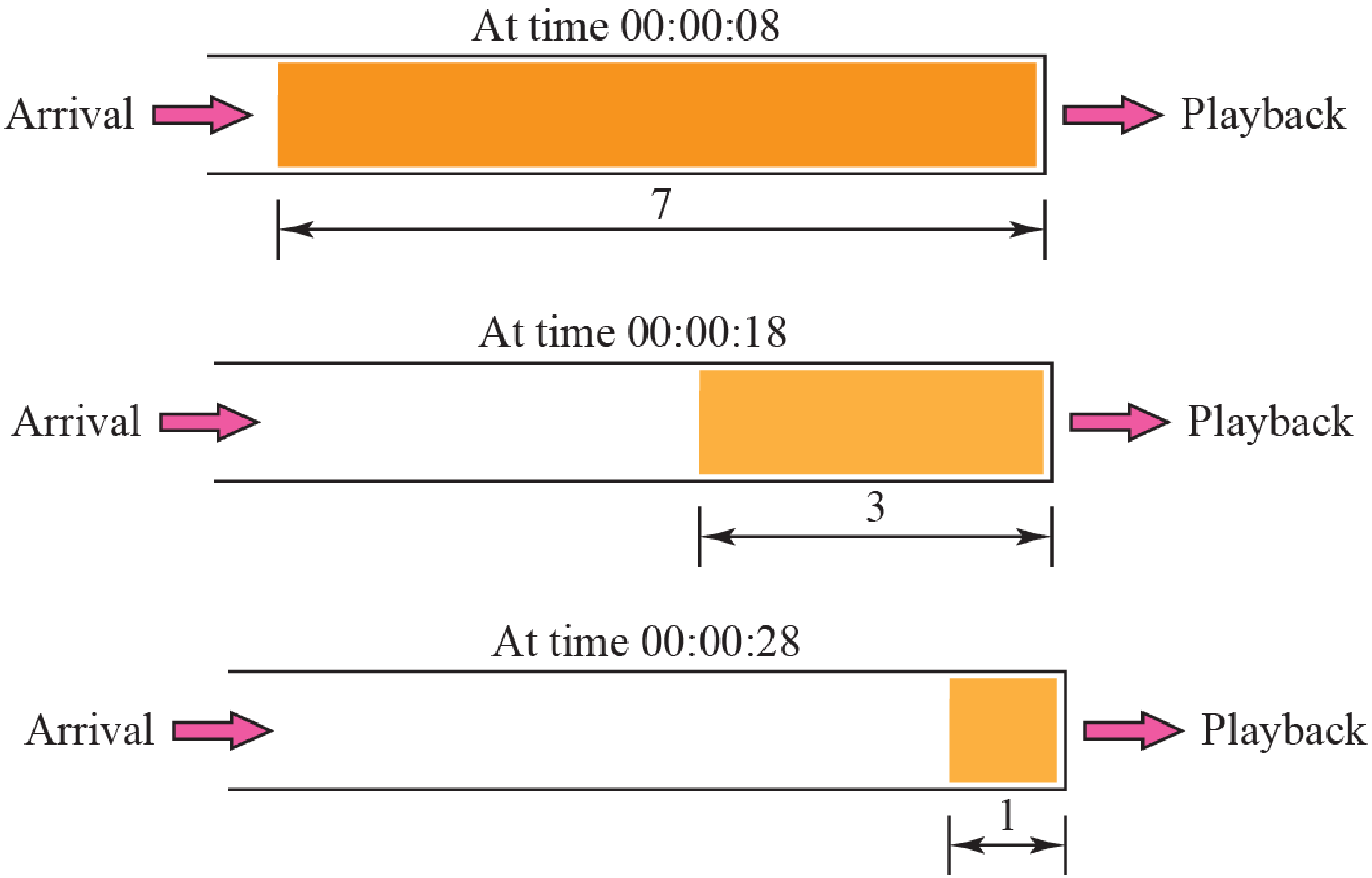
- Effect of Jitter
  - Jitter is introduced in real-time data by the delay between packets

Real-time Audio Video



- To prevent jitter, we can timestamp the packets and separate the arrival time from the playback time.

Playback buffer



- Real-time traffic requires
  - Playback buffer
  - A sequence number to track packet loss
  - Support for multicasting

Other services

- Translation

changing the encoding of a payload to a lower quality to match the bandwidth of the receiving network.

- Mixing

combining several streams of traffic from different sources into one stream.

  - *Such as audio and video*

TCP is not suitable interactive streaming media traffic for its

- *error control mechanism.*
- *No support for timestamping.*
- *No multicasting.*

UDP does not have

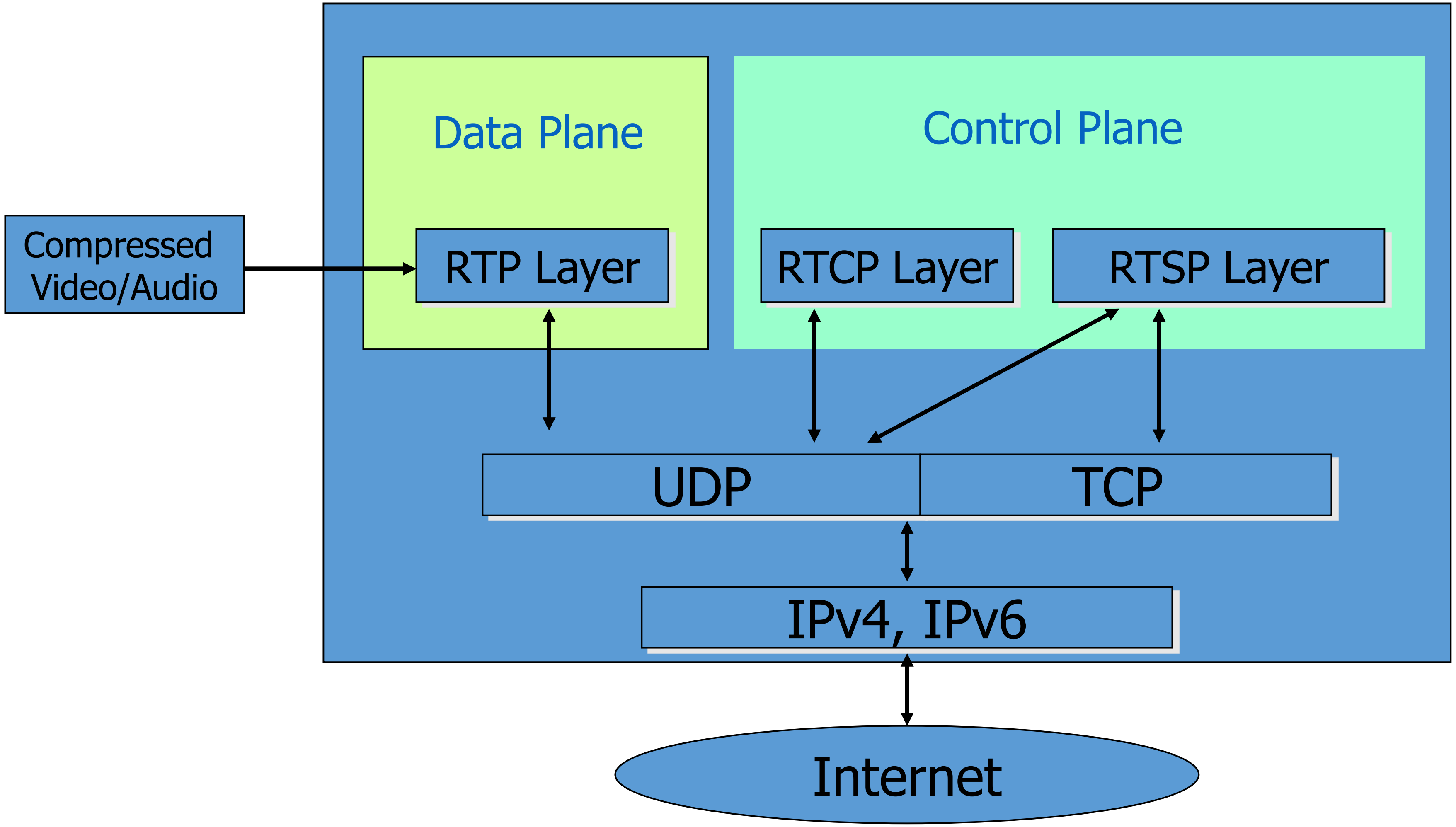
- *sequence numbers.*
- *No timestamping.*
- *No support for mixing*

- *New protocols are needed*

Popular protocols for serving media

- Network transmission control
  - RTP – Realtime Transmission Protocol
  - RTCP – Realtime Transmission Control Protocol
- Session control
  - Real-Time Streaming Protocol (RTSP)
  - Session Description Protocol (SDP) – textual representation of sesion
- VOIP – SIP – Session Initiation Protocol
  - Signaling for IP Telephony
- SAP – Session announcement protocol for multicast sessions

Protocol stack for media streaming

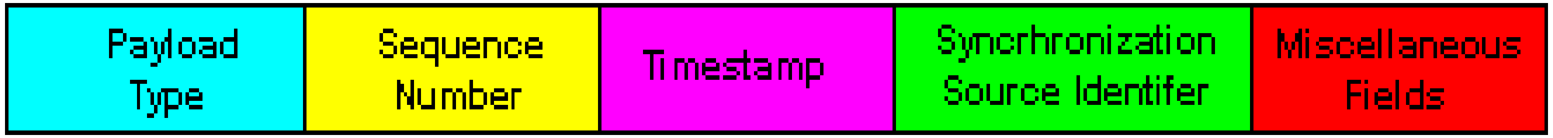


Real Time Protocol (RTP)

- RTP logically extends UDP
  - sits between UDP and application
  - end-to-end transport functions suitable for real-time audio/video applications over multicast and unicast network services
  - implemented as an application library
  - RTP uses a temporary even-numbered UDP port
- What does it do?
  - Framing
  - Multiplexing
  - Synchronization
  - Feedback (RTCP)

Real-time Protocol - RTP

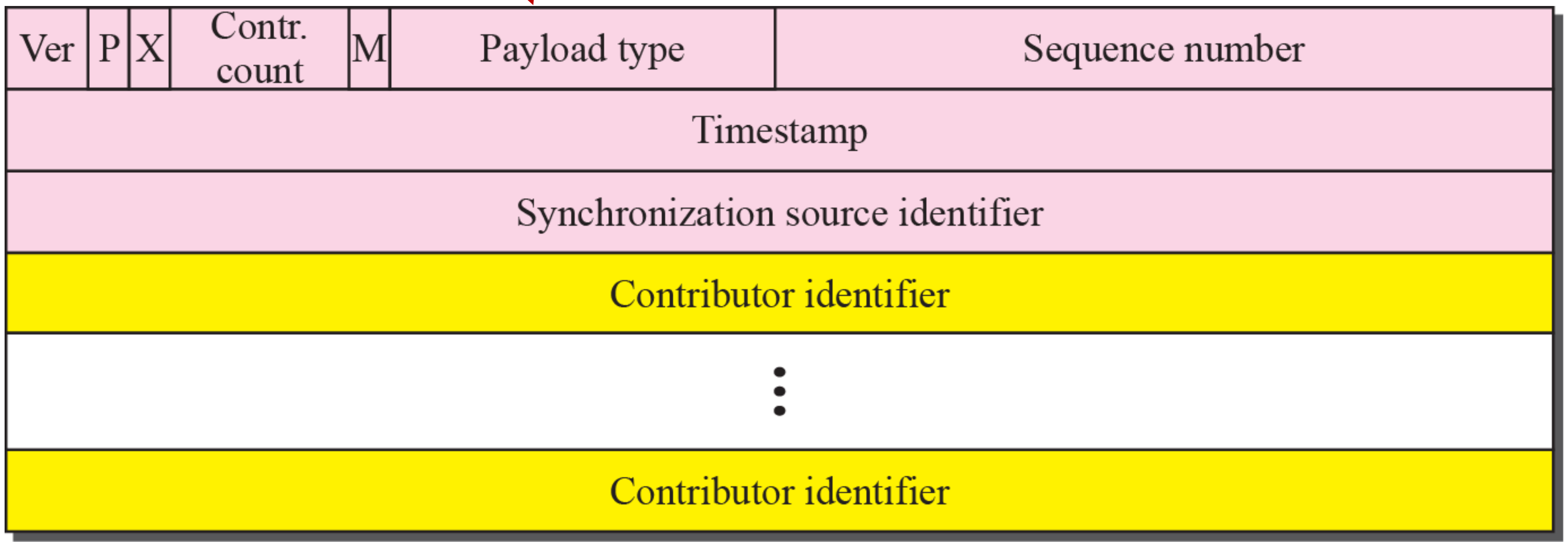
- RTP is a data transfer protocol and RTCP is a control protocol.
- RTP provides services for
  - **Payload type identification**: Identify which kind of information is being transmitted, RTP provides 128 possible different types of encoding; eg MPEG2 video, etc.
  - **Sequencing**: Reassemble the stream and detect packet loss.
  - **Timestamping**: Assure synchronization. Also used for jitter calculation
  - **Source identification:** Provide a means for the receiver to distinguish different sources.
- RTP does not provide
  - Quality of service
  - Reliability in packet delivery
  - Security



RTP Header

- **Payload Type**: 7 bits, providing 128 possible different types of encoding; eg PCM, MPEG2 video, etc.
- **Sequence Number**: 16 bits; used to detect packet loss

- **Timestamp**: 32 bytes; gives the sampling instant of the first audio/video byte in the packet; used to remove jitter introduced by the network
- **Synchronization Source identifier (SSRC)**: 32 bits; an id for the source of a stream; assigned randomly by the source

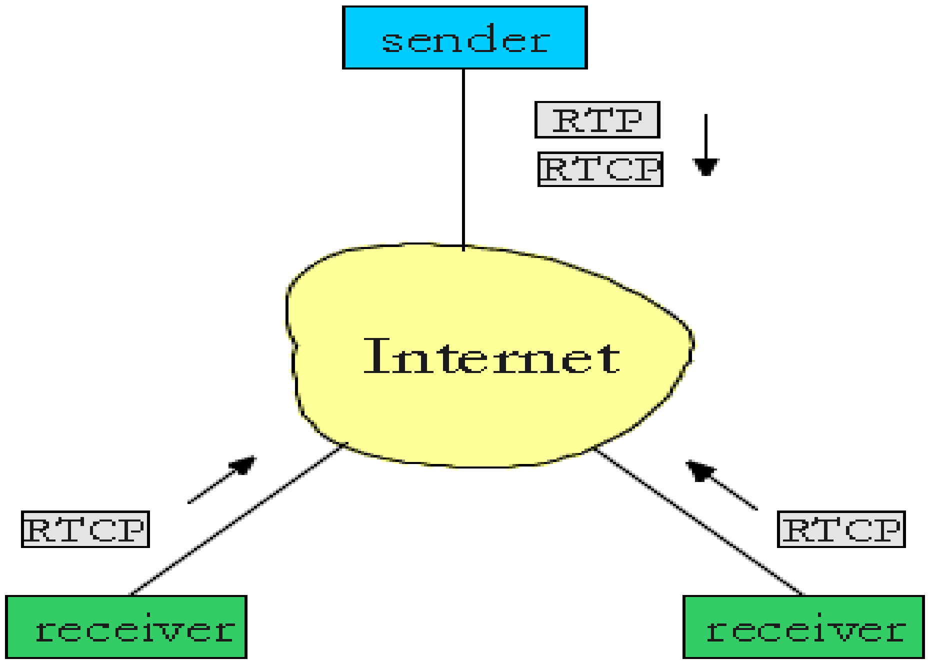


Timestamp vs. Sequence No

- Timestamps relate packets to real time
  - Timestamp value sampled from a media specific clock
- Sequence number relates packets to other packets
- Example of silent audio –
  - Consider audio data type
  - What is sent during silence?
    - Not sending anything
  - Why might this cause problems?
    - Other side needs to distinguish between loss and silence
  - Receiver uses timestamps and sequence no. to figure out what happened

RTP Control Protocol (RTCP)

- Used in conjunction with RTP to exchange control information between the sender and the receiver.
- Control connection is held over a different channel than the RTP.
- Uses an odd-numbered UDP port number that follows the port number selected for RTP.
- Reports can be *Receiver reception*, *Sender*, and *Source description.*
- Reports contain statistics such as the number of packets sent, number of packets lost, inter-arrival jitter
- Multiple RTCP packets can be concatenated by translators/mixers



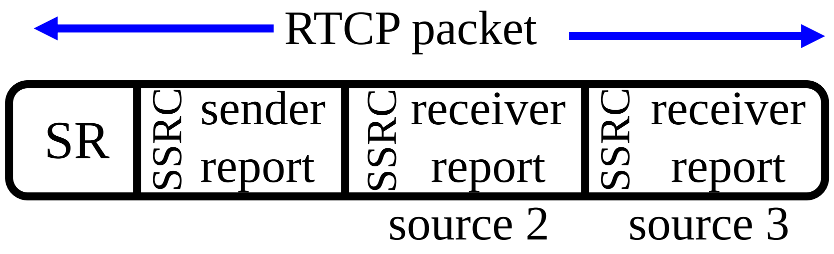
RTP Control Protocol (RTCP)

- RTCP provides
  - **QoS Feedback**: In form of sender reports/receiver reports. Senders adjust transmission rate based on reports.
  - **Participant Identification**: Human-friendly source identification.
  - **Control Packets Scaling**: Typically, limit the RTCP bandwidth to 5% of the session bandwidth, divided between the sender reports (25%) and the receivers reports (75%)
  - **Minimal Session Control Information**: Advanced control functions must be implemented in a higher level protocol.
- Types of RTCP packets:
  - Sender report packet,
  - Receiver report packet,
  - Source Description RTCP Packet,
  - Goodbye RTCP Packet and
  - Application Specific RTCP packets.
- RTCP itself does not provide any flow encryption or authentication. SRTCP protocol can be used for that purpose.

- Five RTCP packets
  - SR sender reports
    tx and rx statistics from active senders
  - RR receiver reports
    rx statistics from other participants, or from
    active senders if more than 31 sources
  - SDES source description, e.g. name (including CNAME), email-address, telephone number and address of the owner or controller of the source
  - BYE explicit leave
  - APP application specific extensions

RTCP packets

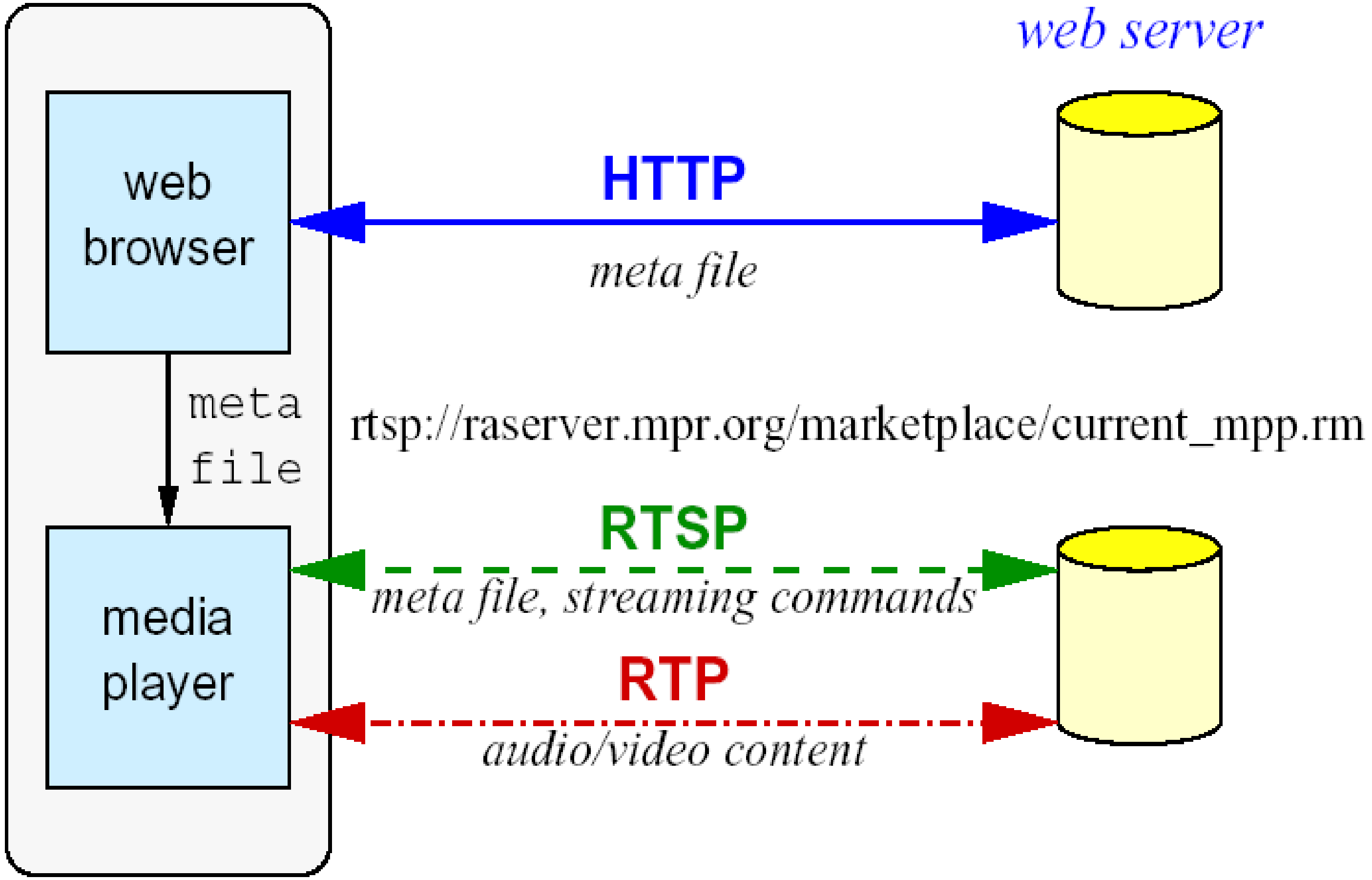
- SR packet includes
  - SSRC of sender - identify source of data
  - NTP timestamp when report was sent
  - RTP timestamp corresponding RTP time
  - packet count - total number sent
  - octet count - total number sent
  - followed by zero or more receiver report…
  - example:

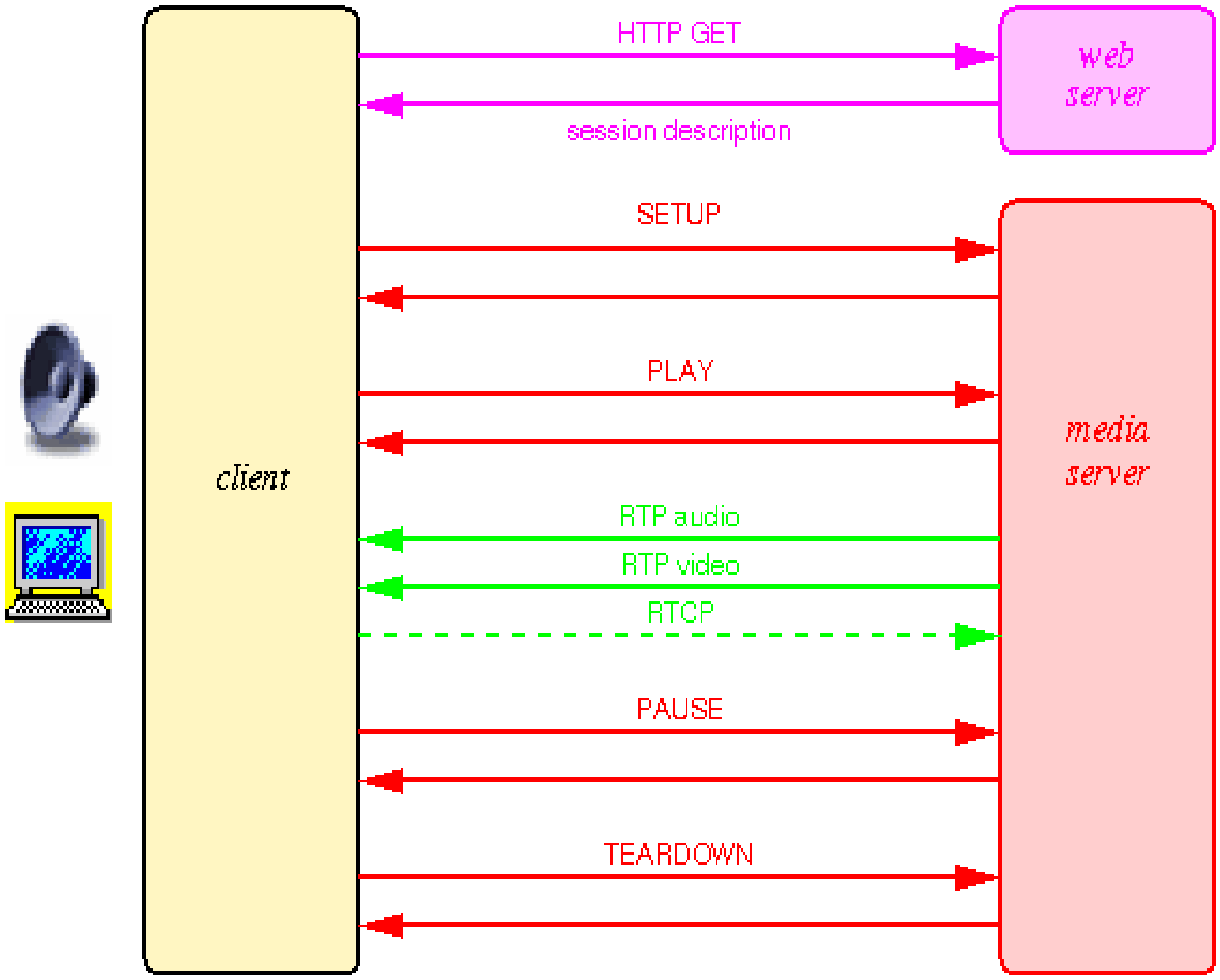


- RR packet includes
  - SSRC of source - identify the source to which this RR block pertains
  - fraction lost since previous RR (SR) sent (=int(256*lost/expected))
  - Cumulative # of packets lost long term loss
  - highest seq # received compare losses
  - interarrival jitter smoothed interpacket distortion
  - LSR time when last SR heard
  - DLSR delay since last SR

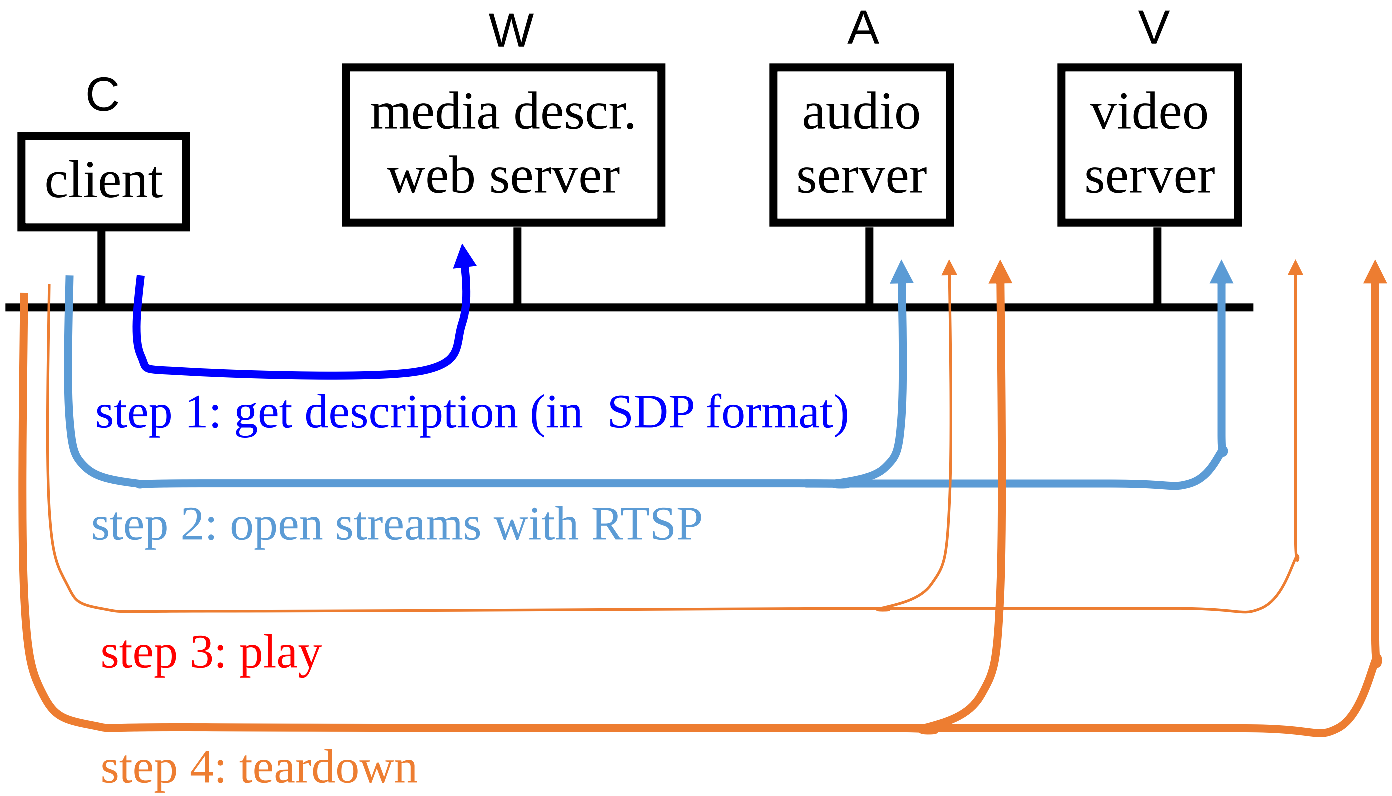
Real Time Streaming Protocol (RTSP)

- Supports VCR-like control operations
  - User controls operations like rewind, fast forward, pause, resume, etc…
- Out-of-band protocol (uses two connections, one for control messages (Port 554) and one for media stream)
- RFC 2326 permits use of either TCP or UDP for the control messages connection, sometimes called the RTSP Channel
- Meta file is communicated to web browser which then launches the Player
- Player sets up an RTSP connection for control messages in addition to the connection for the streaming media
  - Retrieves requested media.
  - Adds media to an existing session.


web server
web
browser
HTTP
meta file
meta
file
rtsp://raserver.mpr.org/marketplace/current_mpp.rm
RTSP
meta file, streaming commands
media
player
RTP
audio/video content


HTTP GET
web server
session description
SETUP
PLAY
media server
client
RTP audio
RTP video
RTCP
PAUSE
TEARDOWN

Another view



RTSP Protocol design

- text-based protocol
- transport protocol independent
- supports any session description (sdp, xml, etc.)
- similar design as HTTP with some differences
- client → server and server → client requests
- server maintains a « session state »
- data carried out-of-band
- works either with unicast or multicast

RTSP Methods

- Major methods
  - SETUP: server allocates resources for a stream and starts an RTSP session
  - PLAY: starts data tx on a stream
  - PAUSE: temporarily halts a stream
  - TEARDOWN: free resources of the stream, no RTSP session on server any more
- Additional methods
  - OPTIONS: get available methods
  - ANNOUNCE: change description of media object
  - DESCRIBE: get low level description of media object
  - RECORD: server starts recording a stream
  - REDIRECT: redirect client to new server
  - SET_PARAMETER: device or encoding control

Session Description Protocol (SDP)

- Text format for describing multimedia sessions
- Not really a protocol (similar to markup language like HTML)
- Can be carried in any protocol, e.g., RTSP or SIP
- Describes unicast and multicast sessions
- There are five terms related to multimedia session description:
  - Conference: set of two or more communicating users along with the software they are using.
  - Session : multimedia sender and receiver and the flowing stream of data.
  - Session Announcement: a mechanism by which a session description is conveyed to users in a proactive fashion
  - Session Advertisement : same as session announcement
  - Session Description : A well defined format for conveying sufficient information to discover and participate in a multimedia session.

Voice over IP (VoIP)

- Internet telephony - Requirements
  - ability of one party to signal to other party to initiate a new call
  - association between a number of participants
  - name translations and user location
    - mapping between names of different levels of abstraction
    - E.g. email address to IP address of host
  - feature negotiation
    - group of end systems must agree on what media to exchange and their respective parameters
    - E.g. different encodings, rates
  - call Participant Management
    - invite participants to existing call, transfer call and hold other users
  - Feature change
    - adjust composition of media sessions during the course of call
      - add or reduce functionality
      - impose or remove constraints due to addition or removal of participants

- Two signaling protocols:
  - **SIP** (IETF Standard) - Simple, cheap. Limited, but popular
  - **H.323** (ITU Standard) - set of protocols

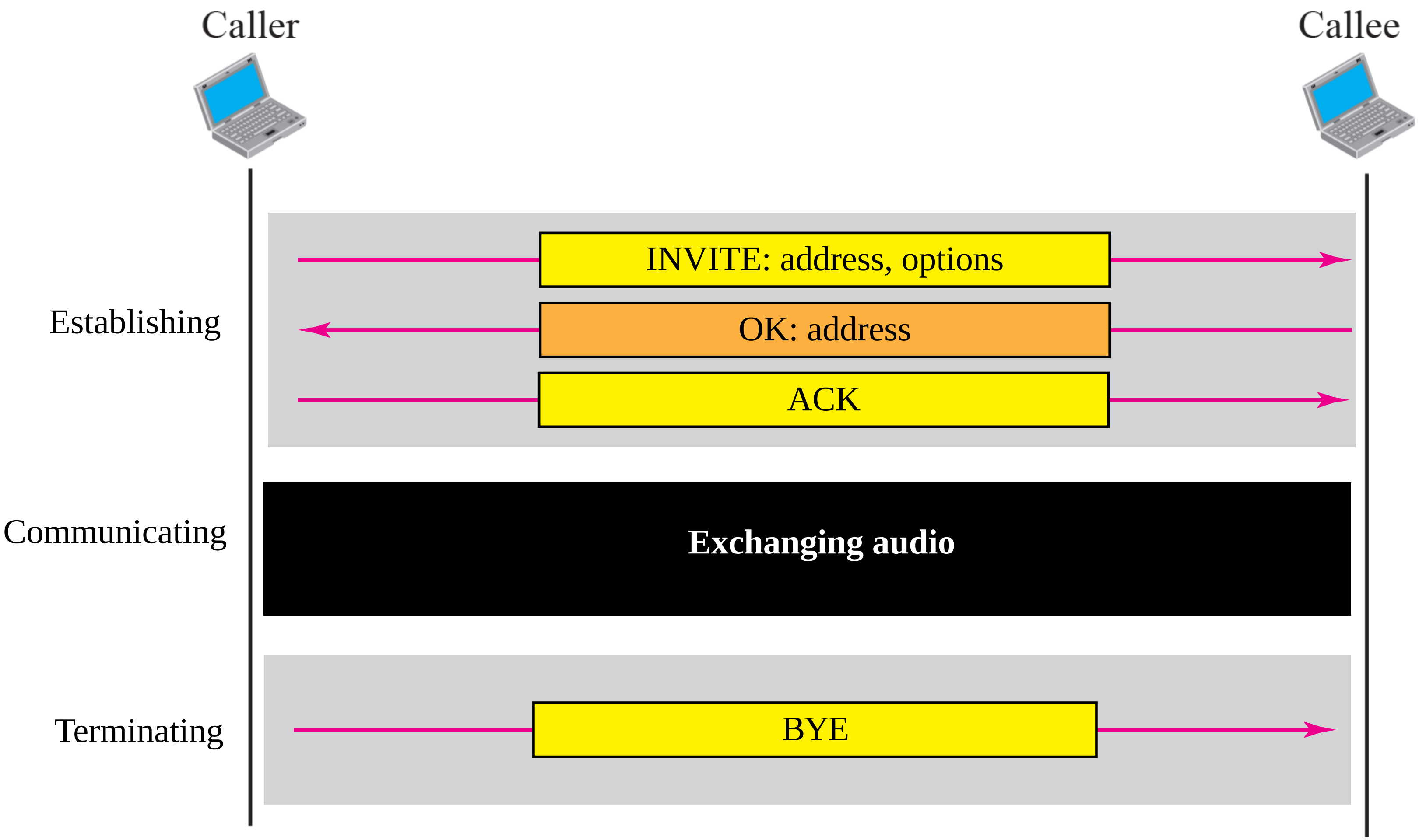
SIP (Session Initiation Protocol)

- Goal: inviting new participants to call
- Client-Server protocol at the application layer
- SIP requests can traverse many proxy servers
- Server may act as redirect server
- Proxies or redirect servers cannot accept/reject requests, only user agent server can
- Requests/Responses are textual

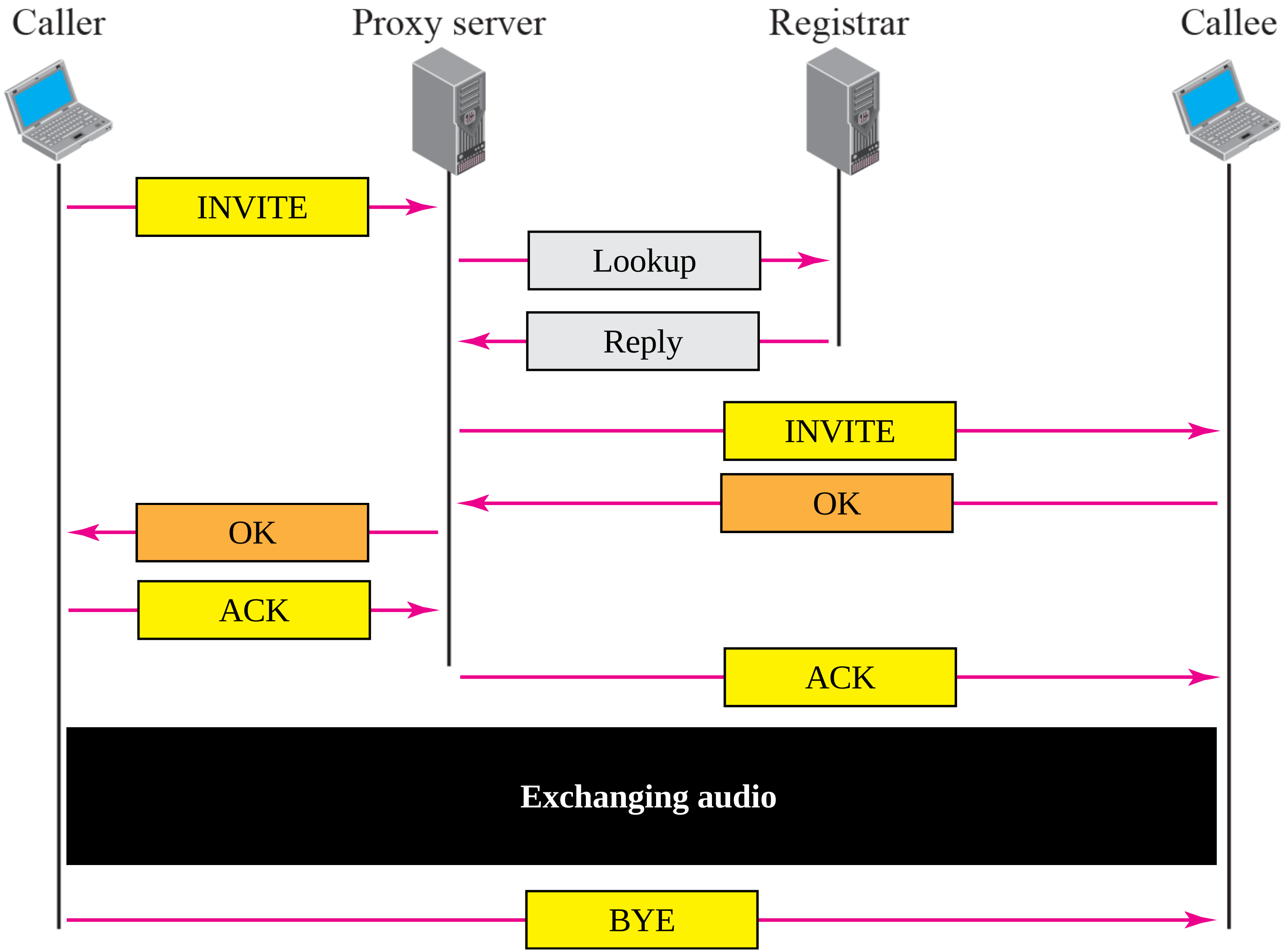
SIP (Session Initiation Protocol)

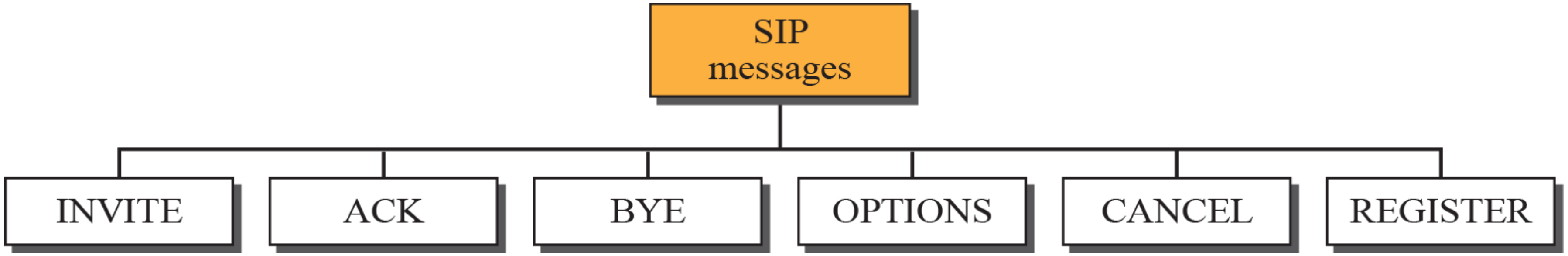
- Calls have unique call ID (carried in Call-ID header field of SIP message)
  - created by the caller and used by all participants
- SIP chooses email-like identifier
  - user@domain
  - user@host
  - user@IPaddress
  - phone-number@gateway

A Simple Session of SIP



Tracking the callee





INVITE—Indicates a user is being invited to participate in a call session.

ACK—Confirms that the user has received a final response to an INVITE request.

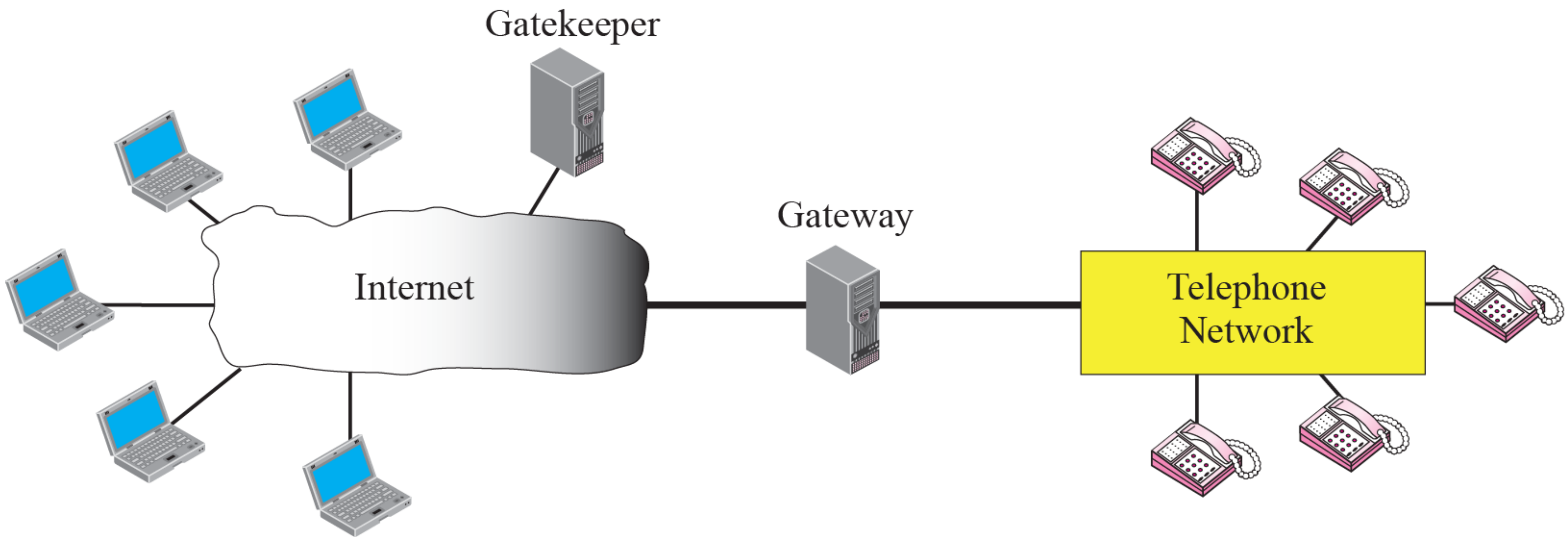
BYE—Terminates a call and can be sent by either the caller or the callee.

CANCEL—Cancels any pending searches but does not terminate a call that has already been accepted.

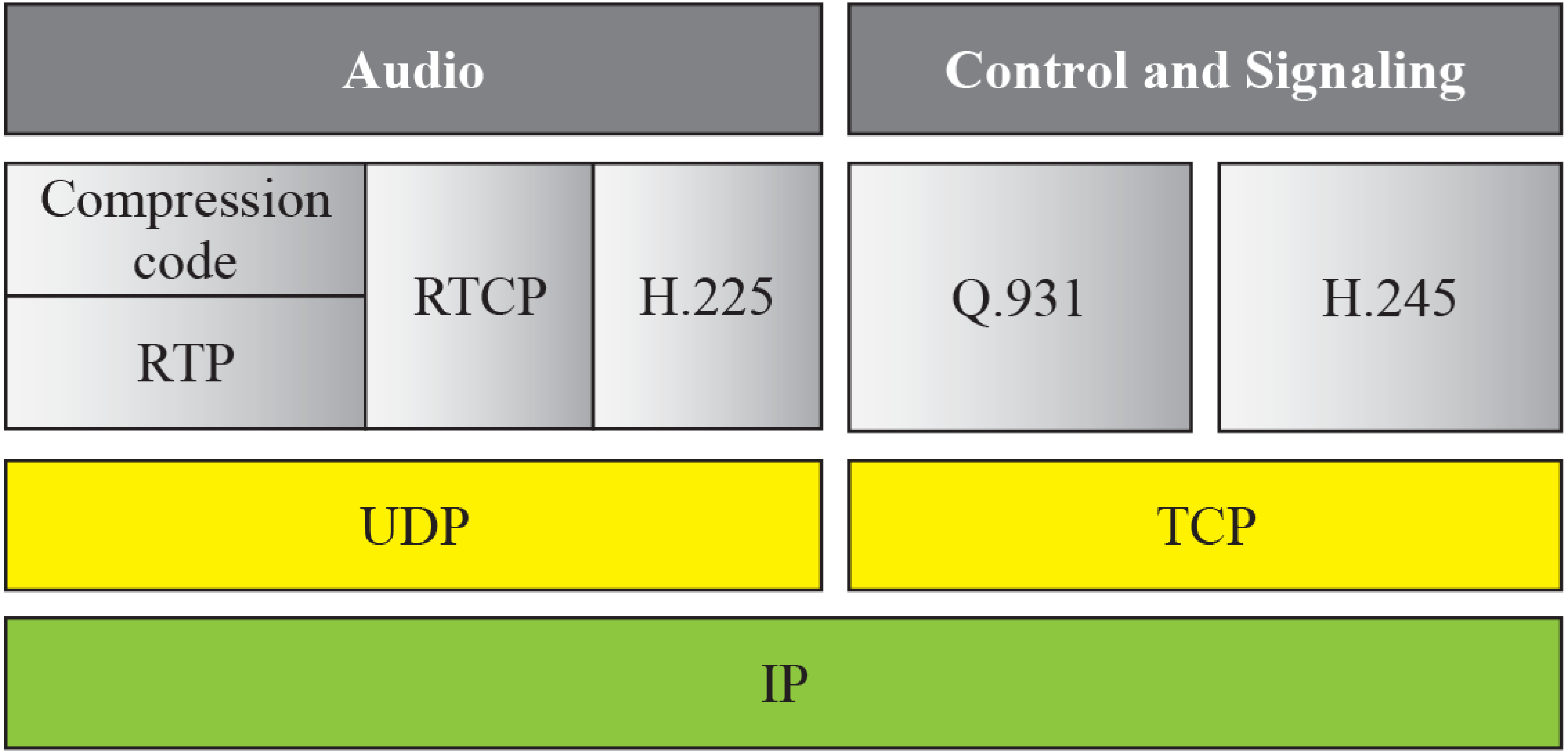
OPTIONS—Queries the capabilities of servers.

REGISTER—Registers the address listed in the To header field with a SIP server.

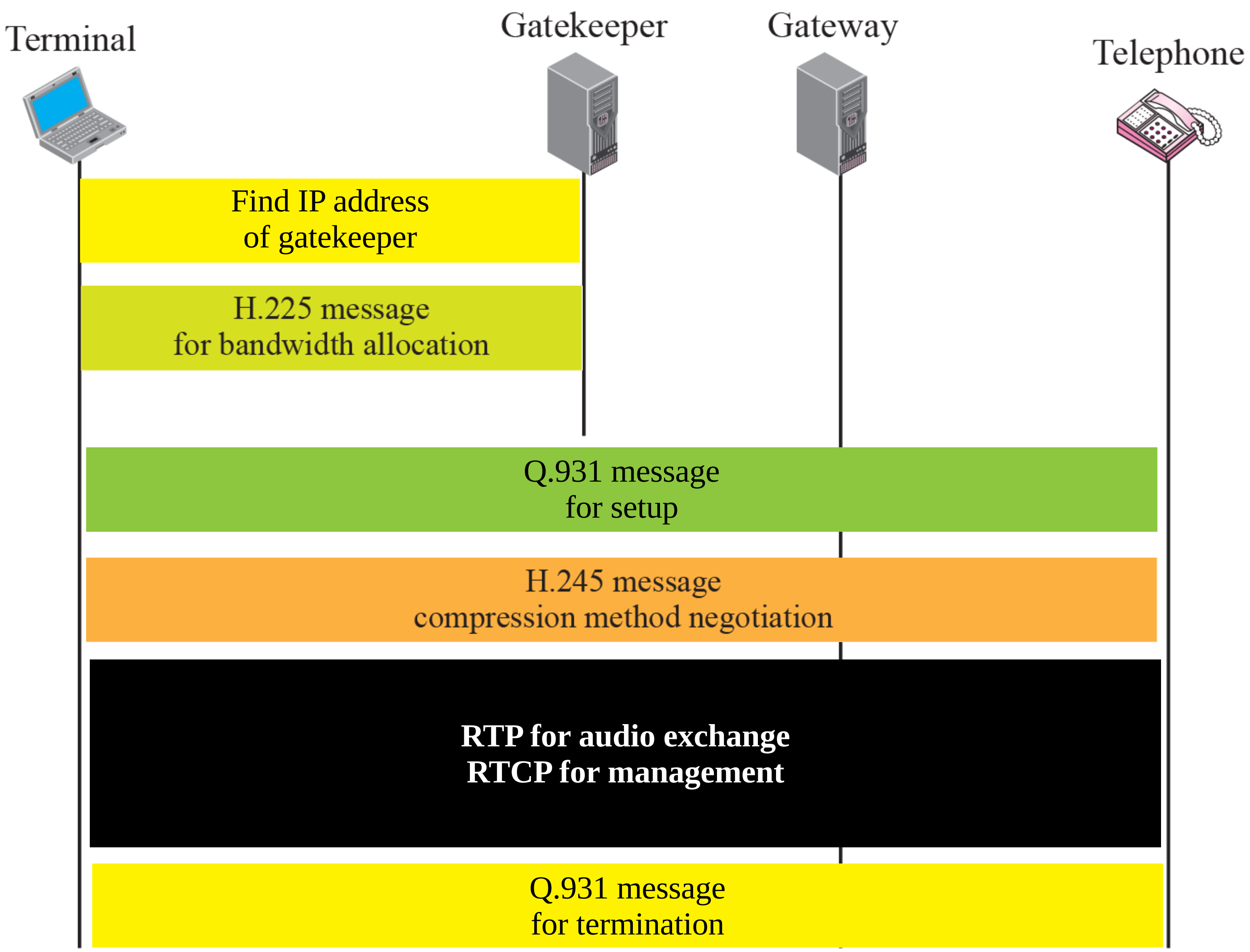
H.323 Architecture



H.323 Protocol Stack



- H.323 uses a logiclal channel on the LAN
- RAS (Registration, admission and status) – H.225
  - Gatekeeper Discovery
  - Endpoint registration
  - Call management
  - Admission procedures
  - and several more


Terminal
Gatekeeper
Gateway
Telephone
Find IP address
of gatekeeper
H.225 message
for bandwidth allocation
Q.931 message
for setup
H.245 message
compression method negotiation
RTP for audio exchange
RTCP for management
Q.931 message
for termination

- The terminal sends a broadcast message to gatekeeper. The gatekeeper responds with its IP address
- The terminal and gatekeeper communicate, using H.225 to negotiate bandwidth.
- The terminal, the gatekeeper, gateway and the telephone communicate using Q.931 to set up a connection.
- The terminal, the gatekeeper, gateway and the telephone communicate using H.245 to negotiate the compression method.
- The terminal, gateway and the telephone exchange audio using RTP under the control of RTCP.
- The terminal, the gatekeeper, gateway and the telephone communicate using Q.931 to terminate a connection.

- Before puja - 67, 02, 44, 26, 46, 06, 16, 20, 25, 32, 35
- 2/11/18 – 25, 2, 67, 61, 60, 55, 6, 44, 9, 35, 14, 16, 17