L75 - Voice over IP

Voice over IP

Voice over IP (VoIP)

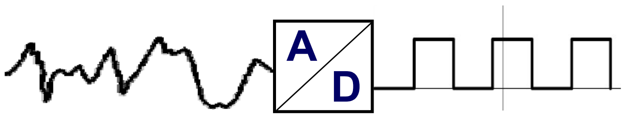
Voice Fundamentals
VoIP Fundamentals
RTP, SIP, H.323

Agenda

- **Digitized Voice**
- **Introduction to Voice over IP**
- **RTP**
- **SIP Basics 1**
- **SIP Basics 2**
- **SIP in Detail**
- **H.323**

Voice over IP (VoIP)

- **VoIP begins with digital voice**



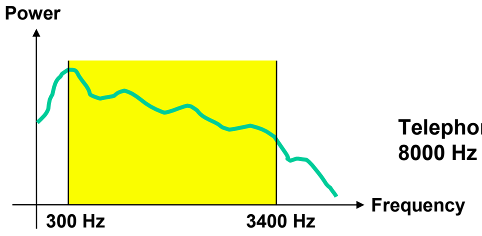
- **Analog-to-digital conversion**
  - speech sampling (8kHz, 16kHz)
  - 64 kbit/s speech
- **Removing redundancies from sample stream**
  - compression techniques/characterization of compressed speech
- **Extracting inactive periods**
  - silence/activity detection

Voice Transmission

- **Digital voice transmission**
  - based on Nyquist´s Theorem
  - analogous voice can be digitized using pulse-code-modulation (PCM) technique requiring a 64kbit/s digital channel
    - voice is sampled every 125usec (8000 times per second)
    - every sample is encoded in 8 bits
  - used nowadays in the backbone of our telephone network
  - today analogous transmission only between home and local office -> so called local loop
- **Synchronous TDM Techniques (e.g. PDH, SDH)**
  - originated from digital voice transmission

Sampling of Voice

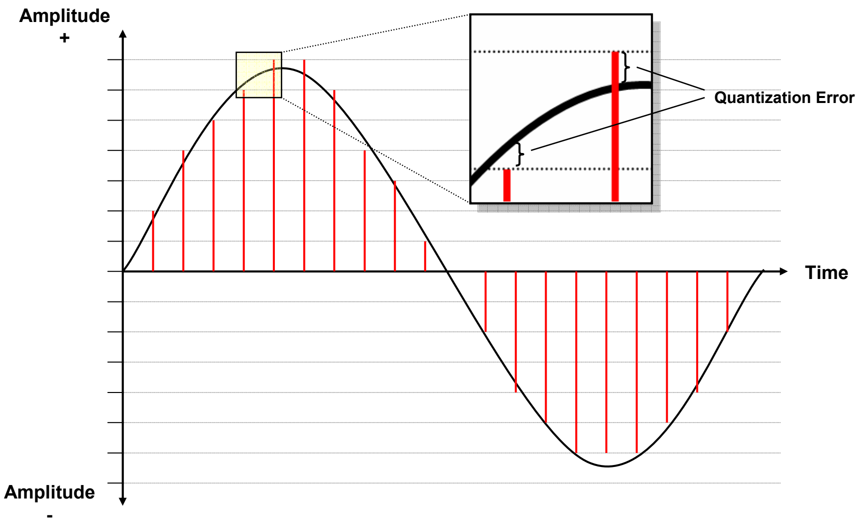
- **Nyquist´s Theorem**
  - any analogue signal with limited bandwidth f_B can be sampled and reconstructed properly when the sampling frequency is $2 \cdot f_B$
  - transmission of sampling pulses allows reconstruction of original analogous signal
  - sampling pulses are quantized resulting in binary code word which is actually transmitted



$$R = 2 * B * \log_2 V$$

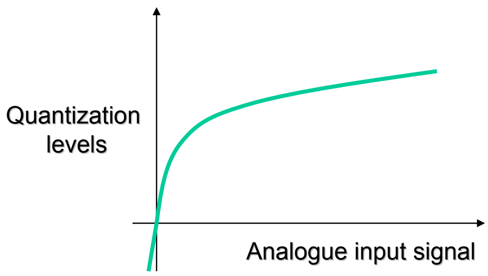
Telephone channel: 300-3400 Hz
8000 Hz x 8 bit resolution = 64 kbit/s

Linear Quantization

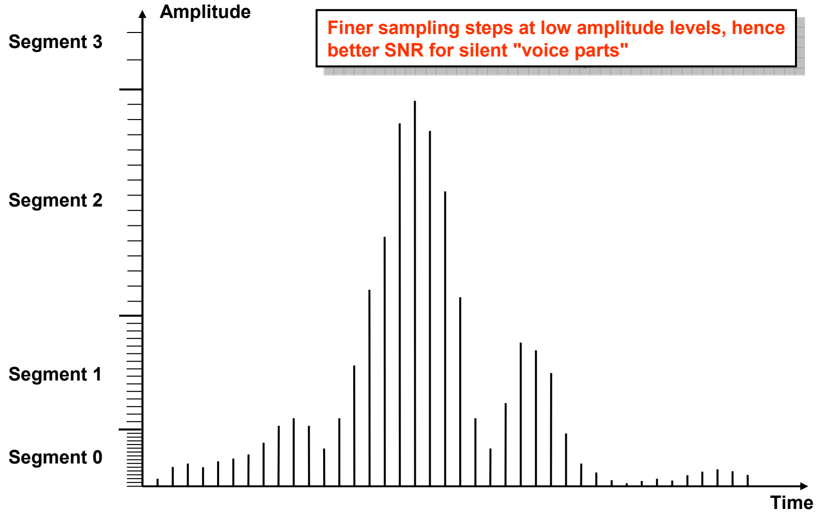


Improving SNR (Signal Noise Ratio)

- **To improve the SNR of speech signals**
  - lower amplitudes receive a finer resolution than greater amplitudes
- **A nonlinear function (logarithmic) is used for quantization**
  - USA: μ-law (Bell)
  - Europe: A-law (ITU)

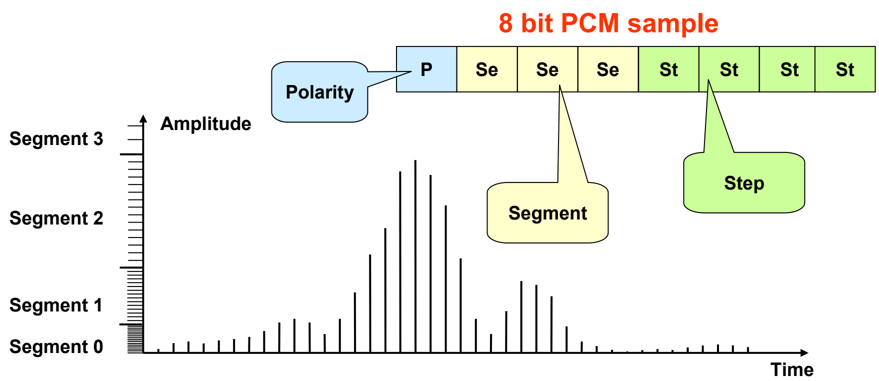


Logarithmic Quantization



Encoding (PCM)

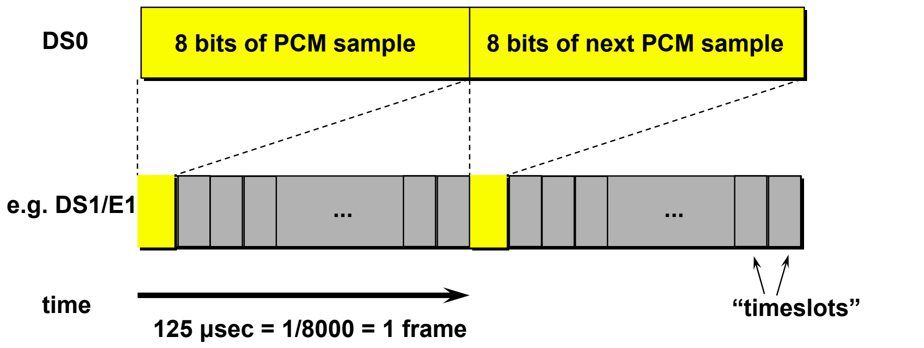
- Putting digital values in a defined form for transmission



Digital Voice Channel

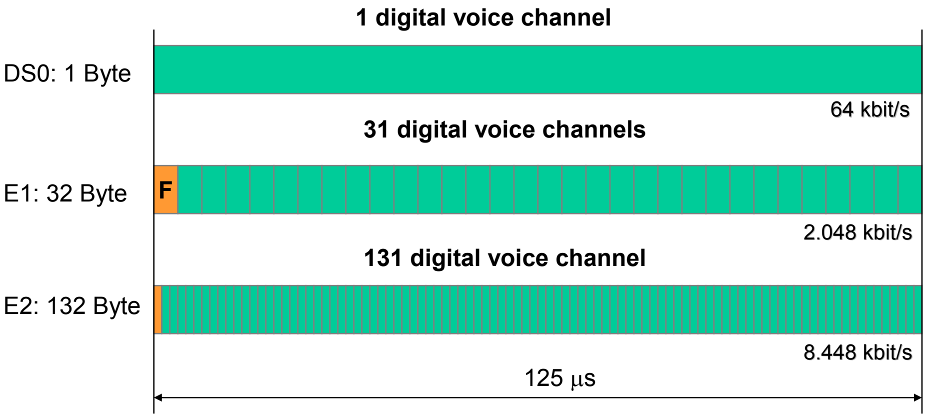
- **DS0 = Digital Signal, Level 0**
  - 1 timeslot in multiplexing frames
- **Base for hierarchical digital communication systems like PDH, SDH**
- **Equals one PCM coded voice channel**
  - 64 kbit/s
- **Each samples (byte) must arrive within 125 μs**
  - To receive 8000 samples (bytes) per second
  - Higher order frames must ensure the same byte-rate per user(!)

TDM Multiplexing Basics



- **frame rate is always 8000 frame per second at all levels of the hierarchy**
- **byte interleaved multiplexing**

Multiplexing Basics



– note: DS0 and higher rates can be used for any transport digital information -> <u>data transmission</u>

Classical Codec for PSTN

- **G.711 is the fundamental codec of legacy PSTN world**
  - Classical PCM (64 kbps)
  - Synchronous TDM hierarchy (PDH, SDH) was originally designed for that
  - Offers reference quality at uncompressed transmission like in ISDN networks but needs 64K transmission rate
  - Usable for VoIP for internal calls with optimal quality (e.g. Ethernet and L2 switching infrastructure)
- **In order to reduce bandwidth requirements**
  - Mathematical models are used to digitally encode (and compress) analog audio information
  - Voice compression
  - But they introduce some delay

Voice Compression

- **Waveform Coders**
  - Non-linear approximation of analog waveform
  - PCM (no compression), ADPCM (with compression)
- **Vocoders**
  - speech is analyzed and compared to a codebook
  - only codebook values are transmitted and speed synthesizer at the receiver
- **Hybrid coders**
  - Combination of waveform coders and vocoders
  - 4.8 kbps to 16 kbps
  - Used for mobile phones
  - CELP, GSM

Standardized Codec 1

- **Adaptive Differential Pulse Code Modulation (ADPCM)**
  - only the difference from one sample pulse to the next will be transmitted
  - fewer bits used for encoding the difference value
  - G.726 (16, 24, 32, 40 kbps)
- **Low Delay Code Excited Linear Predictor (LD-CELP)**
  - G.728 (16 kbps)
- **Conjugate Structure Algebraic Code Excited Linear Predictor (CS-ACELP)**
  - G.729 (8 kbps)

Standardized Codec 2

- **Dual Rate Speech Coding Standard G.723**
  - is the basic standard for voice transmission in IP networks
  - Basis is the CELP-Technique of GSM
  - Uses minimal data rate of 5,3K at fair quality or 6,3K with good quality
  - Very efficient signal processors needed for encoding
- **iLBC (Internet Low Bitrate Codec)**
  - well suited to sustaining reasonable quality on lossy network links

Codec Delays

- **Algorithmic delay**
  - Look-ahead delay (sample N+1) for sample N
  - G.723.1: 7.5ms
- **Coder delay**
  - Coding and compression delay
  - Can be significant and depend on DSP power and complexity
- **Decoding delay (~10% of coding delay)**
- **Packetization delay**
  - Two parts contributes to such a delay
  - 1) Function of sample block size required in order to start with the coding
  - 2) Number of blocks placed in a single frame to be transmitted

Codec

- Target: use bandwidth more efficient due to speech compression
- New encoding and decoding techniques were developed
- Bandwidth and speech quality depending standards from ITU

ITU Specification	Data rate (kbps)	Quality Needed	MIPS	Digitalization (ms)
G.711 PCM	64	Very good	< 1	0,25
G. 726 ADPCM	32	Good		
G.729 and G.729A CS-ACELP	8	Good	20	11,25
G.723.1 MP-MLQ MP-ACELP	6,3 5,3	Good Fair	18	67,5
G.728 LD-CELP	16	Good	30	1,25

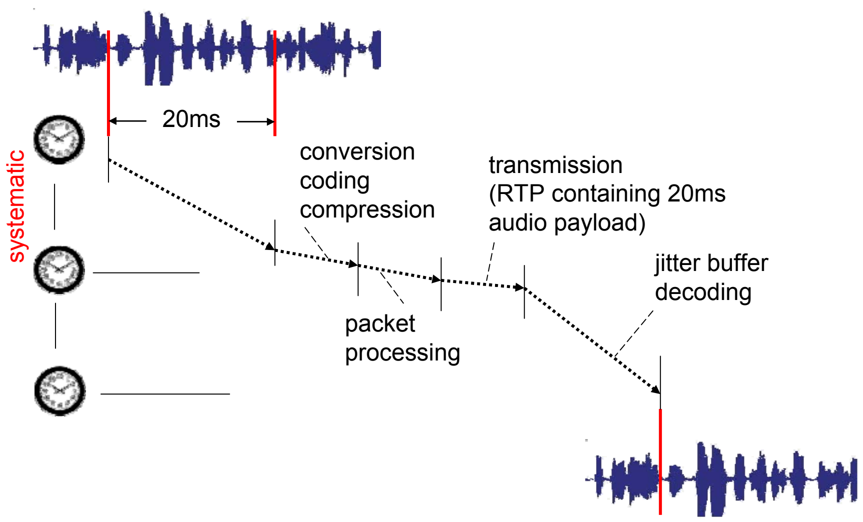
Codec Delay Details

Coder	Rate	Required Sample Block	Best case coder delay	Worst case coder delay	Algorithmic Delay
ADPCM, G.726	32.0 kbit/s	10ms	2.5ms	10ms	0ms
CS-ACELP, G.729	8.0 kbit/s	10ms	2.5ms	10ms	5.0ms
MP-MLQ, G.723.1	6.3 kbit/s	30ms	5.0ms	20ms	7.5ms
MP-ACELP, G.723.1	5.3 kbit/s	30ms	5.0ms	20ms	7.5ms

Delay Budget

- **Delay occurs on transmitting side, network and receiving side**
  - Delay on the transmitting side is due to the codec
  - In the network, delay stems from
    - Transmission (serialization and propagation)
    - Queuing
  - Delay on the receiving side is added by
    - Jitter buffer depth
    - Decoding and processing and audio device
- **ITU delay limits (one-way)**
  - 0-150ms ~ toll quality
  - 150-400ms ~ acceptable

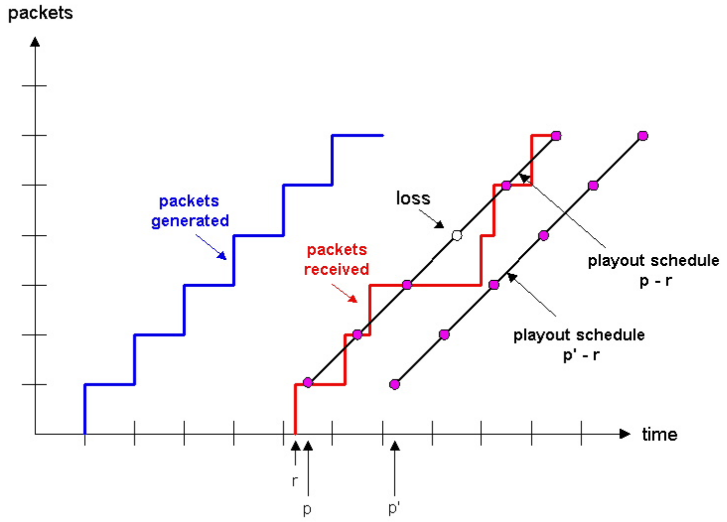
The VoIP home-made or systematic delay



Jitter

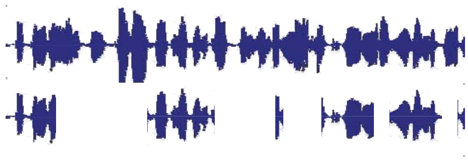
- **Speech is a constant bit-rate service (isochronal)**
  - Packets might have varying transmission time
  - Variable delays must be removed at the receiving end
- **Jitter-buffer transforms variable delay into constant delay**
  - Ensures smooth and continuous playback
  - Adds delay to the overall delay budget
- **Jitter buffer can be adaptive, but maximum delay is fixed**
  - E.g. derived from RTCP information

Jitter buffer … fixed play-out delay



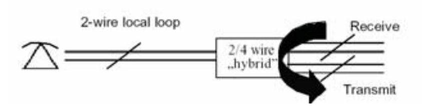
Packet Loss

- **Losses occur due to**
  - bit errors (no error correction in packet voice networks)
  - discarding packets at (i) intermediate nodes (ii) destination
- **Packet losses up to 10% are tolerable if**
  - losses occur at random time instants
  - packets (=speech segments) are relatively short (~10ms)
  - places of lost packets are „filled in"



Echo

- **Two types of echo can deteriorate speech quality**
  - Network echo and acoustic echo
  - if one-way delay exceeds 25ms
- **Network echo (impedance mismatch in PSTN hybrids)**



- **Acoustic echo**
  - Commonly in hands-free equipment
  - Loudspeaker's sound reflects back to the microphone
- **Canceling echo is essential to maintaining high quality**

QoS – Runtime Calculation

- IP Packet Segmentation
  - IP packet size depends on available data rate
  - Router might delay big packets
  - Fast gateways should have powerful processors to minimize computing time
  - Big throughput and efficient memory concepts

Example for runtime calculation:

Reason	**Runtime (ms)**
A-D-encoding	20
Packetizing	30
Other service times	10
Routing over 8000 km	50
Jitter buffering	30
D-A-Decoding	20
Total runtime	**160**

QoS – Jitter, packet losses or corruption

- **Jitters** are accidental oscillations of packet runtime from sender to receiver network
  - To guarantee RT-processing arriving packets have to be stored in jitter buffers from where they are read synchronously
  - Modern systems have a dynamic adaptable jitter buffer size
- **Packet losses or corruption**
  - <5 % are acceptable
  - >5 % make use of Forward Error Correction (FEC)
    - Intrapacket-FEC put additional bits into packets, to reconstruct defective packets
    - Extrapacket-FEC defect packets can be repaired with previous intact packets
    - Loss rate can be reduced until 10 to 20 % but often requires about 30 % more bandwidth.

QoS – Necessary Bandwidth

- Necessary bandwidth dependent on Codec used
- Typical full duplex telephone call uses just 36 to 40 % of capacity because most of the time of the conversation is pause.
- Silence suppression detects whenever it is not spoken on the line so the needed bandwidth can be reduced about 60 %
- Calculation of average Net-Bandwidth when half duplex:

G.723 Codec Bandwidth	6,3K
IP-Header, compressed	2,0K
Total Bandwidth	8,3K
minus• 60% inactivity	-5,0K
Netto-Bandwidth total	**3,3K**

QoS – Fault time

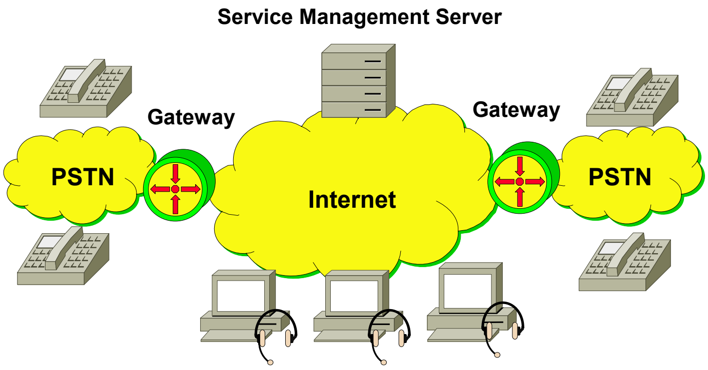
- **Fault time**
  - Reliability of network is essential for commercial use
    - Reliability of 99, 9998 % => 5 minutes fault per year
    - Within LAN 99,8 % realistic => 18 hours per year
    - WAN like Internet only 98 % => fault of 1 week per year!
  - Reliability of network components:
    - Clients: often have troubles with software => better use PC independent IP-telephones
    - Hardware failure of server-components are quite rare due to the redundancy and good type of construction
    - Software-server-problems detected with monitoring systems observation systems

Agenda

- Digitized Voice
- <u>Introduction to Voice over IP</u>
- RTP
- SIP Basics 1
- SIP Basics 2
- SIP in Detail
- H.323

 VoIP, v1.1 31

Elements and Scenarios of VoIP



Elements: PCs, conventional telephones, IP-telephones
Scenarios: PC-to-PC, Phone-to-Phone, PC-to-Phone, Phone-to-PC
PSTN = public switched telephone network

 VoIP, v1.1 32

VoIP Clients, Gateways, Servers

- **Clients**
  - User-interface
  - To call or end a call
  - Analogue – digital encoding of speech
  - IP-data packetizing
  - Decode from digital into analogue speech

  2 Types:
  - Clients: software-clients or IP-telephones
  - Virtual clients: provided by gateway, interface for conventional telecommunication equipment like telephones, fax etc.

- **VoIP-Gateway**
  - Bridge between conventional and IP telephony
  - Allows both users to communicate with their different equipment

- **Server**
  - IP-telephony management and control
  - Management of connection requirements of connection and exchange processes like
  - Call forwarding, conference calls, user administration of their profiles and access rights, call tracking, billing, answering machine, voice-mail function

Tasks of a VoIP Gateway

- **Task 1: matching telephone number - IP-address**

  whenever you call from a conventional phone over the VoIP gateway, the server has to convert the wanted phone number into an IP-address of the remote gateway of the call receiver with a database lookup.
  - peripheral database => directly implemented in gateway performance advantages in speed
  - Central database on a server where all gateways have access bigger latency but no database replication is needed

- **Task 2: Connection establishment**
  - Gateway is contacting the remote-gateway and exchanging security, encoding, capacity and setup information until connection is established

Tasks of a VoIP Gateway

- **Task 3: digitalization and compression**
  - Analogue speech signals have to be digitized before compression.
  - Common technique: 64K Pulse Code Modulation (PCM)
  - ISDN channels can be easily connected because they are already 64K PCM encoded and can be bridged. Compression into one of several codec-formats is done by the Digital Signal Processor (DSP)
- **Task 4: Packetizing and packet delivery**
  - Wrap data into IP packets and dispatch them via UDP and TCP
  - Advantage of UDP: no error detection and recovery => faster, more efficient, retransmission of speech wouldn't make sense => delay

 VoIP, v1.1 35

Current Problems of Internet Telephony 1

- **Standards**
  - interoperability between Internet telephony products and PSTN-based systems and services
  - Users have to have the same kind of software
- **Quality**
  - Voice performance is measured by delay
  - Calls on PSTN have about 50-70 msec delay
  - On internet there is an increased latency of ~ 500 msec
  - But human latency tolerance is only ~ 250 msec
  - Today's products exceed it so it sound like calls routed over a satellite circuit

 VoIP, v1.1 36

Current Problems of Internet telephony 2

- **Capacity**
  - Packet loss occur because of network congestion due to
    - bandwidth limitation
    - traffic overload → transmission delays and packet discards
    - Error performance → inadequate network access links cause bandwidth congestion (very bad on transcontinental links)
    - applications repair lost packets with silence → speech clipping effects → Even the loss of an individual packet has an impact on speech due to the large packet size.

- **Social issues**
    - Traditional telephone providers (often monopolies) are against Internet-based providers because they have an "unfair" advantage in offering cut-rate long distance phone service.
    - Conclusion: It is very political!

Solution to Current Problems 1

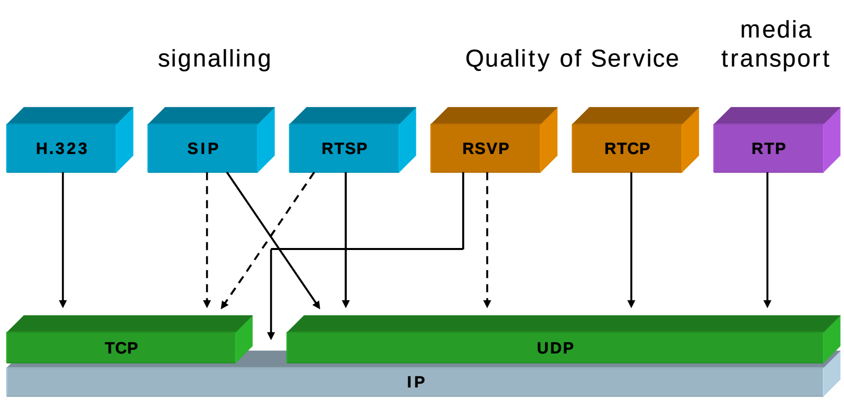
- **Standards**
  - ITU H.323 recommendation for VoN applications
  - Improved voice compression codecs
  - T.120 for data conferencing
  - RTP (Real Time Protocol)
  - RTCP (Real Time Control Protocol)
  - IP QoS (IntSrv with Resource Reservation Protocol or DiffSrv with DSCP)
  - SIP (Session Initiation Protocol)

- **Quality improvements**
  - protocol improvements (IP QoS) and codec improvements
  - bigger routers (gigabit routers)
  - new network architectures and better links

Solution to Current Problems 2

- **Capacity**
  - average hop number of trans-Atlantic call is 20 to 30
  - delay increase with every router hop
    - increase routing speed
    - more routers
    - bigger routers (gigabit router)
      - handle at least 10 times more traffic than conventional router
      - per-packet cost of gigabit routing is 3-4 times less than traditional routing
- **Social issues**
  - encourage new technology
  - trend of technology development
    - more than 100 well known companies are involved

VoIP Protocols - Overview



Agenda

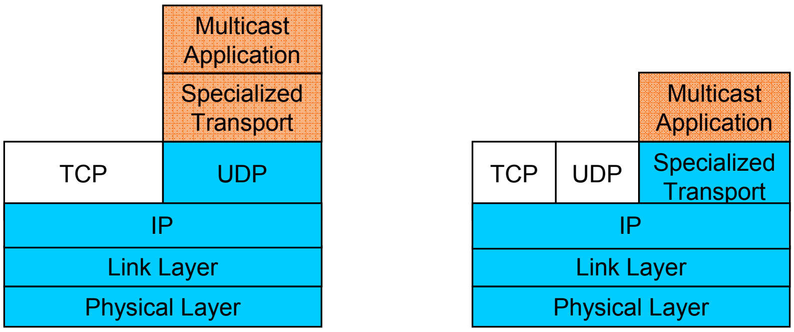
- Digitized Voice
- Introduction to Voice over IP
- RTP
- SIP Basics 1
- SIP Basics 2
- SIP in Detail
- H.323

Real-time (Multicast) Applications

- TCP?
  - Real-time multicast applications must run on top of UDP or interface directly to IP providing their own transport layer
    - TCP is a unicast (point-point) only transport protocol
    - with TCP reliability and flow control mechanisms have not been optimized for real-time broadcasting of multimedia data
    - the potential to lose a small percentage of packets is preferred to the transmission delays introduced with TCP
    - hence multimedia streaming applications need a specialized transport layer
    - such as the Real-Time Transport Protocol RTP which operates over UDP in the application layer with the application

Operation over UDP or IP

– Multicast (real-time) applications must run on top of UDP (e.g. RTP; left picture) or interface directly to IP providing their own customized transport layer (right picture)



Real-time Applications based on RTP/RTPC

- **Well known MBone multicast applications**
  – VAT, VIC, WB, SDR
- **Other famous applications**
  – Quick Time (Apple)
    - provides digital video and media streaming
  – Real Audio and Real Video (RealNetworks)
    - high quality audio and video streaming
  – NetMeeting (Microsoft)
    - provides IP telephony, white boarding, text chats and application and file sharing
  – CU-seeMe (CUseeMe Networks)
    - Internet video chat software supporting video, audio, text and whiteboard communications
  – IP/TV (Cisco Systems)
    - Live video, scheduled video, and video on demand

Real-time Transport

- **Audio/Video are continuous media**
- **Packet networks transport discrete units**
  - Digitize media
  - Compression
  - Packetization
- **No additional multiplexing (beyond UDP/IP) is needed**
  - Transport different media in different packets
  - Can give different CoS (DSCP) to different media
- **Little help from transport protocol is needed**

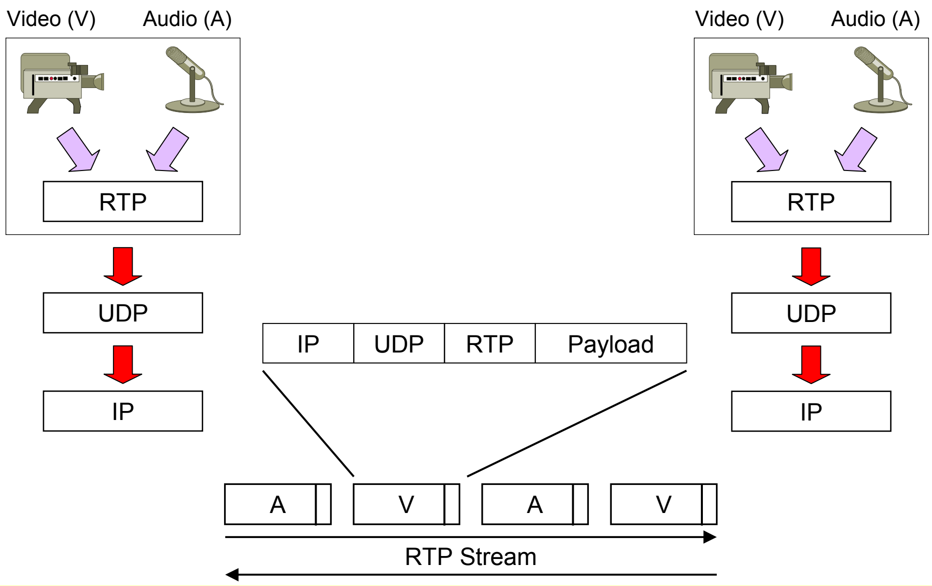
RTP and RTCP Overview

- **RTP = Real Time Transport Protocol**
  - Makes transport of time critical data in IP-networks possible
  - Gives every IP-packet a time stamp with creation time and following number to assemble the packets synchronous in the right order
  - End-to-End service for real time data
  - Unicast and multicast transmissions
  - Allows the protocol to easily adapt to new audio and video standards
- **RTCP = Real Time Control Protocol**
  - Coordinates sender and receiver protocols
  - Provide management and monitoring of real time connections

RTP

- **RTP = Real Time Transport Protocol**
  - Implements the transport features needed to provide synchronization of multimedia data streams
    - RTP may be used to mark the packets associated with the individual video and audio streams
    - Allows the streams to be synchronized at the receiving host
    - Next slide shows the operation of RTP in a multimedia transmission
    - Audio and video data are encapsulated in RTP packets
    - If the multimedia application does not utilize RTP services, the receiver may not be able to associate the corresponding audio and video packets
    - Congestion or other conditions within the network can cause packets to be lost or reordered during transit

RTP



RTP

- **RTP (cont.)**
  - This behavior causes quality problems with typical multimedia applications
  - RTP protocol alone does not include any mechanism to provide guaranteed delivery or other quality of service functions
  - Standard does not prevent out of sequence packet delivery nor does it assume that the underlying network is reliable and delivers packets in sequence
  - It also does not prevent the occurrence of network congestion
  - Designers of applications must determine if these levels of service are acceptable

RTP Header Format

0		4		8		16	31
V	P	X	CSRC	M	Payload Type	Sequence Number	
Timestamp							
Synchronization Source (SSRC) Identifier							
Contributing Source (CSRC) Identifiers							

- First 12 octets are required in every RTP packet
- V: Indicates the RTP version
- P: Contains the padding bit, used by encryption algorithms (bit is set)
- X: If this field is set a header extension follows the fixed header
- CSRC Count: This field contains the number of contributing source identifiers that follow the fixed header
- M: This field allows significant events to be marked in the packet stream (frame boundaries)

RTP Header Format (cont.)

0	4			8	16	31
V	P	X	CSRC	M	Payload Type	Sequence Number
Timestamp						
Synchronization Source (SSRC) Identifier						
Contributing Source (CSRC) Identifiers						

- SSRC identifier: All packets from the same source contain the same SSRC identifier
- This enables the receiver to group packets for playback
- CSRC identifiers: Contains a list of the sources for the payload in the current packet
- This field is used when a mixer combines different streams of packets (see later in this chapter)

RTP Header Format (cont.)

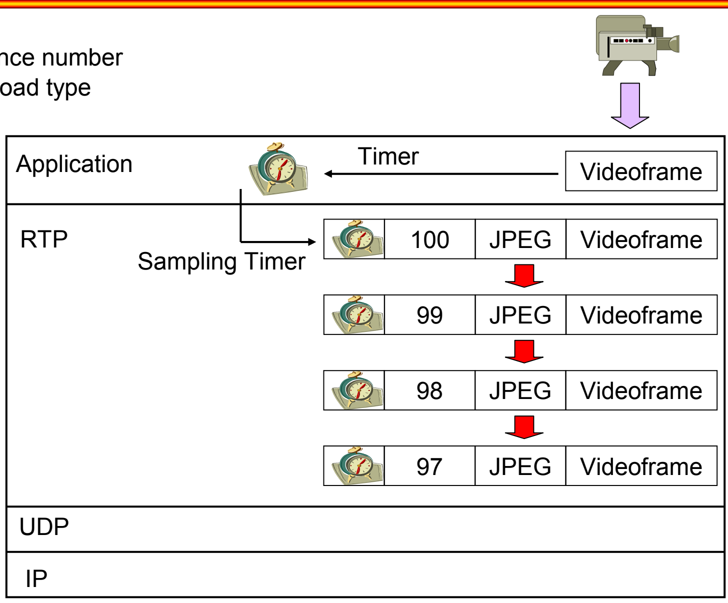
- **RTP protocol services**
  - RTP provides end to end transport services for applications transmitting real-time data
  - Included in the RTP header
  - Payload type identification
    - A RTP packet can contain portions of either audio or video data streams
    - To differentiate between these streams, the sending application includes a payload type identifier within the RTP header
    - Identifier indicates the specific encoding scheme used to create the payload
    - Receiving application uses this identifier to determine the appropriate decoding algorithm

RTP Header Format (cont.)

- **RTP protocol services (cont.)**
  - Sequence numbering
    - Sequence numbers are used by the receiving RTP host to restore the original packet order
    - The receiver is able to detect packet loss using the information in this field
  - Timestamping
    - Time stamps are used in RTP to synchronize packets from different sources
    - Timestamp represents the sampling (creation) time of the first octet in the RTP data packet
    - It is possible that several RTP packets may have the same time stamp
    - For example this can occur when a single video frame is transmitted in multiple RTP packets

RTP Time Stamping

100…sequence number
JPEG…payload type



RTCP

- **RTCP = Real Time Control Protocol**
  - To manage real-time delivery many applications require feedback about the current performance of the network
    - Primary function of RTCP is to provide feedback about the quality of RTP data distribution
    - RTCP is based on periodic transmission of control packets to all participants in a session
    - RTCP uses a separate UDP connection for communication
  - RTCP architecture defines five types of control information used to report current performance

 VoIP, v1.1 55

RTCP

- **Types of RTCP control information (cont.)**
  - Sender report:
    - Sent out by the source of an RTP data stream (in intervals)
    - Provides the transmission and reception statistics observed by the sender
    - Is sent as a multicast packet processed by all RTP session participants
  - Receiver report:
    - Provides reception statistics for participants that are not active senders
    - Is issued if the interval times out and no data flows
  - Source description report:
    - used by an RTP sender to provide local capability information

 VoIP, v1.1 56

RTP Translators and Mixers

- **RTP protocol supports the use of translators and mixers to modify the RTP packet stream**
  - These devices are used when some participants in a multimedia session need to receive data in different formats
- **RTP translators**
  - Used to change the type of data in an RTP packet
  - In the following example, three videoconferencing workstations are exchanging MPEG traffic over a high-speed LAN
  - Each workstation is generating MPEG data (rate 1.5 Mbps)
  - Another workstation connected via a lower-speed serial connection wishes to participate in the videoconference

 VoIP, v1.1 57

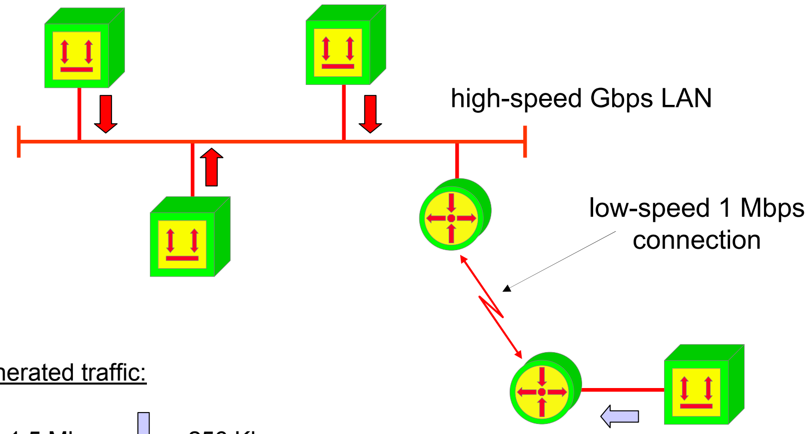
RTP Translators and Mixers

- **RTP translators (cont.)**
  - Bandwidth of this connection is not sufficient to support the video streams
  - One possible solution for this problem is changing all workstations to a video format, producing less traffic (e.g., H.261 with 256 Kbps)
  - But reducing data rate means reducing quality of video
  - An alternate solution uses RTP translation devices
  - Each individual MPEG video stream is converted to an H.261 video stream with 256 Kbps which can be forwarded through the serial line
  - The three LAN attached workstations continue to use the higher quality MPEG format

 VoIP, v1.1 58

Videoconference without Translating

Videoconference with 4 workstations
(only communication LAN -> serial link)

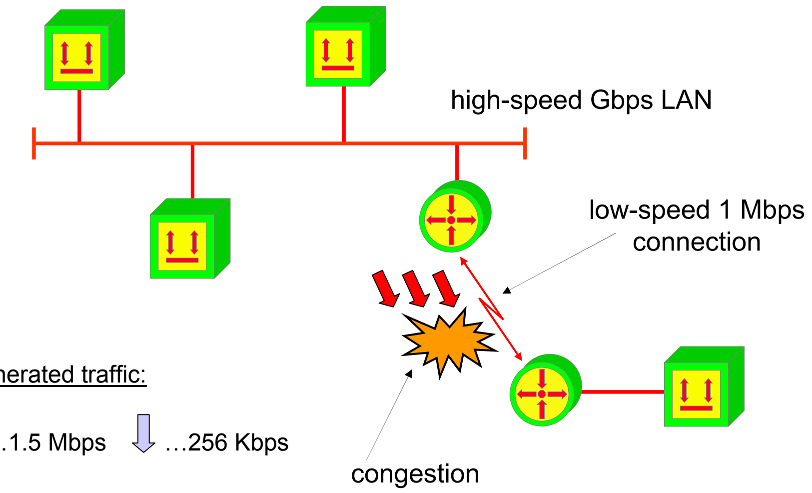


generated traffic:

...1.5 Mbps ...256 Kbps

Videoconference without Translating

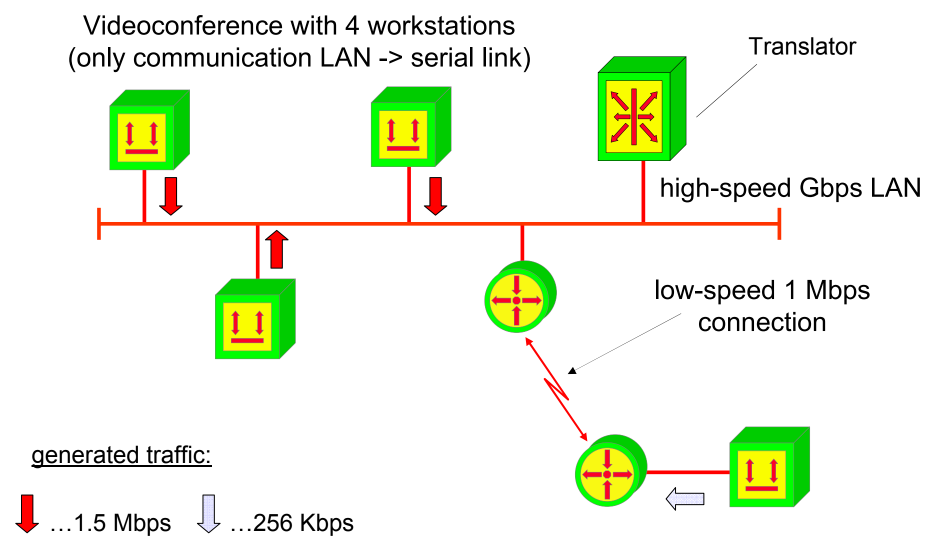
Videoconference with 4 workstations
(only communication LAN -> serial link)



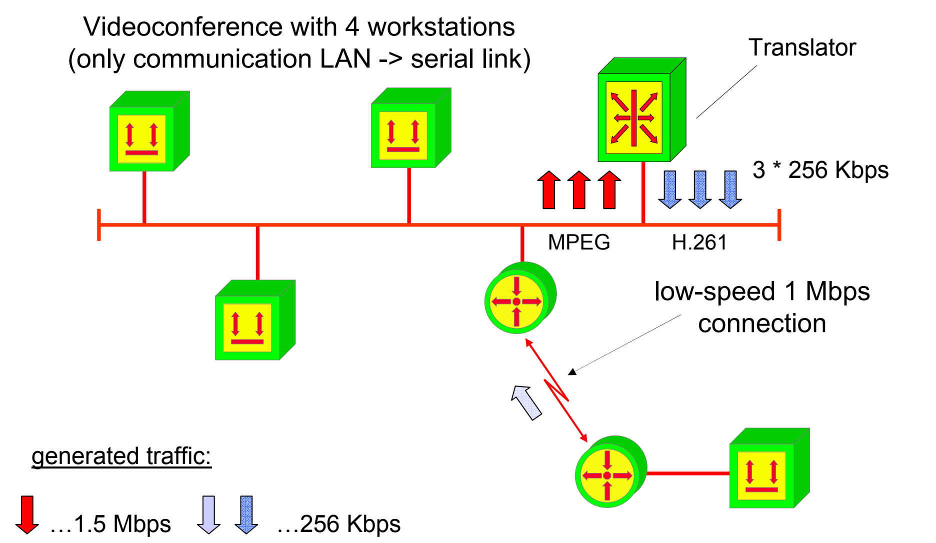
generated traffic:

...1.5 Mbps ...256 Kbps

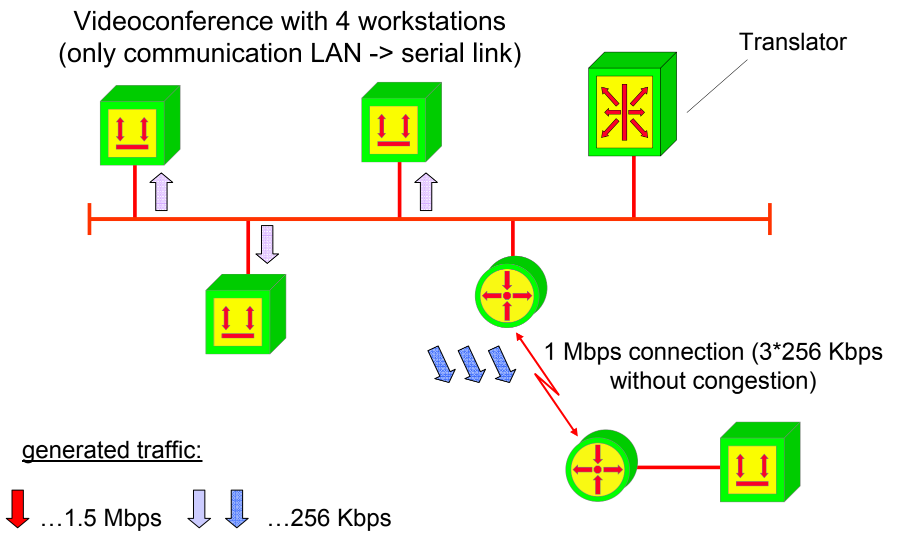
Videoconference with RTP Translating



Videoconference with RTP Translating



Videoconference with RTP Translating



RTP Translators and Mixers

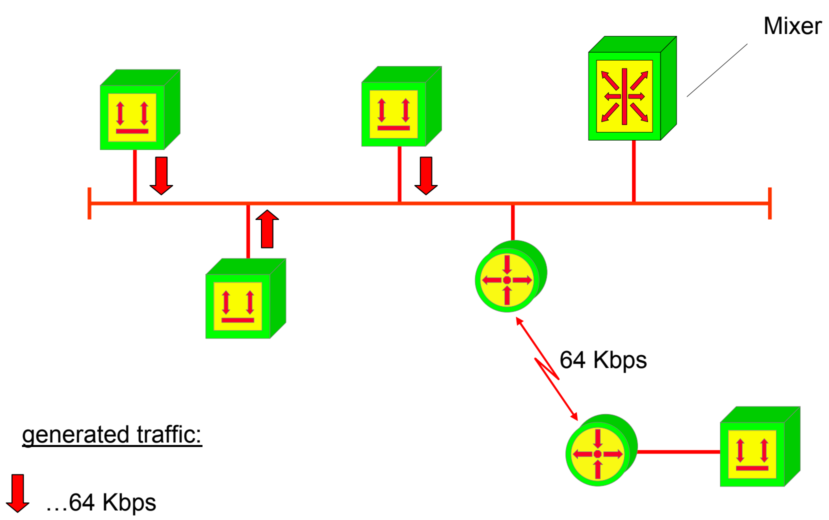
- **RTP translators (cont.)**
  - RTP translators are also used in case of firewalls which don`t pass multicast packets
  - Two translators on each side of the firewall
  - One for secure tunneling the multicast packets
  - The second forwards information as multicast packets
- **RTP mixers**
  - RTP mixers are used to combine multiple data streams into a single RTP stream
  - These devices are used to support audio transmission applications where there are only one or two simultaneous speakers

RTP Translators and Mixers

- **RTP mixers (cont.)**
  - RTP mixing is not usable in video application environments
  - In the following example, three audioconferencing workstations produce PCM audio streams at a rate of 64 Kbps
  - Another workstation connected via a lower speed serial connection wishes to participate in the audio conference
  - The bandwidth of this connection is not sufficient to support the combined 192 Kbps
  - An RTP mixer merges the three sender streams into a single 64 Kbps stream
  - This allows the new station to join the conference

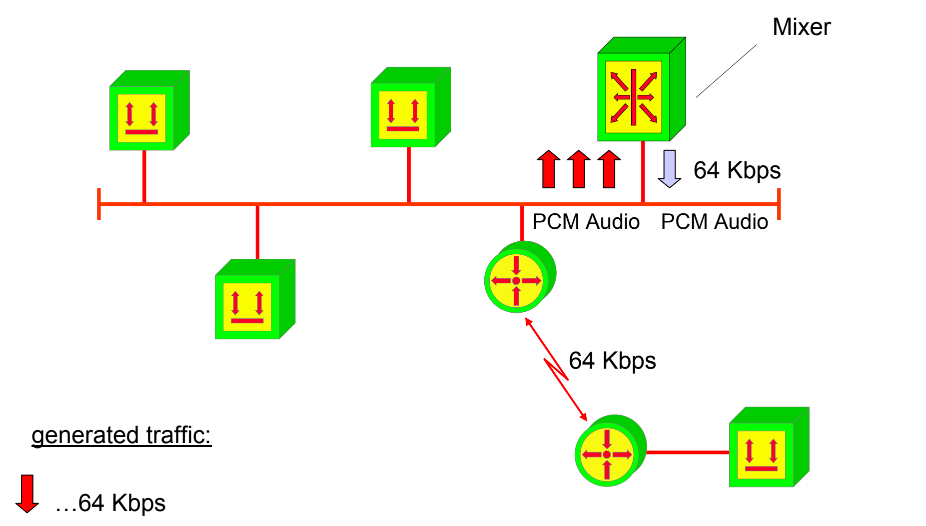
 VoIP, v1.1 65

RTP Mixing

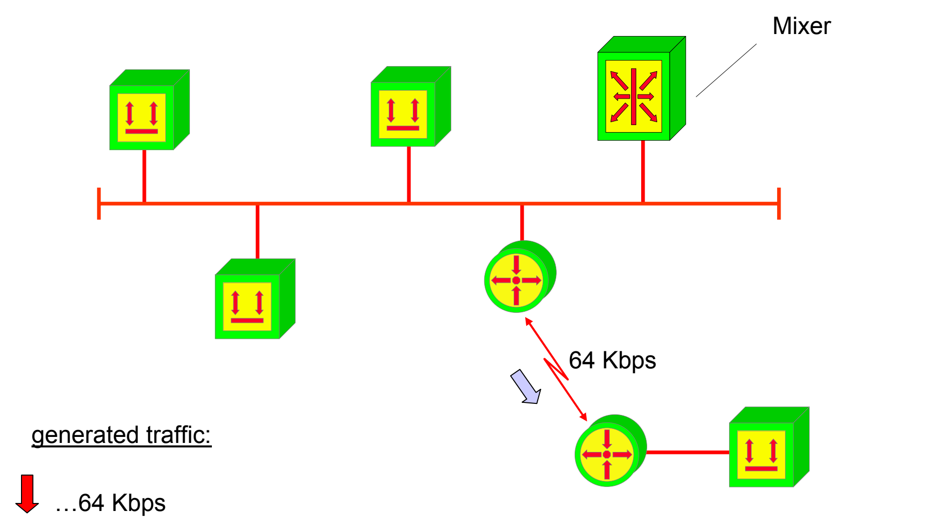


 VoIP, v1.1 66

RTP Mixing



RTP Mixing



RTP Translators and Mixers

- **RTP mixers (cont.)**
  - Payload type of the incoming and outgoing packets remain the same
  - It is possible to combine RTP mixing and RTP translating in the same environment
  - This would be required if the workstation is connected via a lower-speed link
  - Payload format of the PCM stream must be changed to a lower bandwidth specification

Agenda

- **Digitized Voice**
- **Introduction to Voice over IP**
- **RTP**
- **<u>SIP Basics 1</u>**
- **SIP Basics 2**
- **SIP in Detail**
- **H.323**

Session Initiation Protocol (RFC 3261)

- **SIP is not limited to Internet telephony**
  - SIP establishes user presence
  - SIP messages can convey arbitrary signaling payload:
    - session description, instant messages, JPEGs, any other types
- **Suitable for applications having a notion of session**
  - Distributed virtual reality systems,
  - Network games (Quake II/III implementations),
  - Video conferencing, etc.
- **Applications may leverage SIP infrastructure (Call Processing, User Location, Authentication)**
  - Instant Messaging and Presence
  - SIP for appliances

SIP Philosophy

- **Internet Standard**
  - IETF - http://www.ietf.org
- **Reuse Internet addressing**
  - URLs, DNS, proxies
  - Utilizes rich Internet feature set
- **Reuse HTTP coding**
  - Text based
- **Makes no assumptions about underlying protocol:**
  - TCP, UDP, X.25, frame, ATM, etc.
  - Support of multicast

SIP Clients and Servers

- **SIP uses client/server architecture**
- **Elements:**
  - SIP User Agents (SIP Phones)
  - SIP Servers (Proxy or Redirect - used to locate SIP users or to forward messages.)
    - Can be stateless or stateful
  - SIP Gateways:
    - To PSTN for telephony inter-working
    - To H.323 for IP Telephony inter-working
- **Client - originates message**
- **Server - responds to or forwards message**

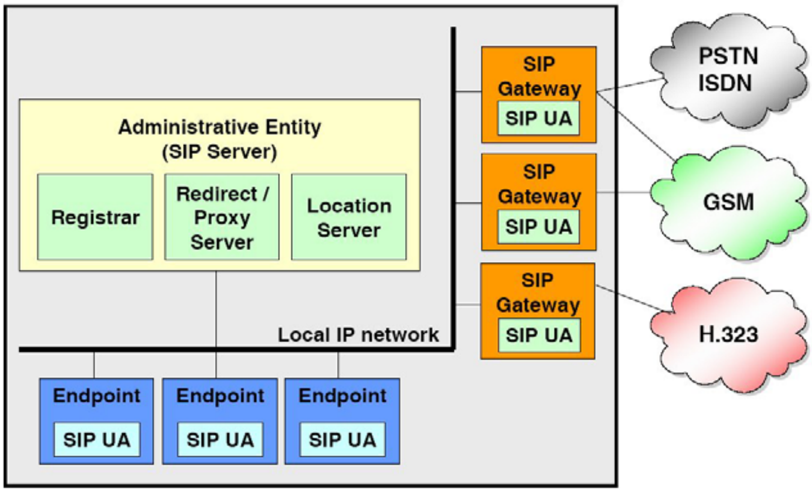
SIP Client and Servers

Local SIP entities are:

- **User Agents**
  - User Agent Client (UAC): Initiates SIP requests
  - User Agent Server (UAS): Returns SIP responses
- **Network Servers**
  - Registrar: Accepts REGISTER requests from clients
  - Proxy: Decides next hop and forwards request
  - Redirect: Sends address of next hop back to client

The different server types may be collocated

Local SIP Architecture

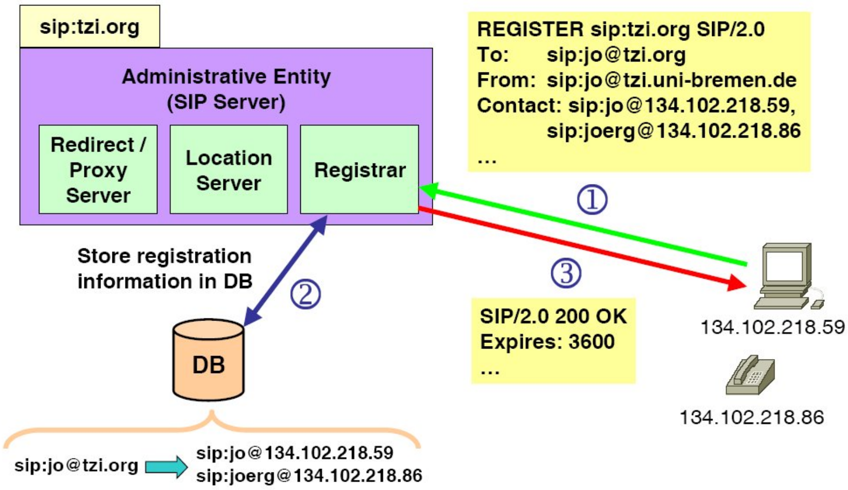


SIP Addressing

- **SIP gives you a globally reachable address**
  - Callees bind to this address using SIP REGISTER method.
  - Callers use this address to establish real-time communication with callees.
- **URLs used as address data format; examples:**
  - sip:manfred@frequentis.com
  - sip:voicemail@frequentis.com?subject=callme
  - sip:sales@hotel.xy; geo.position:=48.54_-123.84_120
- **Addresses must include host, may include user name, port number, parameters (e.g., transport), etc.**
- **Address space unlimited**

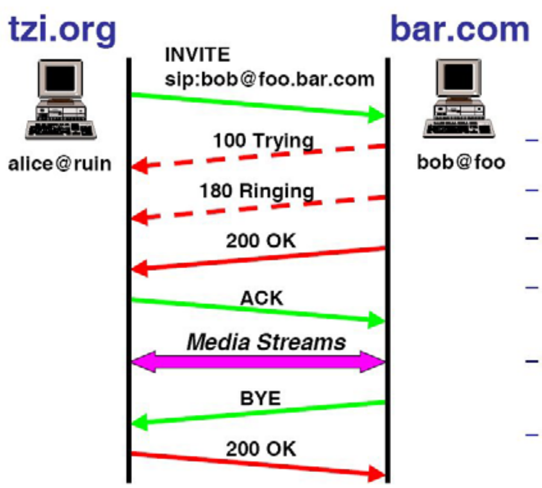
L75 - Voice over IP

SIP User Registration



 VoIP, v1.1 77

SIP Session Setup Example

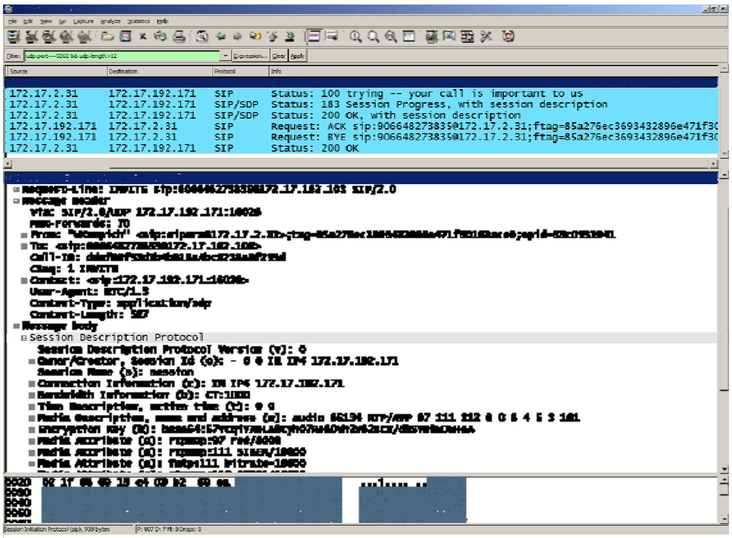


Note:
Three-way handshake is performed only for INVITE requests.

- Caller knows callee's hostname or address
- Called UA reports status changes
- After Bob accepted the call, OK is signaled
- Calling UA acknowledges, call is established
- Media data are exchanged (e. g. RTP)
- Call is terminated by one participant

 VoIP, v1.1 78

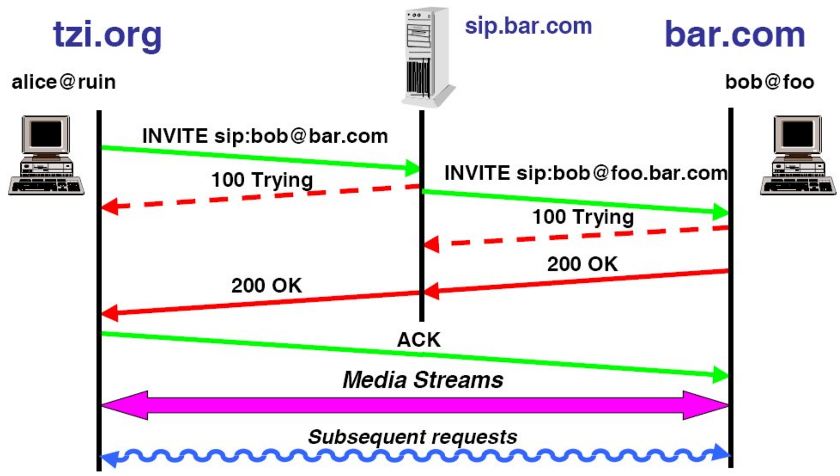
SIP Session Setup Example



Proxy Server Functionality

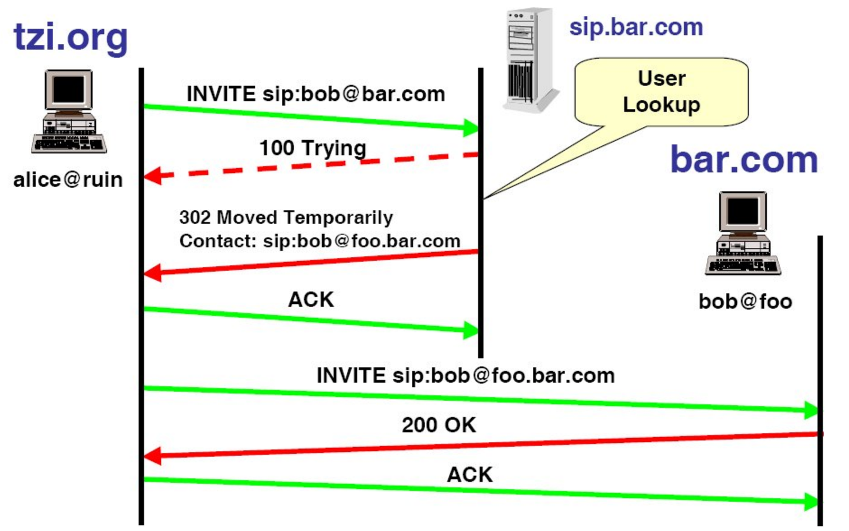
- **Serve as rendezvous point at which callees are globally reachable**
- **Perform routing function, i.e., determine to which hop (UA/proxy/redirect) signaling should be relayed**
- **Allow the routing function to be programmable - arbitrary logic may be built on top of the protocol**
  - AAA (authentication, authorization and accounting)
  - firewall control
  - etc.
- **Forking: Several destinations may be tried for a request sequentially or in parallel.**

Proxy Server Example



 VoIP, v1.1 81

Redirect Server Example



 VoIP, v1.1 82

SIP Requests

- **SIP Requests (Messages) defined as:**
  - Method SP Request-URI SP SIP-Version CRLF
  - Example: INVITE sip:manfred@frequentis.com SIP/2.0

Method	Description
INVITE	A session is being requested to be setup using a specified media
ACK	Message from client to indicate that a successful response to an INVITE has been received
OPTIONS	A Query to a server about its capabilities
BYE	A call is being released by either party
CANCEL	Cancels any pending requests. Usually sent to a Proxy Server to cancel searches
REGISTER	Used by client to register a particular address with the SIP server

SIP Requests Example

- **Required Headers (fields):**

```
INVITE sip:manfred@frequentis.com SIP/2.0
Via: SIP/2.0/UDP host.frequentis.com:5060;branch=z9hG4b
From: Wolfgang <sip:wolfgang@frequentis.com>;tag=123
To: Manfred <sip:manfred@frequentis.com>
Call-ID: 314159@host.frequentis.com
CSeq: 1 INVITE
```

Uniquely identifies this session request

  - **Via**: Shows route taken by request
  - **Call-ID**: unique identifier generated by client
  - **tag**: serves as a general mechanism to identify a dialog
  - **CSeq**: Command Sequence number
    - generated by client
    - incremented for each successive request

SIP Requests Example

- **Typical SIP Request:**

```
INVITE sip:manfred@frequentis.com SIP/2.0
Via: SIP/2.0/UDP host.frequentis.com:5060;branch=z9hG4b
From: Wolfgang <sip:wolfgang@frequentis.com>;tag=123
To: Manfred <sip:manfred@frequentis.com>
Call-ID: 314159@host.frequentis.com
CSeq: 1 INVITE
Contact: sip:wolfgang@frequentis.com
Content-Type: application/sdp
Content-Length: 124

v=0
o=wolfgang 5462346 332134 IN IP4 host.frequentis.com
t=0 0
c=IN IP4 10.64.1.1
m=audio 49170 RTP/AVP 0 3
```


SIP Responses

- **SIP Responses defined as (HTTP-style):**
  - SIP-Version SP Status-Code *SP* Reason-Phrase *CRLF*
  - Example: SIP/2.0 404 Not Found
  - First digit gives class of response:

	Description	Examples
1xx	Informational – Request received, continuing to process request.	180 Ringing 181 Call is Being Forwarded
2xx	Success – Action was successfully received, understood and accepted.	200 OK
3xx	Redirection – Further action needs to be taken in order to complete the request.	300 Multiple Choices 302 Moved Temporarily
4xx	Client Error – Request contains bad syntax or cannot be fulfilled at this server.	401 Unauthorized 408 Request Timeout
5xx	Server Error – Server failed to fulfill an apparently valid request.	503 Service Unavailable 505 Version Not Suported
6xx	Global Failure – Request is invalid at any server.	600 Busy Everywhere 603 Decline

SIP Responses Example

- **Required Headers (fields):**

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP host.frequentis.com:5060;branch=z9hG4b
From: Wolfgang <sip:wolfgang@frequentis.com>;tag=123
To: Manfred <sip:manfred@frequentis.com >;tag=987
Call-ID: 314159@host.frequentis.com
CSeq: 1 INVITE
```

  - `Via`, `From`, `To`, `Call-ID`, and `CSeq` are copied exactly from request
  - `To` and `From` are **NOT** swapped!
  - `tag`: serves as a general mechanism to identify a dialog

SIP Responses Example

- **Typical SIP Response (containing SDP):**

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP host.frequentis.com:5060;branch=z9hG4b
From: Wolfgang <sip:wolfgang@frequentis.com>;tag=123
To: Manfred <sip:manfred@frequentis.com>;tag=987
Call-ID: 314159@host.frequentis.com
CSeq: 1 INVITE
Contact: sip:wolfgang@frequentis.com
Content-Type: application/sdp
Content-Length: 107

v=0
o=wolfgang 124333 67895 IN IP4 frequentis.com
t=0 0
c=IN IP4 11.234.2.1
m=audio 3456 RTP/AVP 0
```

SIP Message Body

- **Message body can be any protocol**
- **Most implementations:**
  - SDP - Session Description Protocol
  - RFC 2327 4/98 by Handley and Jacobson
    - http://www.ietf.org/rfc/rfc2327.txt
  - Used to specify info about a multi-media session.
  - SDP fields have a required order
  - For RTP - Real Time Protocol Sessions:
    - RTP Audio/Video Profile (RTP/AVP) payload descriptions are often used

Session Description Protocol (RFC 2327)

- **Convey sufficient information to enable participation in a multimedia session**
- **SDP includes description of:**
  - Media to use (codec, sampling rate)
  - Media destination (IP address and port number)
  - Session name and purpose
  - Times the session is active
  - Contact information
- **Note: indeed SDP is a data format rather than a protocol**

SDP Examples

SDP Example 1

v=0
o=manfred 5462346 332134 IN IP4 host.frequentis.com
s=Let's Talk
t=0 0
c=IN IP4 10.64.1.1
m=audio 49170 RTP/AVP 0 3

Field	Descripton
Version	v=0
Origin	o=<username> <session id> <version> <network type> <address type> <address>
Session Name	s=<session name>
Times	t=<start time> <stop time>
Connection Data	c=<network type> <address type> <connection address>
Media	m=<media> <port> <transport> <media format list>

SDP Example 2

v=0
o=wolfgang 124333 67895 IN IP4 pc.frequentis.com
t=0 0
c=IN IP4 11.234.2.1
m=audio 3456 RTP/AVP 0

PSTN Features with SIP (Examples)

- **Features implemented by SIP Phone**
  - Call answering: 200 OK sent
  - Busy: 483 Busy Here sent
  - Call rejection: 603 Declined sent
  - Caller-ID: present in From header
  - Hold: a re-INVITE is issued with IP Addr =0.0.0.0
  - Selective Call Acceptance: using From, Priority, and Subject headers
  - Camp On: 181 Call Queued responses are monitored until 200 OK is sent by the called party
  - Call Waiting: Receiving alerts during a call

PSTN Features with SIP (Examples)

- **Features implemented by SIP Server**
  - Call Forwarding: server issues 301 Moved Permanently or 302 Moved Temporarily response with Contact info
  - Forward Don't Answer: server issues 408 Request Timeout response
  - Voicemail: server 302 Moved Temporarily response with Contact of Voicemail Server
  - Follow Me Service: Use forking proxy to try multiple locations at the same time
  - Caller-ID blocking - Privacy: Server encrypts From information

Authentication & Encryption

- **SIP supports a variety of approaches:**
  - end to end encryption
  - hop by hop encryption
- **Proxies can require authentication:**
  - Responds to INVITEs with 407 Proxy-Authentication Required
  - Client INVITEs with Proxy-Authorization header.
- **SIP Users can require authentication:**
  - Responds to INVITEs with 401 Unathorized
  - Client INVITEs with Authorization header

SIP Summary

- **SIP is:**
  - mainly establishes the IP addresses and port numbers at which the end systems can send and receive data
  - Relatively easy to implement and very flexible in service creation
  - extensible and scaleable
- **SIP is not:**
  - going to solve all IP Telephony issues (QoS)
  - designed for distribution of media data
  - a generic transport protocol
- **SIP does not dictate ...**
  - product features and services (color of a phone and distinctive ringing melodies, number simultaneous calls a phone can handle …
  - network configuration

SIP vs. H.323

- **H.323 (ITU-T)**
  - Deployment started earlier
  - Shorter messages (ASN.1 encoded)
  - Special parsers needed to map into readable form and vice versa
  - Implementation and debugging complicated
- **SIP**
  - Scalability, extensibility, less complexity
  - Ease of Implementation and customization
  - Call forking, third-party call control …
  - SIP is best described as toolbox offering a number of standardized tools to create any applications you like

SIP User Agents - Software

- X-Lite: eyeBeam

SIP User Agents - Hardware

- Low-cost SIP Phones

SIP User Agents - Hardware

- High-end SIP Phones

SIP User Agents - Hardware

- Analog Adapter

SIP is a defined standard

- IETF
- RFC 2543 (main document)
- RFC 2782 (DNS – SRV resource record type)

Agenda

- Digitized Voice
- Introduction to Voice over IP
- RTP
- SIP Basics 1
- <u>SIP Basics 2</u>
- SIP in Detail
- H.323

Protocol Design

- **Simple - text based format (HTTP)**
  - therefore programmable (CGI, JavaApplets,..)
- **Infrastructure follows IP model**
  - intelligence and state in end-devices
  - low cpu consumption in servers
  - high scalability (no single point of failure)
- **uses UDP**
  - faster set-up
  - less states

What is SIP?

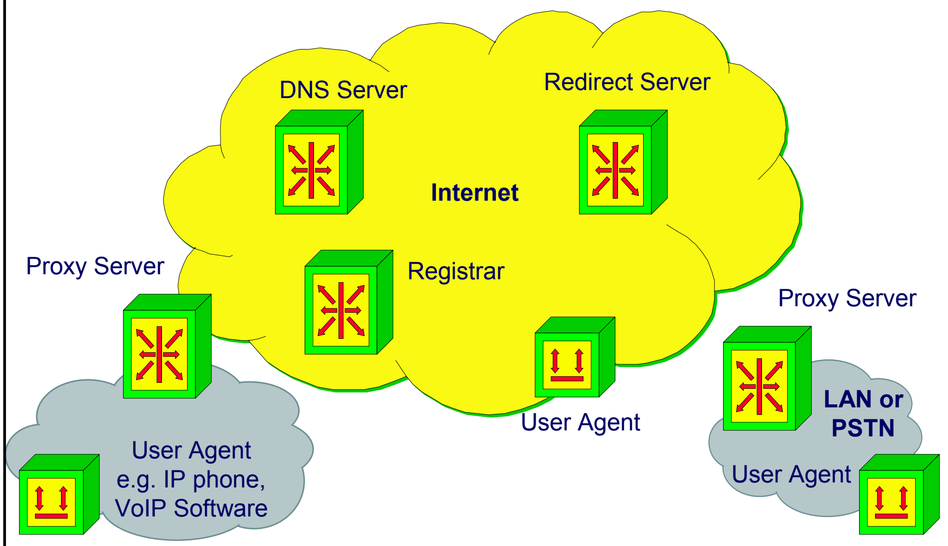
- **SIP (Session Initiation Protocol)**
- **establishes connection between 2 or more IP nodes for media (e.g. VoIP)**
- **client – server session signaling protocol**
- **provides presence information**
- **offers possibility for mobility**

What SIP is not

- **Transport Protocol**
  - media path is not the same as the call setup path
  - RTP plays that role in VoIP
  - SIP is not responsible for the data format or a compression type
- **offers no QoS features**
  - can partly be implemented with SDP

 VoIP, v1.1 105

SIP Environment



 VoIP, v1.1 106

Elements of an SIP System

- **User Agent (user application)**
  - User Agent client (originates calls)
  - User Agent server (listens for incoming calls)
  - can be Hardware or Software
- **SIP Proxy Server**
  - relays call signalling
- **SIP Redirect Server**
  - redirects callers to other servers
- **SIP Registrar Server**
  - registers users

often implemented in a single application running on a server

SIP Addresses

- **globally unique and globally reachable**
- **Users bind to address with Register Message at Location Server**
- **format: sip:user@host**
  - may include port, parameters, password
- **Examples:**
  - sip:mike@aol.com
  - sip:harry@aon.at?subject=answer
  - sip:luke@msn.com:5060;transport=tcp
  - sip:0245256842@telekom.de;user=phone

SIP Call Signalling Methods

- **Format: <method><address><sip-version>**
- **INVITE**
  - initiates sessions
  - session description included in message body
- **REINVITE**
  - used for session mobility
- **ACK**
  - confirms session establichment
  - only used with INVITE
- **BYE**
  - terminates session

 VoIP, v1.1 109

SIP Call Signalling Methods (cont.)

- **CANCEL**
  - cancels a pending INVITE
- **OPTIONS**
  - queries a User Agent for its capabilities
- **REGISTER**
  - binds an address to current location
  - sent from User Agent to Registrar Server
- **PRACK**
  - User Agent requests delivery of informational responses
- **COMET (extended method)**
  - used for SDP answers

 VoIP, v1.1 110

SIP Response Codes

- Borrowed from HTTP
  - 3 digit number xyz + explanatory text,
  - Receiver needs to understand x
- 1yz Informational
  - 100 Trying
  - 180 Ringing
  - 181 Call is Being forwarded
- 2yz Success
  - 200 Ok
- 3yz Redirection
  - 300 multiple Choices
  - 302 moved Temporarily

SIP Response Codes (cont.)

- 4yz Client error
  - 400 Bad Request
  - 401 Unauthorized
  - 482 Loop detected
  - 486 Busy
- 5yz Server failure
  - 500 Internal server error
- 6yz Global failure
  - 600 Busy everywhere
  - 603 Decline

Message structure

INVITE: sip:luke@aon.com SIP/2.0	Message Method
Via: SIP/2.0/UDP msn.com:5060 **From:** Max <sip:max@msn.com> **To:** Luke <sip:luke@aol.com> **Call-ID:** **Cseq:** 1 INVITE **Subject:** Call me back **Contact:** Max <sip:max@msn.com> **Content-Type:** application/sdp **Content-Length:** 147	Message Header
v=0 o=Max 41542546 4152546 IN IP4 msn.com s=Session SDP c=IN IP4 100.101.102.103 t=0 0 m=audio 49172 RTP/AVP 0 a=rtpmap:0 PCMU/8000	SDP

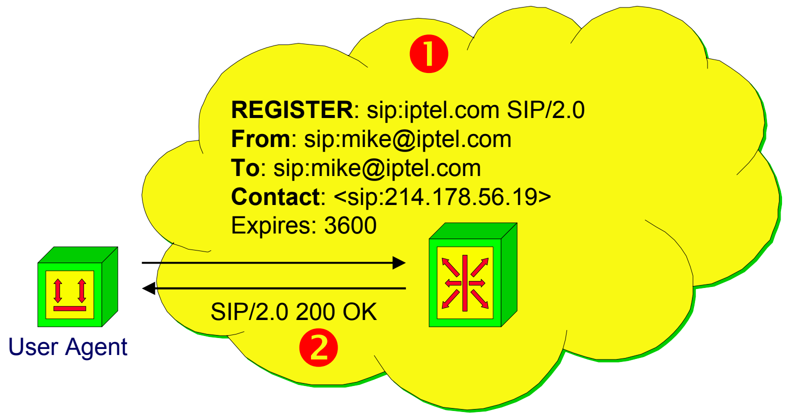
SDP – Session Description Protocol

- **body of SIP Messages (INVITE, ACK, OPTIONS)**
- **identifies all attributes of a session**
- **Attributes:**
  - v protocol version
  - o owner and session identifier
  - s session name
  - c connection information
  - t time the session is active
  - m media name and transport address
  - a media attributes

SIP Programming

- SIP follows HTTP programming model
- suggested by IETF: CGI, Call Processing Language (CPL), Servlets
- Users and third parties may code
- usable to establish call policies like:
  - „redirect authenticated friends to my cell phone, anyone else to my recorder“
  - „if busy, retrun my homepage and redirect to recorder“
- Information sent to Redirect/Proxy Server

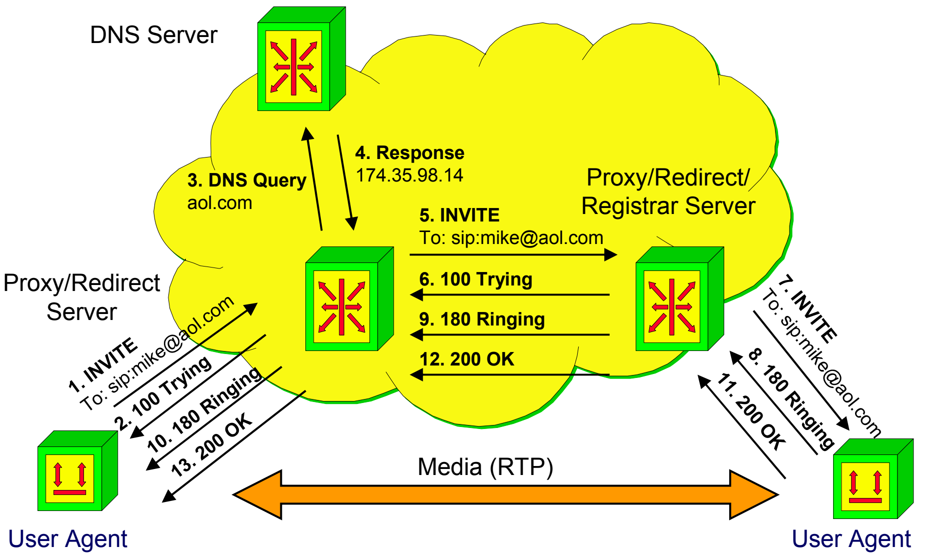
SIP Registration



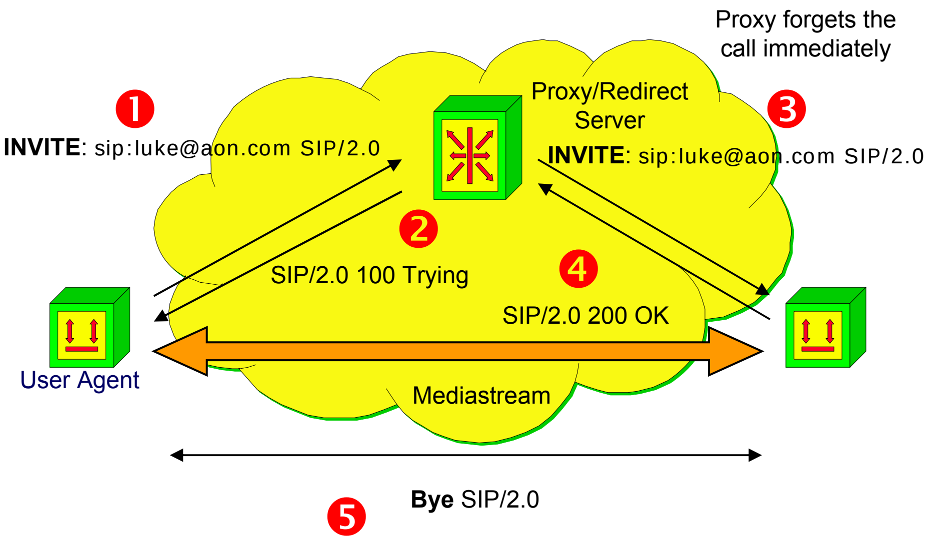
Why using a proxy

- Proxy not essential
- easier to manage
- single point of reference
- may be needed to get through firewall

Call set up via Proxy



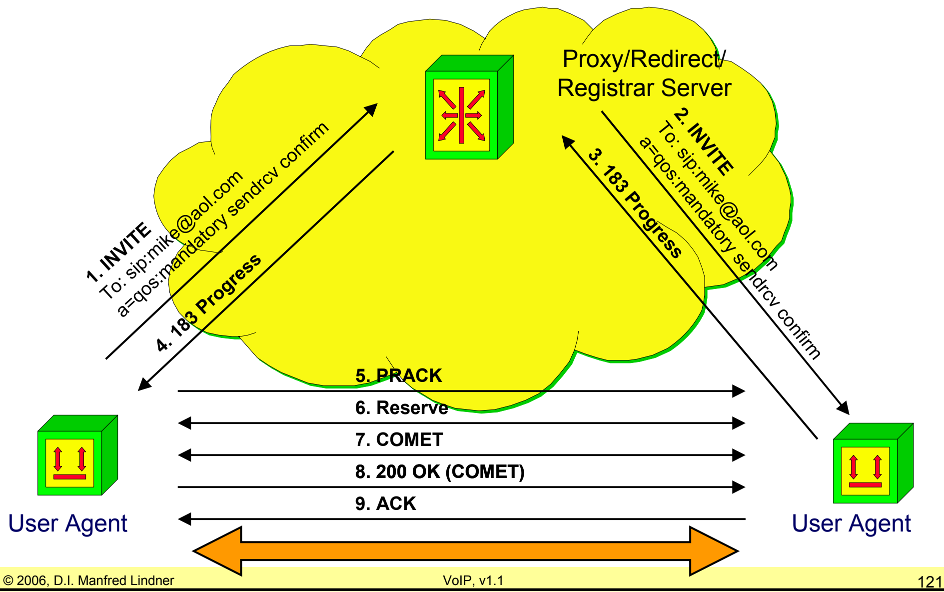
SIP is stateless



SIP and QoS

- SIP does not provide any QoS support
- Preconditions can be specified by SDP
- Objective is to ensure that these preconditions are met before the phone rings
- COMET method indicates if preconditions are met or not

Quality of Service with SDP



Finding a registrar

- Static configured
- Multicast (224.0.1.75 – sip.meast.net)
- DNS (SRV resource record type)
- DHCP (configuration file)

Mobility

- Mobile hosts inform their home Proxy about their new location using REGISTER
  - binds person to a device
- Proxy redirects call to foreign Proxy or IP address
- Mid-call mobility (session mobility) is achieved with REINVITE (mobile phones)
- Services like address book, call policy stored at home Proxy - Service mobility

Agenda

- Digitized Voice
- Introduction to Voice over IP
- RTP
- SIP Basics 1
- SIP Basics 2
- SIP in Detail
- H.323

Terminology

- **User Agent Client (UAC)**
  - endpoint, initiates SIP transactions
- **User Agent Server (UAS)**
  - handles incoming SIP requests
- **Redirect server**
  - retrieves addresses for callee and returns them to caller
- **Proxy (server)**
  - UAS/UAC that autonomously processes requests
  - forwards incoming messages (probably modified)
- **Registrar**
  - stores explicitly registered user addresses
- **Location server**
  - provides information about a target user's location

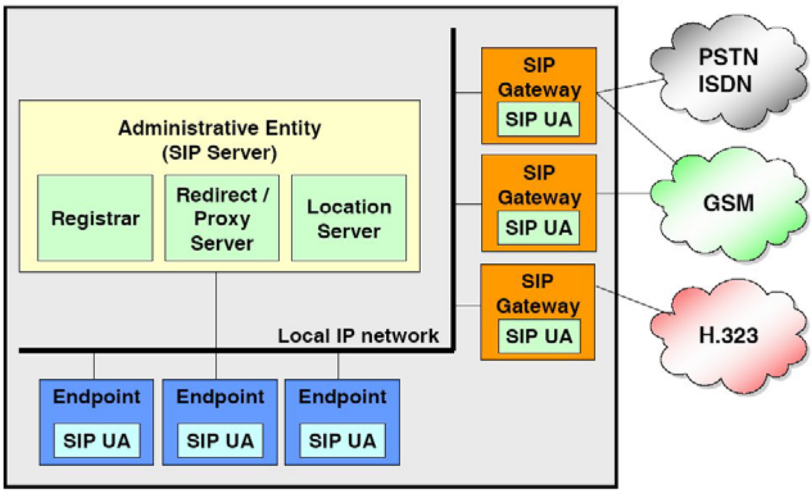
Main SIP-Messages

- **REGISTER**
  - registration request sent to registrar
- **INVITE**
  - session invitation
- **ACK**
  - acknowledge message
- **OK**
  - the request has succeeded
- **CANCEL**
  - used to cancel a previous request
- **BYE**
  - session close-down
- **OPTIONS**
  - used for determining the capabilities of a UA

Responses

- **1xx: Provisional**
  - request received, continuing to process the request
- **2xx: Success**
  - the action was successfully received, understood, and accepted
- **3xx: Redirection**
  - further action needs to be taken in order to complete the request
- **4xx: Client Error**
  - the request contains bad syntax or cannot be fulfilled at this server
- **5xx: Server Error**
  - the server failed to fulfill an apparently valid request
- **6xx: Global Failure**
  - the request cannot be fulfilled at any server

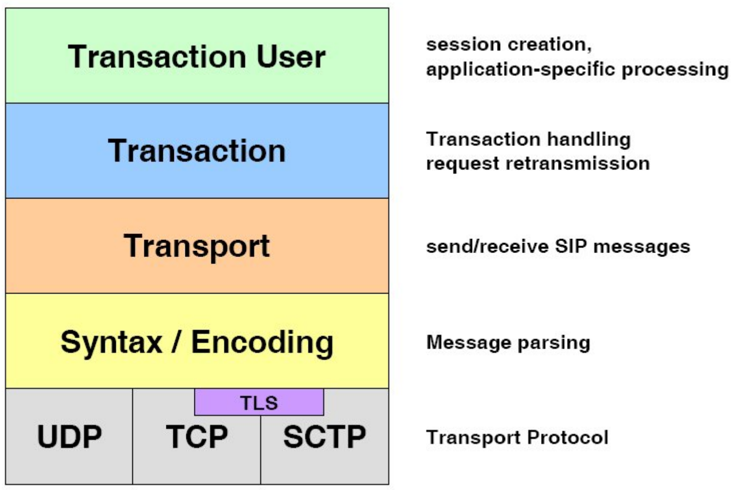
Local SIP Architecture



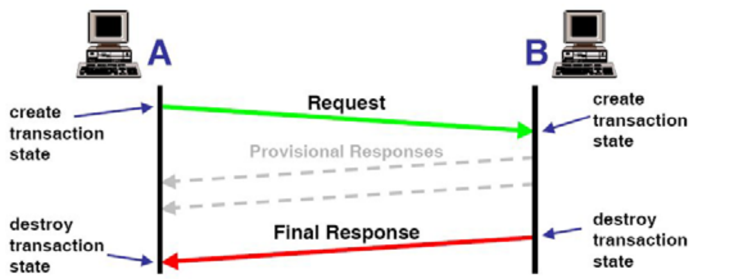
Protocol Characteristics

- **Transaction oriented**
  - request-response sequences
- **Independent from lower layer transport protocol**
  - works with a number of unreliable and reliable transports
    - UDP, TCP, SCTP
    - secure transport: TLS over TCP, IPSec
  - retransmissions to achieve reliability over UDP
  - optionally use IP multicast - anycast service
- **Independent of the session to be (re-) configured**
- **Re-use syntax of HTTP 1.1**
  - text -based protocol (UTF-8 encoding)
- **Enable servers maintaining minimal state info**
  - stateless proxies, transaction-stateful proxies
  - dialog (call) state in endpoints (optional for proxies)

Functional Layers



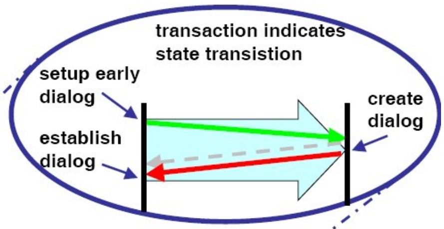
SIP Transactions

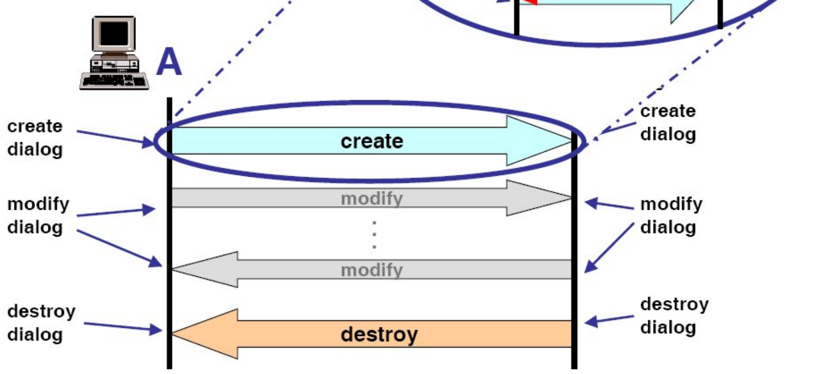


- RPC-like approach:
  - Initial request
  - Wait for final response
- Provisional responses:
  - Additional status information
  - May be unreliable
- Unique identifier (transaction id) (originator, recipient, unique token, sequence number, ...)
- Independent completion

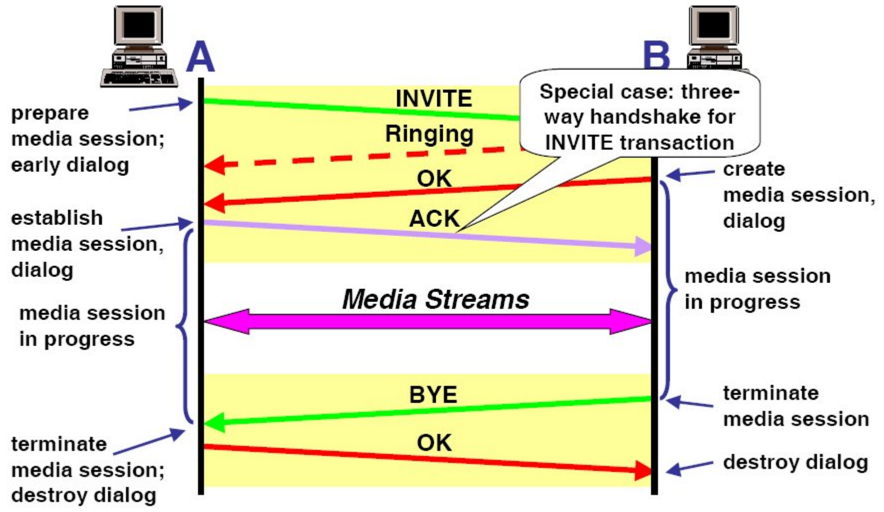
SIP Dialogs

- Signaling vs. media session
- Distributed state between endpoints
  - State change if transaction suceeds
  - No change on error
- Unique dialog identifier





Dialog Example: Media Session



SIP Message Syntax: Request

Start line

```
INVITE sip:user@example.com SIP/2.0
```

Message headers

```
To: John Doe <sip:user@example.com>
From: sip:jo@tzi.uni-bremen.de;tag=4711
Subject: Congratulations!
Content-Length: 117
Content-Type: applicaton/sdp
Call-ID: 2342344233@134.102.218.1
CSeq: 49581 INVITE
Contact: sip:jo@134.102.224.152:5083
          ;transport=udp
Via: SIP/2.0/UDP 134.102.218.1
```

Message body (SDP content)

```
v=0
o=jo 75638353 98543585 IN IP4 134.102.218.1
s=SIP call
t=0 0
c=IN IP4 134.102.224.152
m=audio 47654 RTP/AVP 0 1 4
```

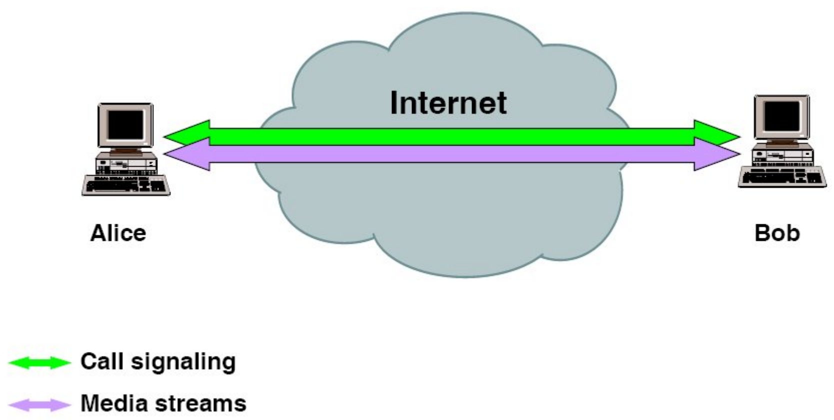
SIP Addressing Scheme

- **SIP URI: generic syntax specified in RFC 2396**
- **Two roles:**
  - naming a user; typically sip:user@domain
  - contact address of user or group; typically contains host name or IP address, port, transport protocol, …
- **May contain header fields for SIP messages**
- **Support for telephone subscribers instead of user**
  - use phone number as specified in RFC 2806

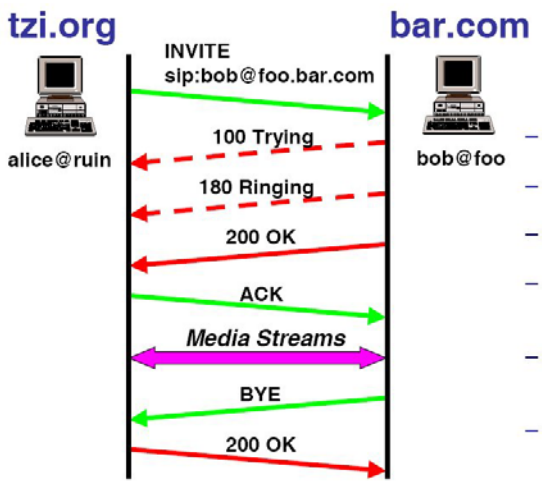
´sip:´[user[´:´[passwd]´@´] host [´:´port] params [´?´ headers]

params::= (´;´ name[´=´ value])*
headers::= field ´=´ value?[´&´headers]

Application Scenario: Direct Call UA-UA



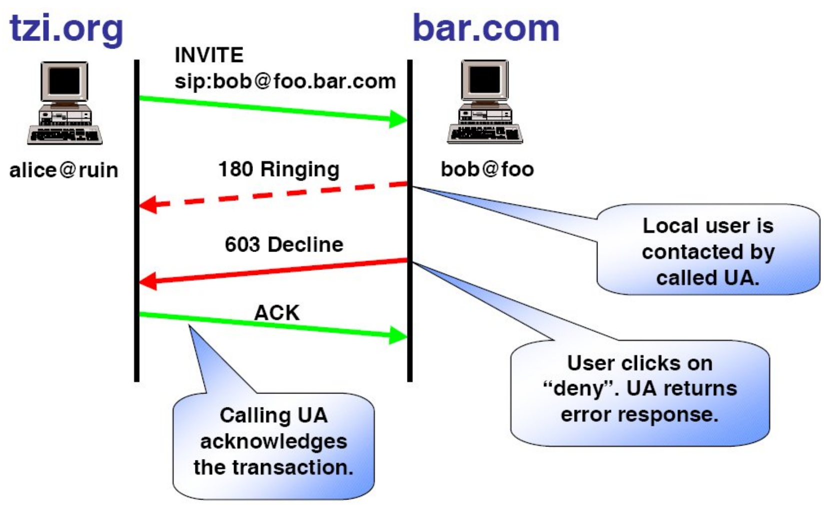
Direct Call



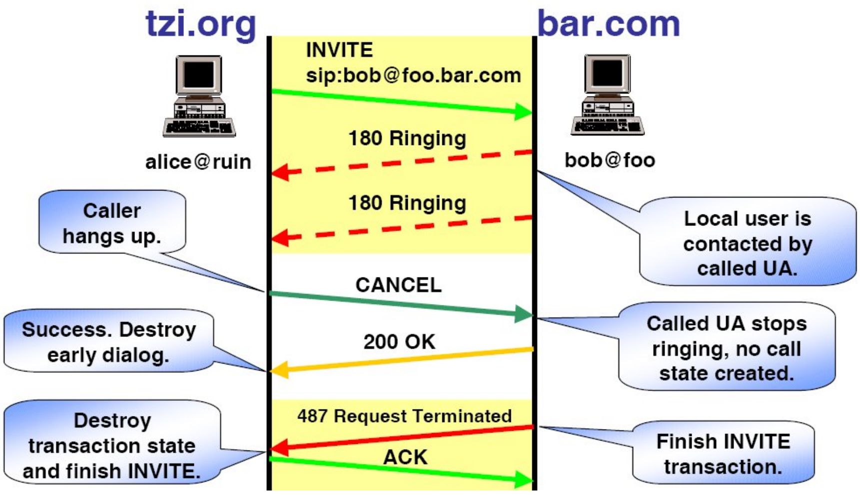
Note:
Three-way handshake is performed only for INVITE requests.

- Caller knows callee's hostname or address
- Called UA reports status changes
- After Bob accepted the call, OK is signaled
- Calling UA acknowledges, call is established
- Media data are exchanged (e. g. RTP)
- Call is terminated by one participant

Callee Declines Call

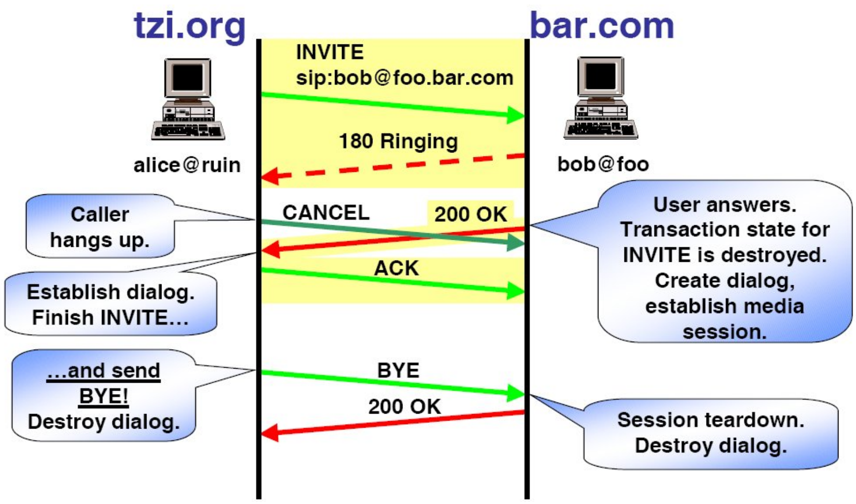


Caller Gives Up



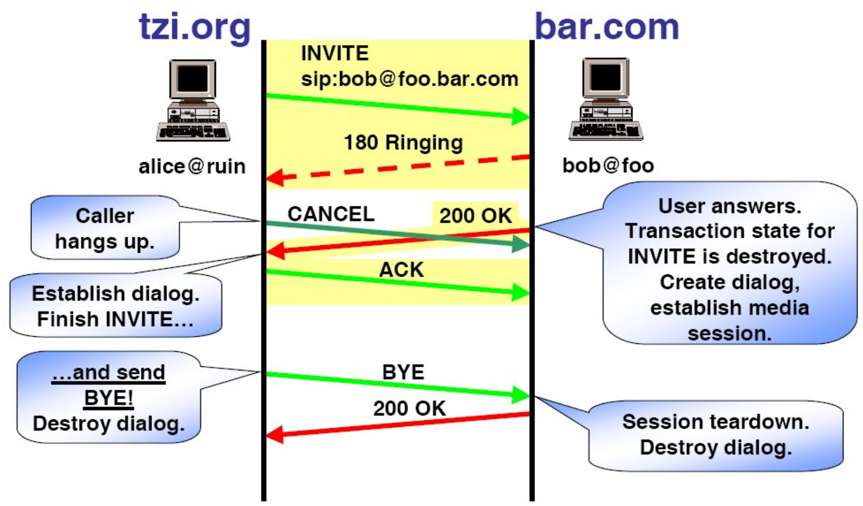
© 2006, D.I. Manfred Lindner VoIP, v1.1 139

Caller Gives Up While Call Established



© 2006, D.I. Manfred Lindner VoIP, v1.1 140

Caller Gives Up While Call Established



 VoIP, v1.1 141

How to Find the Callee?

- **Direct calls require knowledge of callee's address**
- **SIP provides abstract naming scheme:**

  sip:user@domain

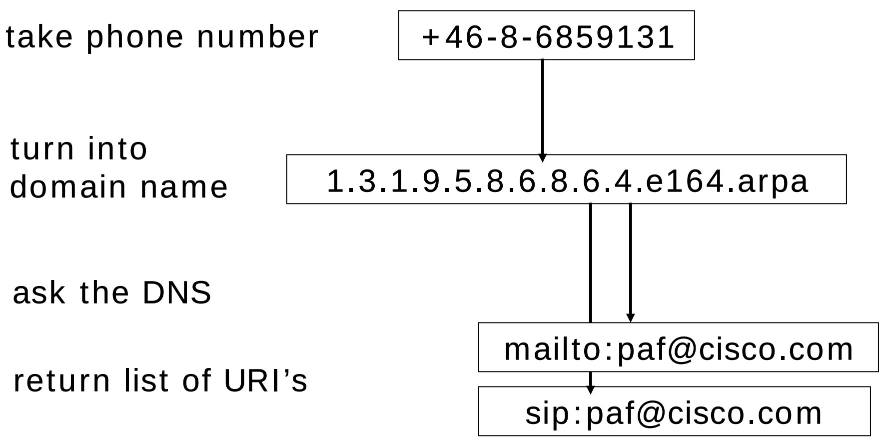
- **Define mapping from SIP URI to real locations**
  - explicit registration
    - UA registers user's name and current location
  - location service
    - use other protocols to find potentially correct addresses
- **Caller sends INVITE to any SIP server knowing about the callee's location**
- **Receiving server may either redirect, refuse or proxy**

 VoIP, v1.1 142

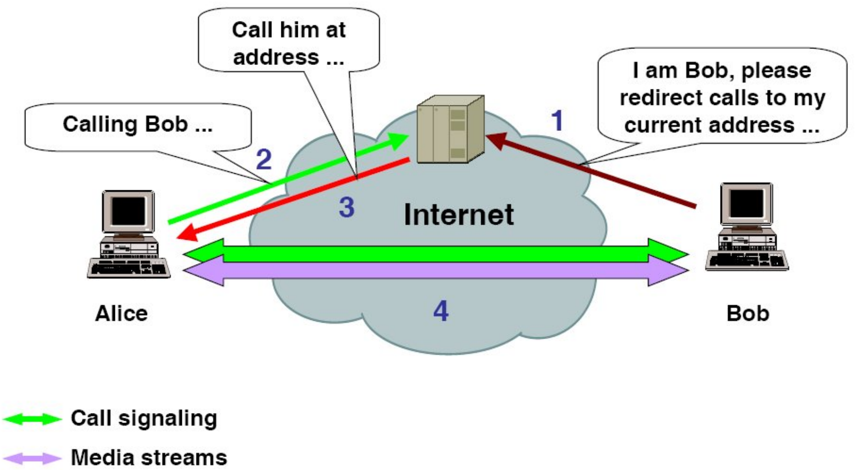
Finding the Next Hop

- **UAC may use a (manually) configured outbound proxy**
  - outbound proxy may also have be learned upon registration
- **If request URI contains IP address and port, message can be sent directly**
- **Otherwise, determine far-end SIP server via DNS**
  - if entries found, try as specified in RFC 2782
- **Last resort query A records**
  - for specified domain name
  - e.g. for sip.domain

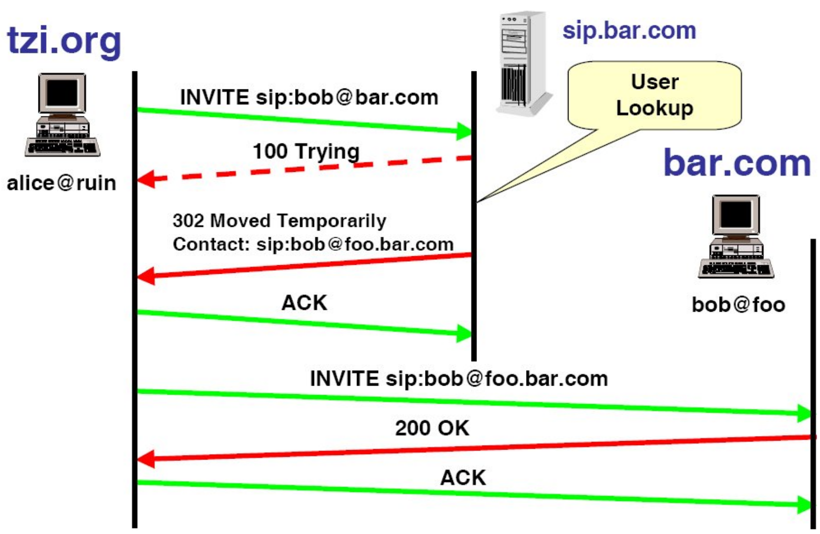
ENUM in a Nutshell



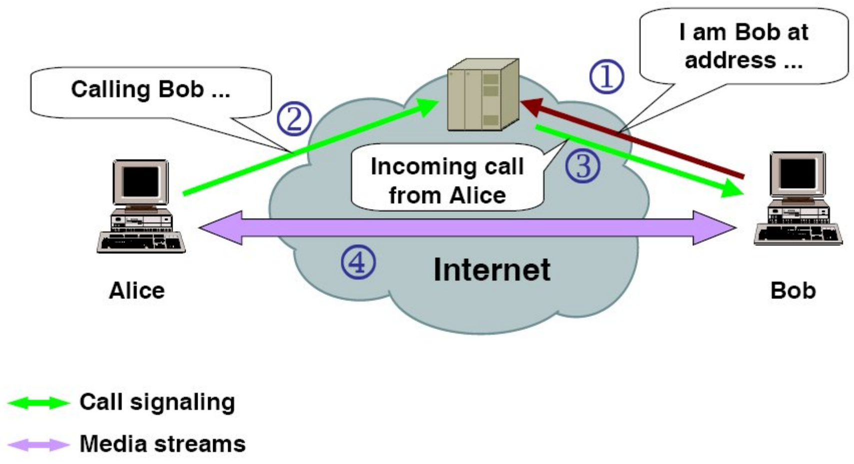
Application Scenario: Redirect Call



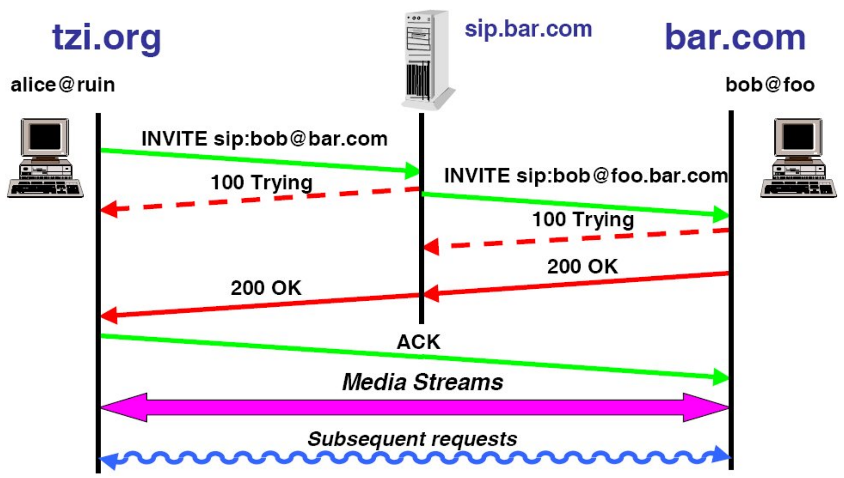
Redirect Call



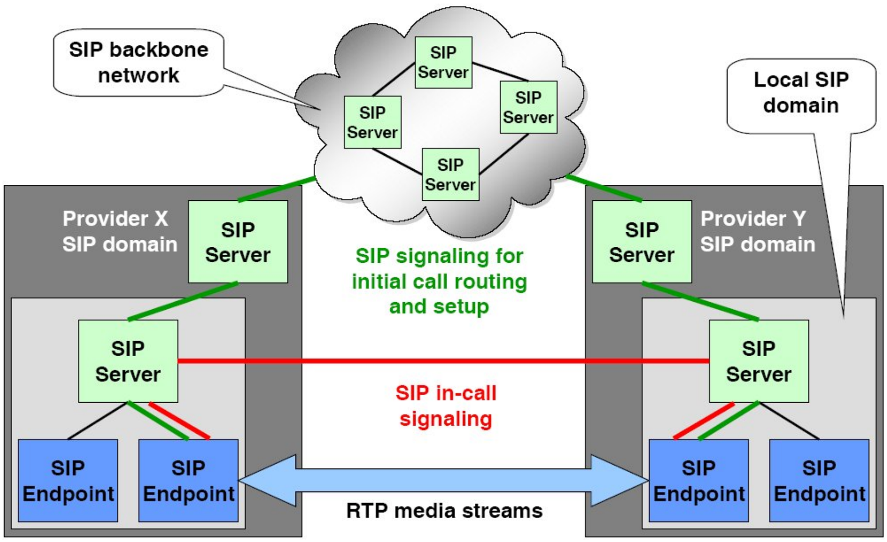
Application Scenario: Proxied Call



Proxied Call



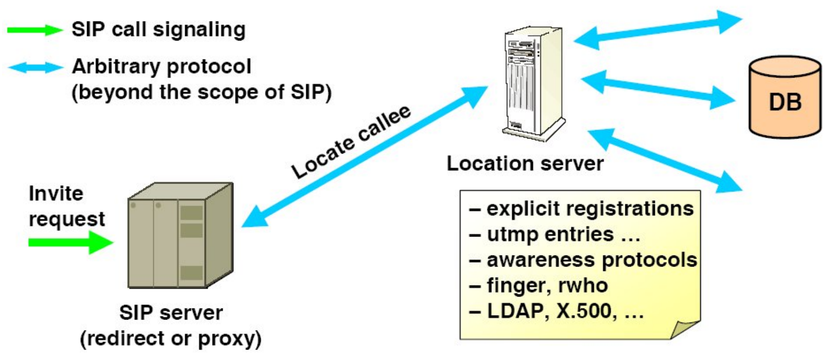
Global SIP Architecture



SIP (Proxy) Server Functionality

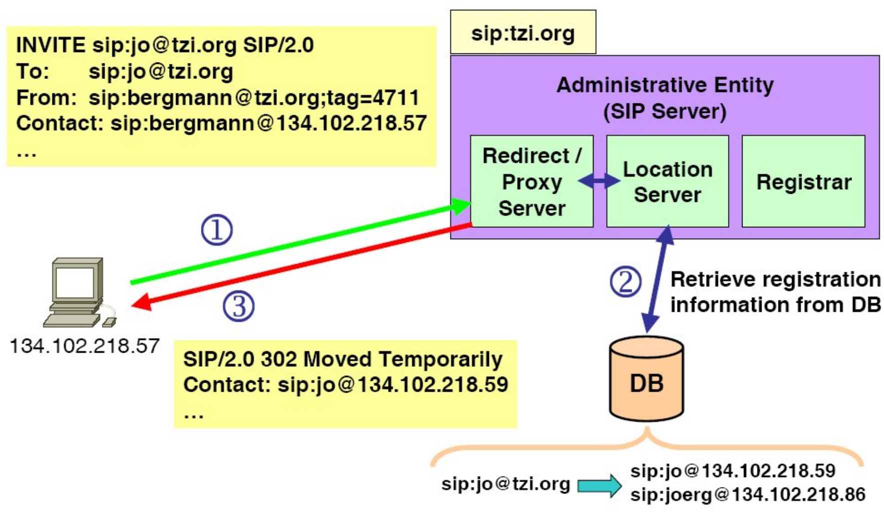
- **Stateless vs. stateful**
  - Stateless: efficient and scalable call routing (backbone)
  - Stateful: service provisioning, firewall control, …
- **Some roles for proxies**
  - outbound proxy
    - perform address resolution and call for endpoints
    - pre-configured for endpoint (manually, DHCP, …)
  - backbone proxy
    - essentially call routing functionality
  - access proxy
    - user authentication and authorization, accounting
    - hide network internals (topology, devices, users, etc.)
  - local IP telephony server (IP PBX)
  - service creation in general

User Location

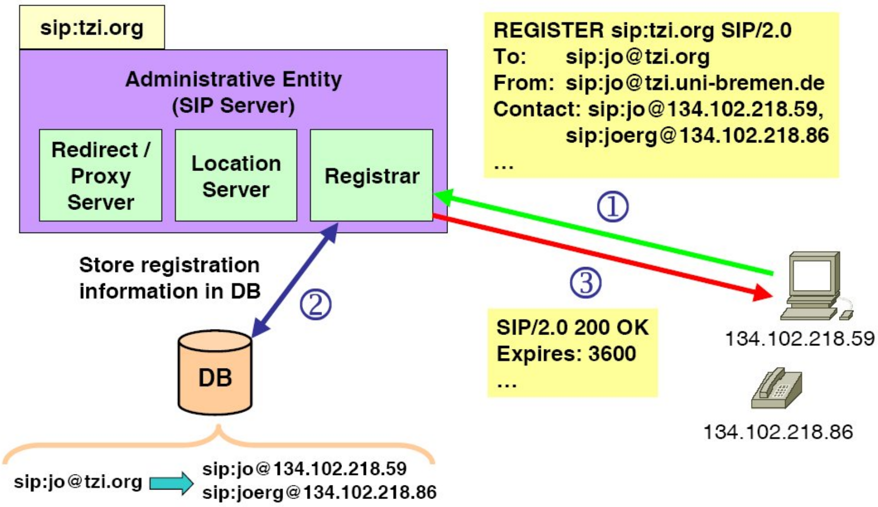


- SIP server asks location server where to find callee
- Location server returns list of contact addresses
- SIP server proxies or redirects request according to address list

User Location



User Registration



 VoIP, v1.1 153

User Registration

- **Send REGISTER request to registrar**
- **Request URI *sip:domain***
  - registrar may refuse may refuse requests for foreign domains
- To: **canonic name for registered user**
  - usually sip:user@domain
- From: **responsible person**
  - may vary from **To:** for third party registration
- Contact: **contact information for the registered user**
  - address, transport parameters, redirect/proxy
- **Specified addresses are merged with existing registrations**
- **Registrar denotes expiration time in** Expires: **header**
- **Client refreshes registration before expiry**

 VoIP, v1.1 154

Registration Expiry

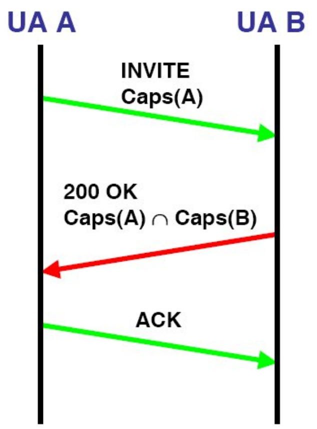
- **Client requests lifetime**
  - Contact: -header parameter *expires*
  - SIP message header field **Expires:**
  - relative duration (seconds) or absolute date
  - default if no expiry time requested: 3600 seconds
- **Registrar may use lower or higher value, indicated in OK response**
  - registrar must not increase expiry interval, may decline request with "423 Registration Too Brief" and **Min-Expiry**: header
- **After expiration, registrar silently discards corresponding database entries**

Capability Negotiation

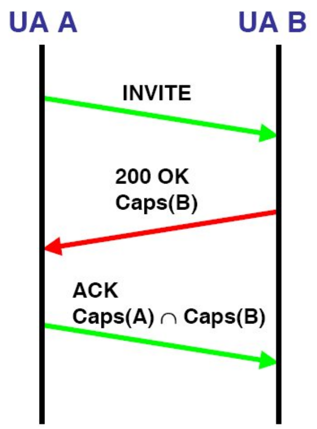
- **SDP: Session Description Protocol, RFC 2327**
- **Caller includes SDP capability description in INVITE**
  - time information may be set to "t=0 0" or omitted
  - for RTP/AVT, use of *rtpmap* mappings is encouraged
- **For each media stream (*m*-part of SDP message), callee returns own configuration in response**
  - indicate destination address in *c*-field
  - indicate port and selected media parameters in *m*=-field
  - set port to zero to suppress media streams
- **UA may return user's capability in 200 OK response when receiving an OPTIONS request**

Media Negotiation During Call Setup

“Normal” operation



Delayed description



Example SDP Alignment

v=0 o=jo 7849 2873246 IN IP4 ruin.inf… s=SIP call t=0 0 c=IN IP4 134.102.218.1 m=audio 52392 RTP/AVP 98 99 a=rtpmap:98 L8/8000 a=rtpmap:99 L16/8000 m=video 59485 RTP/AVP 31 a=rtpmap:31 H261/90000	v=0 o=cabo 82347 283498 IN IP4 dmn.inf… s=SIP call t=0 0 c=IN IP4 134.102.218.46 m=audio 49823 RTP/AVP 98 a=rtpmap:98 L8/8000 m=video 0 RTP/AVP 31

Resulting configuration:

Send/Receive Only

- **Media streams may be unidirectional**
  - indicated by *a=sendonly, a=recvonly*
- **Attributes are interpreted from sender's view**
- **Sendonly**
  - recipient of SDP description should not send data
  - connection address indicates where to send RTCP receiver reports
  - multicast session: recipient sends to specified address
- **Recvonly**
  - sender lists supported codecs
  - receiver chooses the subset he intends to use
  - multicast session: recipient listens on specified address
- **Inactive**
  - to pause a media stream (rather than deleting it)

SIP and Security

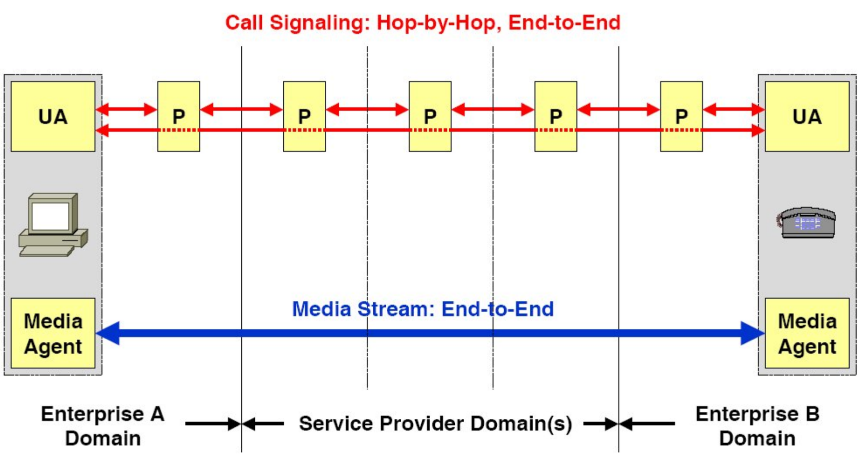
- **SIP entities are potential target of a number of attacks, e.g.**
  - spoofing identity
  - eavesdropping
    - media stream
    - call signaling
  - traffic analysis
  - theft of service
  - denial of service (DoS)
- **Some countermeasures**
  - client and server authentication
  - request authorization
  - encryption
  - message integrity checks + reply protection

Why SIP Security?

- **Ensure privacy**
  - media encryption
  - anonymous calls
  - personalized services
- **Billing and accounting**
  - probably pay for assured bandwidth, etc.
- **Regulatory requirements**
  - call id blocking
  - call tracing facility
  - emergency call service
  - multi-level prioritization and preemption

 VoIP, v1.1 161

SIP Security Overview

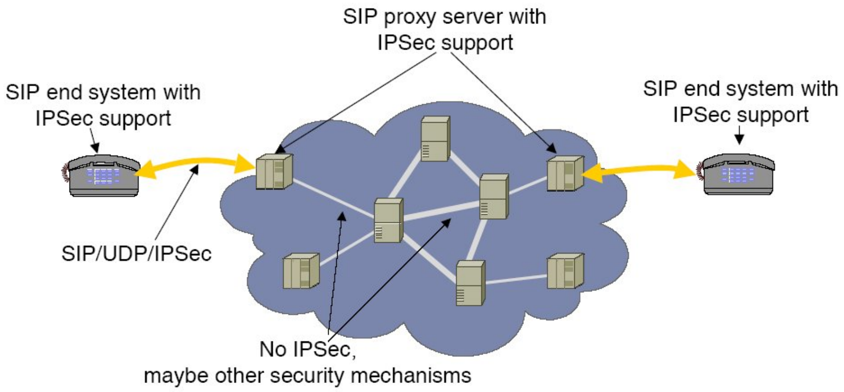


 VoIP, v1.1 162

Hop-by-hop Encryption of SIP Messages

- Lower layer mechanisms
  - applicability depends on link layer technology
- VPN-like tunnel using IPSec
  - suitable e.g. for coupling site of a company
  - need OS-support (required for IPv6 anyway)
- SIP over TLS (Transport Layer Security)
  - access to outbound proxy
  - call routing to ITSP
  - call routing between neighboring ITSPs (agreements!)
  - in most cases, only servers have certificates
- Chain of trust: suitable also for authentication
  - e.g. in trusted networks

Hop-by-hop Encryption with IPSec

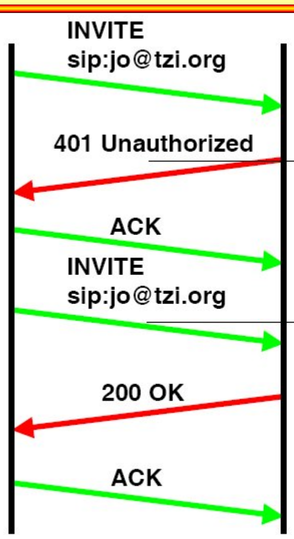


- Possible deployment scenario
  - IPSec between hosts inside an administrative domain
  - Established trust relationships, pre-shared keys
- Security functions independent of SIP layer

SIP Media Privacy

- **Encryption of (RTP) media streams**
  - use old RTP encryption scheme
  - use secure RTP (SRTP) profile
    - currently finalized within the IETF
- **Secure key distribution between endpoints *in a call***
- **Original SDP allows only one per media key field ("k=")**
- **SDP extensions for better keying support**
  - requires encrypted SDP in SIP message body
  - requires protected communication path
- **Further SDP extensions for secure media keying**
  - MIKEY allows for end-to-end negotiation of keys
  - protection of the exchanged information within SIP

Authentication: Example Call Flow



WWW-Authenticate: Digest realm="TZI", domain="sip:tzi.org", nonce="qf73…", stale=FALSE, algorithm=MD5

Authorization: Digest username="jo", realm="TZI", nonce="qf73…", response="50c6a6071bc8..."

- Security domain is identified by realm and original request URI
- Caller must create new request with incremented CSeq value
- Attention: parameters separated by "," — header-split not allowed!

Authentication for Proxies

- **Similar to endpoints (HTTP Digest)**
- **Proxy rejects client request with “407 Proxy Auth required”**
  - **Proxy-Authenticate**: header
  - multiple proxies along the path may challenge
- **Client resubmits request with credentials for proxy**
  - in **Proxy-Authorization**: header
  - multiple headers with credentials may need to be included

Agenda

- **Digitized Voice**
- **Introduction to Voice over IP**
- **RTP**
- **SIP Basics 1**
- **SIP Basics 2**
- **SIP in Detail**
- **H.323**

What is H.323?

- **H.323* is a multimedia conferencing protocol, which includes voice, video, and data conferencing, for use over packet-switched networks**

*H.323 is "ITU-T Recommendation H.323: Packet-based multimediacommunications systems"

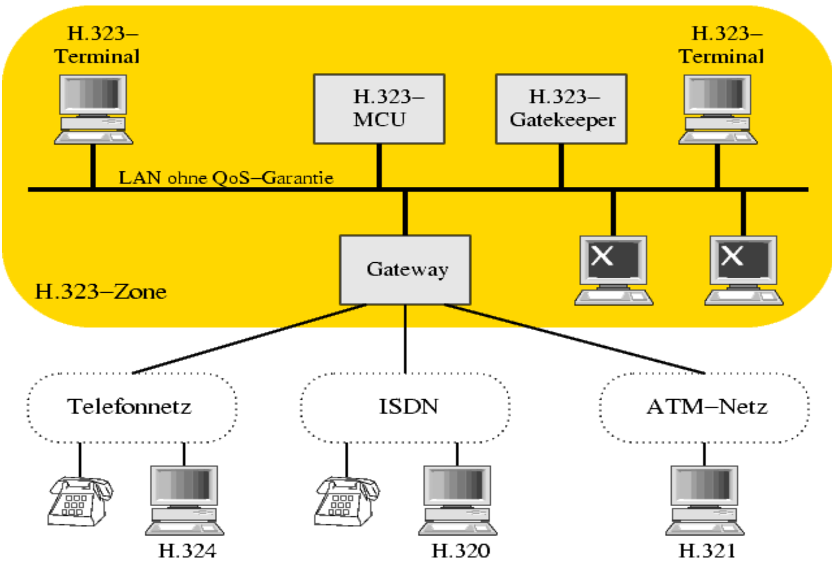
Who Defined H.323?

- **Recommendation H.323 is a standard published by the International Telecommunications Union Telecommunications Sector (ITU-T)**
  - Formerly known as CCITT
  - Refer to http://www.itu.int/ITU-T/

Base H.323 Documents

- H.323 – "Umbrella" document that describes the usage of H.225.0, H.245, and other related documents for delivery of packet-based multimedia conferencing services
- H.225.0 – Describes three signaling protocols (RAS, Call Signaling, and "Annex G")
- H.245 – Multimedia control protocol (common to H.310, H.323, and H.324)

H.323 System Components



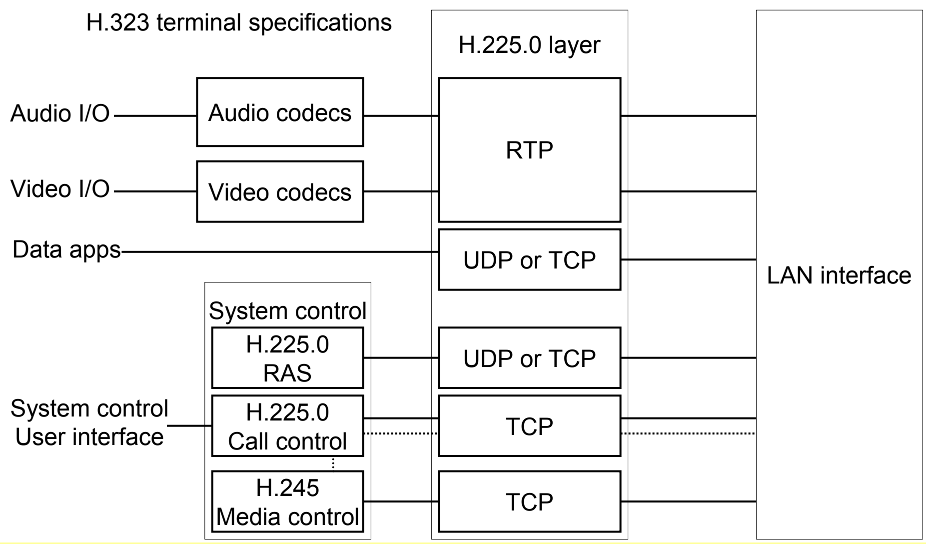
Elements of an H.323 System

- Terminals
- Multipoint Control Units (MCUs)
- Gateways
- Gatekeeper

Terminals

- Telephones
- Video phones
- IVR devices
- Voicemail Systems

Functional Elements of an H.323 Terminal



MCUs

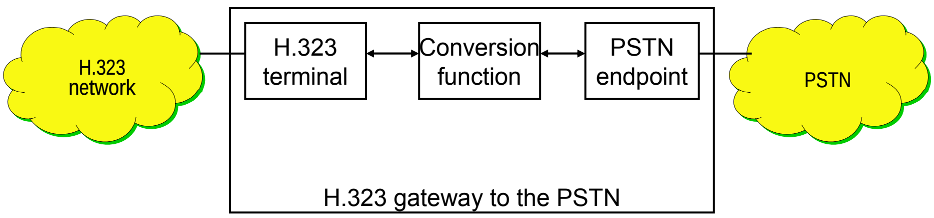
- **Components:**
  - Multipoint Controller
  - Mulitpoint Processor
- **Responsible for managing multipoint conferences (three or more endpoints engaged in a conference)**
- **The MCU contains a Multipoint Controller (MC) that manages the call signaling and may optionally have Multipoint Processors (MPs) to handle media mixing, switching, or other media processing**

Gateways

- The Gateway is composed of a "Media Gateway Controller" (MGC) and a "Media Gateway" (MG), which may co-exist or exist separately
- The MGC handles call signaling and other non-media-related functions
- The MG handles the media
- Gateways interface H.323 to other networks, including the PSTN, H.320 systems, other H.323 networks (proxy), etc.

 VoIP, v1.1 177

Logical structure of an H.323 Gateway



 VoIP, v1.1 178

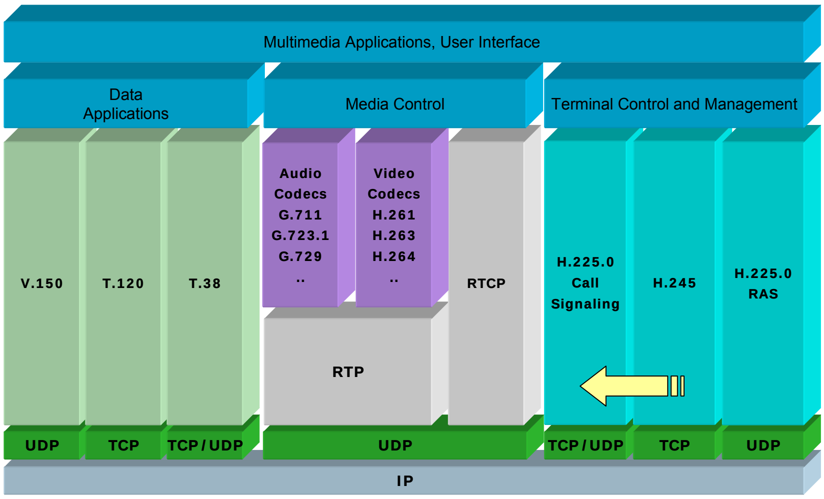
Gatekeeper

- Controls an H.323 zone
- The Gatekeeper is an *optional* component in the H.323 system which is used for admissions and bandwidth control and address translation
- The gatekeeper may allow calls to be placed directly between endpoints or it may route the call signaling through itself to perform functions such as follow-me/find-me, forward on busy, etc.

Addressing H.323/IP networks

- Network addresses and transport service access point (TSAP) identifiers
- H.323 aliases
- Alias-naming conventions for interzone communication
- Determining network addresses and TSAP identifiers

Typical H.323 Protocol Stack



H.323 Signaling

- **H.225.0 – RAS Registration, Admission, and Status between the endpoint and its Gatekeeper**
- **H.225.0 – Q.931 connection establishment and connection clearing**
- **H.245 provides “control” to the multimedia session that has been established**

RAS (H.225.0)

- Registration, Admission, and Status
- User between the endpoint and its Gatekeeper in order to
  - Allow the Gatekeeper to manage the endpoint
  - Allow the endpoint to request admission for a call
  - Allow the Gatekeeper to provide address resolution functionality for the endpoint
- RAS signaling is required when a Gatekeeper is present in the network (i.e., the use of a Gatekeeper is conditionally mandatory)

General Format of RAS

- RAS messages generally have three types
  - Request (xRQ)
  - Reject (xRJ)
  - Confirm (xCF)
- Exceptions are
  - Information Request / Response / Ack / Nak
  - The “nonStandardMessage”
  - The “unknownMessage” response
  - Request in Progress (RIP)
  - Resource Available Indicate / Confirm (RAI/RAC)
  - Service Control Indication / Response

RAS Port

- Typically, RAS communications is carried out via UDP through port 1719 (unicast) and 1718 (multicast)
  - For backward compatibility sake, an endpoint should be prepared to receive a unicast message on port 1718 or 1719
  - Only UDP is defined for RAS communications
- GRQ and LRQ may be send multicast, but are generally sent unicast
- All other RAS messages are sent unicast

Gatekeeper Request - GRQ

- When an endpoint comes to life, it should try to "discover" a gatekeeper by sending a GRQ message to a Gatekeeper
  - Address of a Gatekeeper may be provisioned
  - The endpoint may send a multicast GRQ
  - Address of a Gatekeeper may be found through DNS queries (Annex O/H.323)
- There may be multiple Gatekeepers that could service an endpoint, thus an endpoint should look through potentially several GCF/GRJ messages for a reply

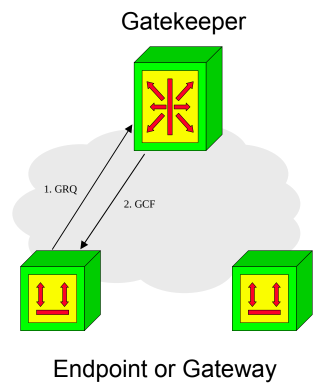
Gatekeeper Reject - GRJ

- If a Gatekeeper does not wish to provide service to the endpoint, it will generally send a GRJ message to the endpoint
  - As a security consideration to avoid DoS attacks, one might want to consider ignoring requests from unknown endpoints
- The GRJ message will carry one of several rejection reasons

Gatekeeper Confirm - GCF

- If the Gatekeeper wishes to provide service to the endpoint, it will return a GCF message
- The GCF message will contain a number of data elements that will later be used by the endpoint

Gatekeeper Discovery



Gatekeeper Registration - RRQ

- Once a Gatekeeper has been "discovered", the endpoint will then register with the Gatekeeper in order to receive services
- Communication is exclusively via port 1719 (unicast)
- Endpoint will send an RRQ and expect to receive either an RCF or RRJ
- Reception of an RRJ simply means that the endpoint will not receive services from the Gatekeeper, not that the endpoint cannot communicate on the network

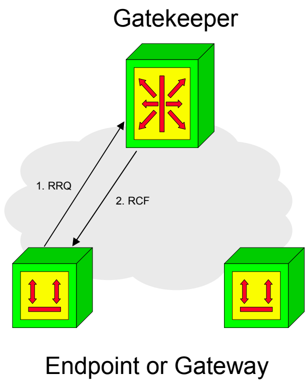
Gatekeeper Registration (cont.)

- During the registration process, the Gatekeeper will assign an "endpoint identifier" to the endpoint, which is to be used during subsequent communications with the Gatekeeper
- The endpoint will supply a list of endpoint alias addresses and the Gatekeeper will indicate which ones it accepts
- The Gatekeeper may grant the endpoint permission to place calls without using the ARQ/ACF exchange (called "pre-granted ARQs")
- The endpoint will indicate a "time to live" and the Gatekeeper may accept that or a lower TTL value

Lightweight RRQs (Registration reject)

- The "time to live" indicated in the RRQ tells the Gatekeeper when it may freely unregister the endpoint due to inactivity
- The endpoint may renew its registration by sending either a full RRQ message or a "lightweight RRQ" (LW RRQ)
- The LW RRQ message only contains a few elements and is only intended to refresh the endpoint's registration

Gatekeeper Registration



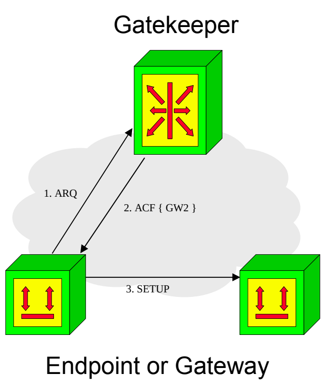
Admission Request - ARQ

- Once registered with a Gatekeeper, the endpoint may only initiate or accept a call after first requesting “admission” to the Gatekeeper via the ARQ message (except in the case that “pre-granted ARQs” is in use)
- The Gatekeeper may accept (ACF) or reject (ARJ) the request to place or accept a call
- The endpoint will indicate the destination address(es) and the Gatekeeper may (if “canMapAlias” is true) return an alternate set of destination addresses
- The endpoint uses a unique “call reference value” (CRV) between itself and the GK to refer to this call (link significant)

Admission Request (cont.)

- The endpoint will provide a Call Identifier (CallID), which is a globally unique value
- The endpoint will indicate a conference ID (CID), or 0 if the conference ID is not known
  - This is unique if the call is point to point
  - This value is shared by all participants in the same multipoint conference
  - Some devices do not properly handle CID=0
- The endpoint will indicate the desired bandwidth and the Gatekeeper may adjust that value to a lower value
- The endpoint will indicate whether it is originating or answering a call

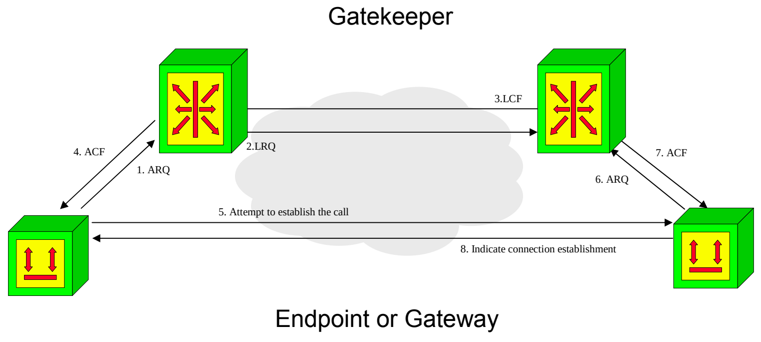
Admission Request (cont.)



Location Request - LRQ

- The LRQ message is sent by either an endpoint or a Gatekeeper to a Gatekeeper in order to resolve the address of an alias address (e.g., to turn a telephone number into an IP address)
- While LRQs may be sent by endpoints, they are almost exclusively sent by Gatekeepers

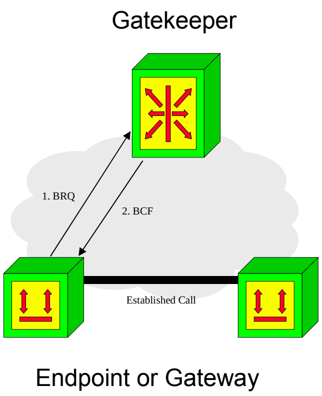
Location Request (cont.)



Bandwidth Request - BRQ

- Subsequent to initial call setup, the endpoint may wish to use more or less bandwidth than previously indicated via the BRQ
  - Note that, while it is syntactically legal for the GK to send a BRJ to a request asking for less bandwidth, this makes no sense and should not be done
- An endpoint must send a BRQ subsequent to initial call establishment if the actual bandwidth utilized is less than initially requested

Bandwidth Request (cont.)



Disengage Request - DRQ

- Once a call completes, the endpoint sends a DRQ message to the Gatekeeper
  - The Gatekeeper may send a DRJ, but this is strongly discouraged… if an endpoint is sending a DRQ, it means the call is over and cannot be “rejected”!
- The DRQ is an opportunity for the endpoint to report information useful for billing
- The Gatekeeper may also send a DRQ to force the endpoint to disconnect the call

Information Request - IRQ

- The IRQ is sent by the Gatekeeper to the endpoint to request information about one or all calls
- There are many details about each call that are reported to the Gatekeeper in the Information Response (IRR) message
- There are provisions in H.323 to allow the endpoint to provide call information periodically and unsolicited
- The Gatekeeper may acknowledge or provide negative acknowledgement to an unsolicited IRR

Request In Progress - RIP

- A RIP message may be sent by the endpoint or the Gatekeeper to acknowledge receive of a RAS message that cannot be responded to in normal processing time

Resource Availability - RAI

- The "Resource Available Indicate" (RAI) message is sent by an endpoint to indicate when it has neared resource limits or is no longer near a resource limit

- The Gatekeeper replies with "Resource Available Confirm" (RAC)

Service Control Indication - SCI

- This message is sent by either the endpoint or the Gatekeeper to invoke some type of service
- The responding entity replies with “Service Control Response” (SCR)
- The SCI/SCR messages are used for specific services that are and will be defined for H.323, including Gatekeeper requested tones and announcements and “stimulus control” (Annex K/H.323)

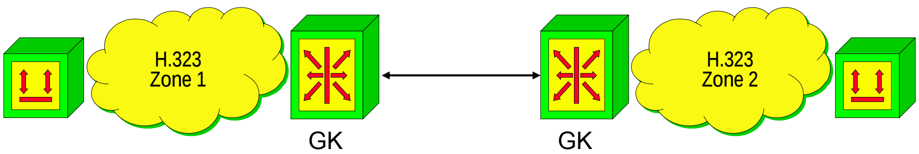
Miscellaneous Messages

- “Unknown Message Response” is sent to an unrecognized message
- “Non-Standard Message” is used to allow Gatekeepers and endpoints to exchange messages that are not standard

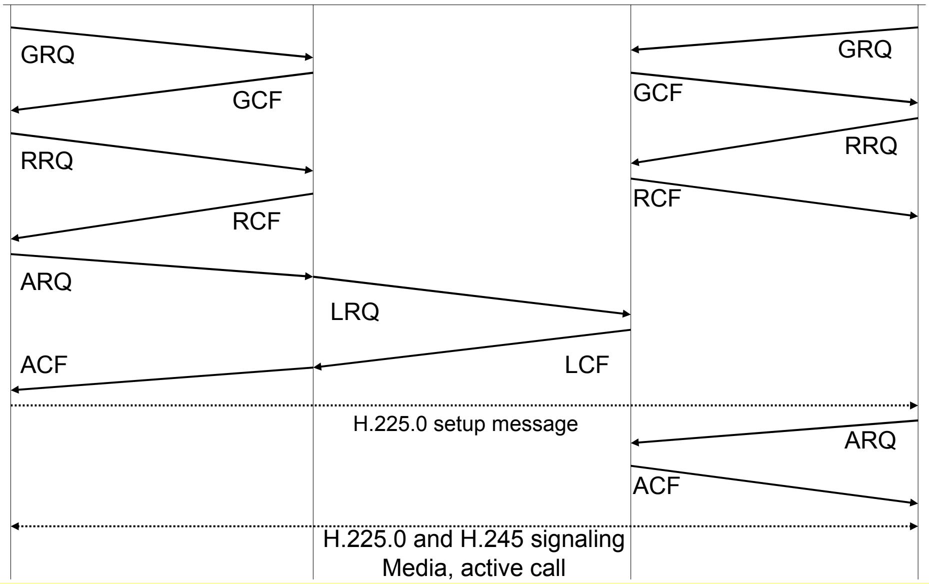
RAS Timers and Retries

RAS message	Time-out value (s)	Retry count
GRQ	5	2
RRQ	3	2
URQ	3	1
ARQ	5	2
BRQ	3	2
IRQ	3	1
IRR	5	2
DRQ	3	2
LRQ	5	2
RAI	3	2
SCI	3	2

H.225.0 RAS Exchange



H.225.0 RAS Exchange (cont.)

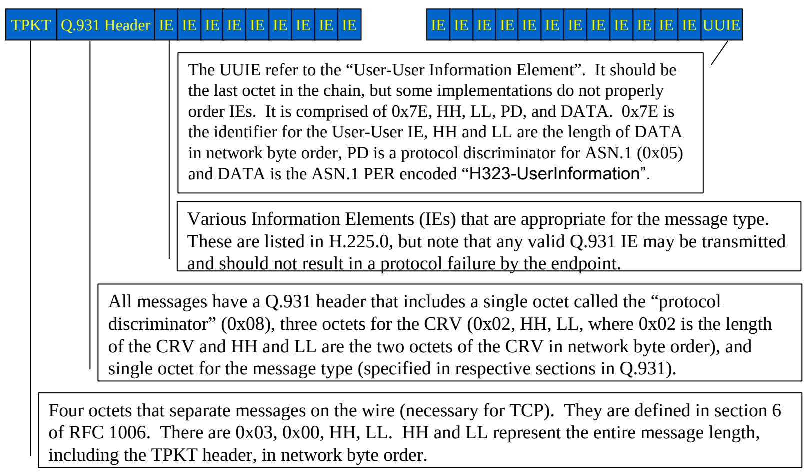


H.225.0 Call Signaling

Introduction

- H.225.0 Call Signaling is used to establish calls between two H.323 entities
- It was derived from Q.931 (ISDN call signaling), but was modified to be suitable for use on a packet based network
- ASN.1 was added to augment to Q.931 information and is stored in the "User to User" Information Element from Q.931
- H.225.0 also borrows messages from Q.932

H.225.0 Call Signaling Message



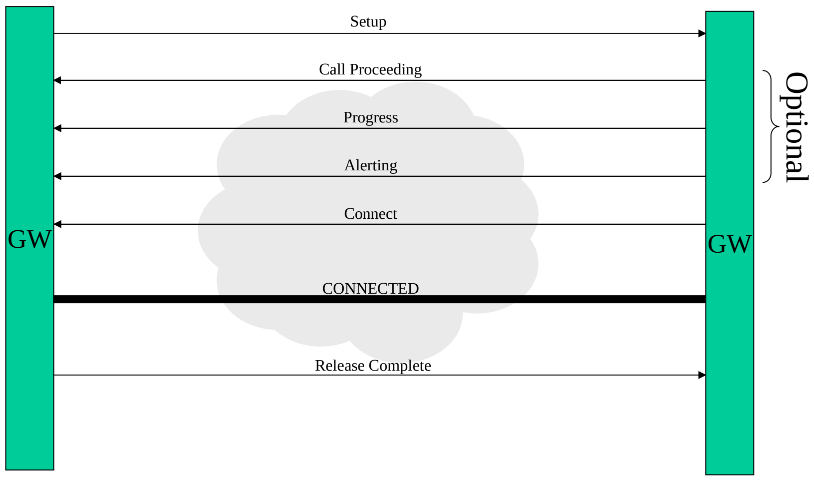
Information Elements

- Information elements carry additional information related to the specific message
- For example, SETUP has, among other things, a "Calling Party Number" IE, "Called Party Number" IE, "Display" IE, etc.
- Every H.225.0 message has a UUIE, though this is not true of Q.931
- H.225.0 made a number of changes to Q.931 and should be the guiding document

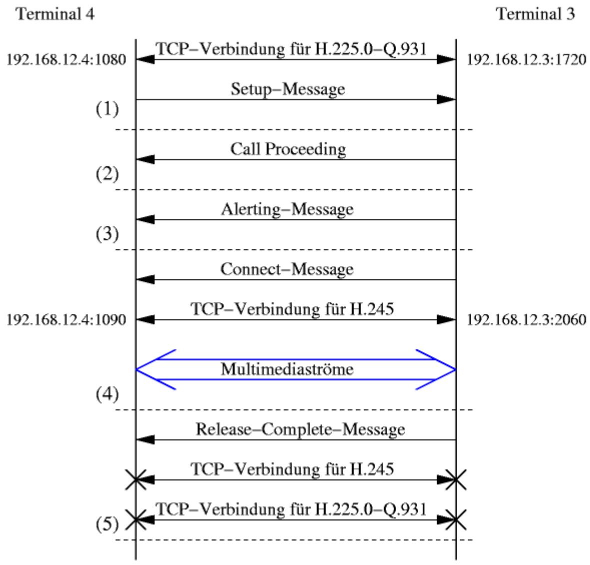
H.225.0 Call Signaling Messages

- Setup
- Call Proceeding
- Alerting
- Information
- Release Complete
- Facility
- Progress
- Status
- Status Inquiry
- Setup Acknowledge
- Notify
- Connect

Basic Call Setup Signaling



Basic Call Setup Signaling - Example



Comments on Call Establishment

- The basic call setup procedures are pretty straight forward
- The setup procedure can be as simple as "Setup" and "Connect"
- Intermediate messages (labeled as optional on the previous slide) are generally useful to prevent timeout errors and to provide in-band tones and announcements

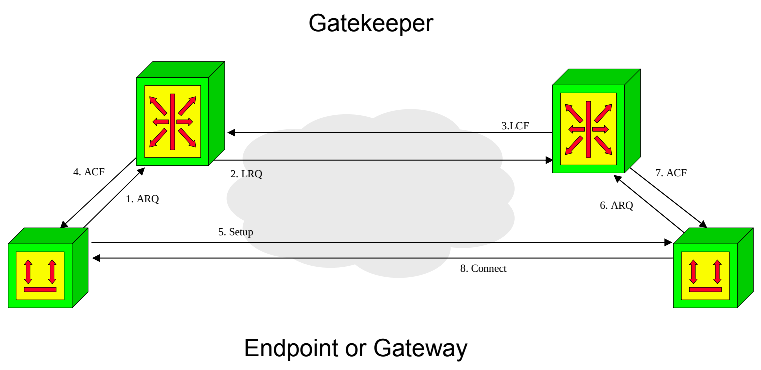
 VoIP, v1.1 217

Progress Message and Progress Indicator

- When a user places a call, he or she expects to hear a ringing tone or an announcement providing some information about why the call failed
- These "in-band tones and announcements" are provided by using the Progress message and the Progress Indicator IE (PI)
- Section 8.1.7.4/H.323 describes this more fully

 VoIP, v1.1 218

RAS and H.225.0 Call Signaling



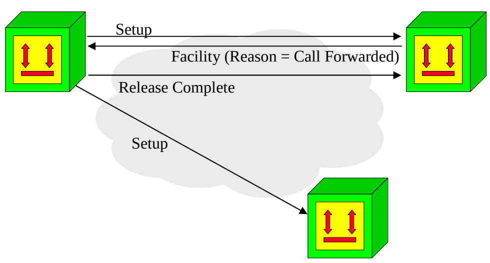
Overlapped Sending

- In some cases, the user may not have entered a complete telephone number
- Overlapped sending allows the calling endpoint to provide additional dialed digits to the called endpoint during the call establishment procedure
- Overlapped sending is generally most useful in H.225.0 Call Signaling, but RAS also allows for overlapped sending (refer to 8.1.12/H.323 for details)

Call Forwarding

- A Facility message with reason "callForwarded" allows for simple call redirection
- The H.323 standard states that this shall only be used for forwarding a call prior to "connect"
- In reality, many vendors use it as a lightweight means of performing a call transfer operation
- H.450.2 more fully describes a call transfer mechanism for H.323 systems

Call Forwarding (cont.)



H.245 media control

Comments on H.245

- H.245 is a protocol shared by a number of H.32x series protocols, including H.324M, which is used for multimedia conferencing within 3GPP wireless networks
- Like Q.931, not everything inside H.245 is applicable to H.323
- Refer to Annex A/H.323 for H.245 messages used by H.323 endpoints
- There are a *lot* of H.245 messages… but don't let that scare you
- H.245 signaling is intended to be carried out *in parallel* to H.225.0 signaling and preferably before the CONNECT message… waiting for the CONNECT will delay media establishment and result in media clipping

Purpose

- **H.245 provides "control" to the multimedia session that has been established**
  - Terminal capability exchange
  - Master/Slave determinations
  - Logical channel signaling
  - Conference control

H.245 Control Channel

- **H.245 messages are carried via a special "channel" called the H.245 Control Channel**
- **Opening the H.245 Control Channel is optional**
- **The H.245 channel is often a separate TCP connection, but it may be "tunneled" inside of the H.225.0 Call Signaling Channel**
- **When using UDP for call signaling, the H.245 Control Channel *must* be tunneled inside the H.225.0 call signaling channel**

H.245 Message

TPKT | H.245 PDU | H.245 PDU

H.245 PDU | H.245 PDU

Additional H.245 PDUs may be encoded following the first one. However, many implementations cannot handle this and, as such, it is ill-advised to place them end-to-end like this. It is strongly recommended to place only one between each TPKT header, but do be prepared for the case that more than one PDU does exist following TPKT

H.245 messages are encoded in ASN.1 PER and follow the TPKT header in the H.245 Control Channel.

Four octets that separate messages on the wire (necessary for TCP). They are defined in section 6 of RFC 1006. There are 0x03, 0x00, HH, LL. HH and LL represent the entire message length, including the TPKT header, in network byte order.

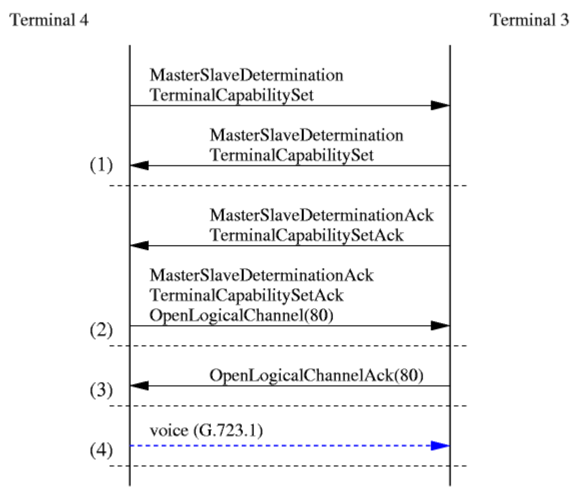
H.245 Tunneling

- **H.245 is generally transmitted on a separate TCP connections by most older endpoints**
- **Newer endpoints generally support “H.245 Tunneling”, which is the ability to place the H.245 inside the H.225.0 Call Signaling channel**

Four H.245 Message Types

- **Request**
  - masterSlaveDetermination
  - terminalCapabilitySet
- **Response**
  - masterSlaveDeterminationAck
  - terminalCapabilitySetAck
- **Command**
  - sendTerminalCapabilitySet
- **Indication**
  - userInput

H.245 Messages



Capabilities Exchange

- The capability exchange (or “caps exchange”) allows two endpoints to exchange information about what media capabilities they possess, such as G.711, G.723, H.261, and H.263
- Along with the type of media, specific details about the maximum number of audio frames or samples per packet is exchanged, information about support for silence suppression (VAD), etc. are exchanged
- Using this capability information, endpoints can select preferred codecs that are suitable to both parties
- The terminalCapabilitySet (TCS) must be the first message transmitted on the H.245 Control Channel

Capabilities are Numbered

- Each capability is numbered in a “capability table”
- All attributes (VAD, frames/packet, etc.) are part of the the capability in the table

Sample Capability Table

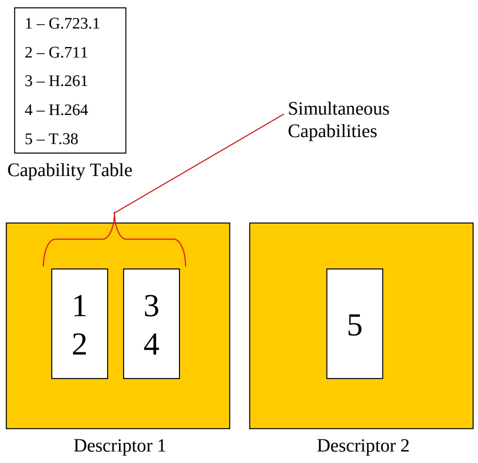
1 – G.723.1
2 – G.711
3 – H.261
4 – H.264
5 – T.38

Simultaneous Capabilities

- When endpoints advertise capabilities, they also advertise which capabilities may be performed simultaneously
- It may not be possible, due to bandwidth limits, to open a high bit-rate video codec at the same time as a high bit-rate audio codec

Capability Descriptors

- The capability descriptor contains the sets of simultaneous capabilities
- One descriptor may be used at a time (i.e., capabilities from descriptor 1 and descriptor 2 may not be used simultaneously)



Master Slave Determination

- Once capabilities are exchanged, the endpoints negotiate master and slave roles
  - Actually the master/slave messages can be sent along with the TCS message (terminalCapabilitySet)
- The master in a point to point conference really only has the power to indicate when channels are in conflict (e.g., when one the other terminal tries to open a channel that is not compatible)
- The slave device must yield to the requests of the master device and reconfigure channels appropriately

Logical Channel Signaling

- Channels are opened by exchanging “openLocalChannel” (OLC) messages
- The OLC will contain one of the capabilities that was previously advertised by the other endpoint
- Voice and video channels are “unidirectional”, so each end must transmit an OLC to open a logical channel

Logical Channel Signaling (cont)

- **Within the OLC, a "session ID" is assigned**
- **Session 1 is the default audio session, 2 is the default video session, and 3 is the default data session**
- **Additional session IDs may be used, but are assigned by the master in the call**
- **There is a relationship between H.245 sessions IDs and RTP: OLCs with the same session ID are considered to be part of the same RTP/RTCP session**

Closing the H.245 Control Channel

- **H.323 specifies that, in order to close the H.245 Control Channel, the endpoint must:**
  - Close all open logical channels
  - Wait for all acknowledgement messages
  - Send an "endSession" command
  - Wait for an "endSession" from the other side
- **In reality, most endpoint vendors don't bother– they just use the H.225.0 Release Complete command to terminate the call and close the H.245 Control Channel, as that is much more efficient**