VoIP & SIP

Summary

- What is VoIP
- Real Time Protocol
- SIP
  - Servers, proxies, registrars
  - NAT traversal
  - mobility
  - Android support

Voice over IP (VoIP)

Sinnreich
Ch 18

- **Did you know that :**
  - **Most telephony today is transported with VoIP?**
  - **Most PBX-es installed use VoIP?**
  - **In Ethiopia, Oman VoIP use is a criminal offence?**
  - **Hangouts, Whatsapp, Skype, TeamSpeak, TeamViewer, Viber, Yahoo Mesg all use VoIP?**

Seven Myths About VoIP

1. VoIP is free
2. The only difference between VoIP and regular telephony is the price
3. Quality of service isn’t an issue nonadways, because there’s plenty of bandwidth in the network
4. VoIP can’t replace regular telephony, because it still can’t guarantee quality of service
5. VoIP is just another data application
6. VoIP isn’t secure
7. A Phone is a Phone is a Phone

What is VoIP?

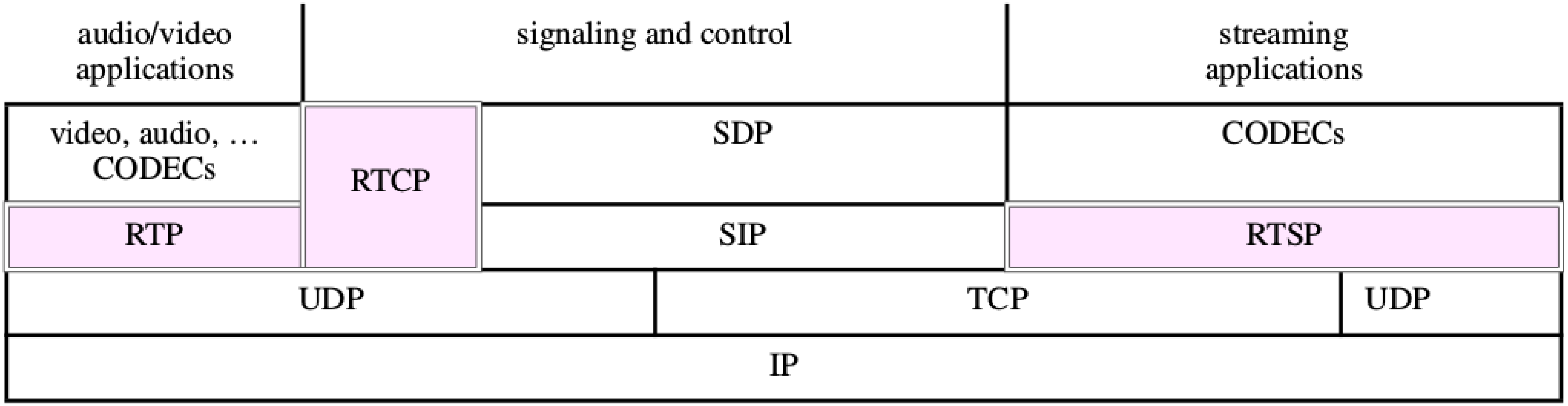
- VoIP is an end-to-end architecture
  – Voice transported in IP packets
- Comparison with PSTN
  – Circuit switch vs. Packet switch
  – Latency
  – Dataplane, control plane
  – Mobility
- VoIP headsets
  – Physical, Software
  – Built into Android

Packet Encapsulation

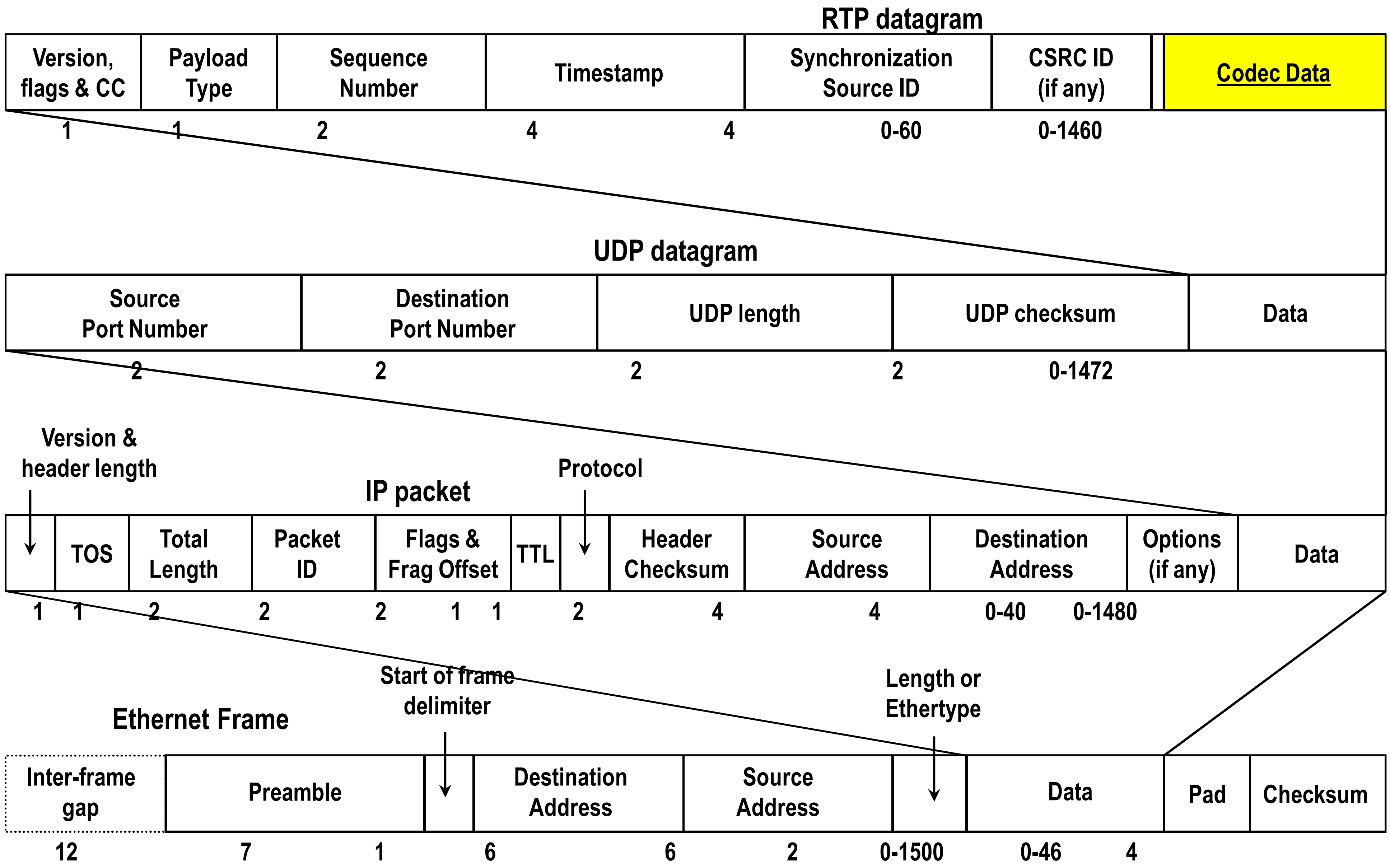
- Encapsulate 10-20ms of speech in a packet

IP	UDP	RTP	Voice
20	8	12	20-240

- RTP, RTCP, RTSP

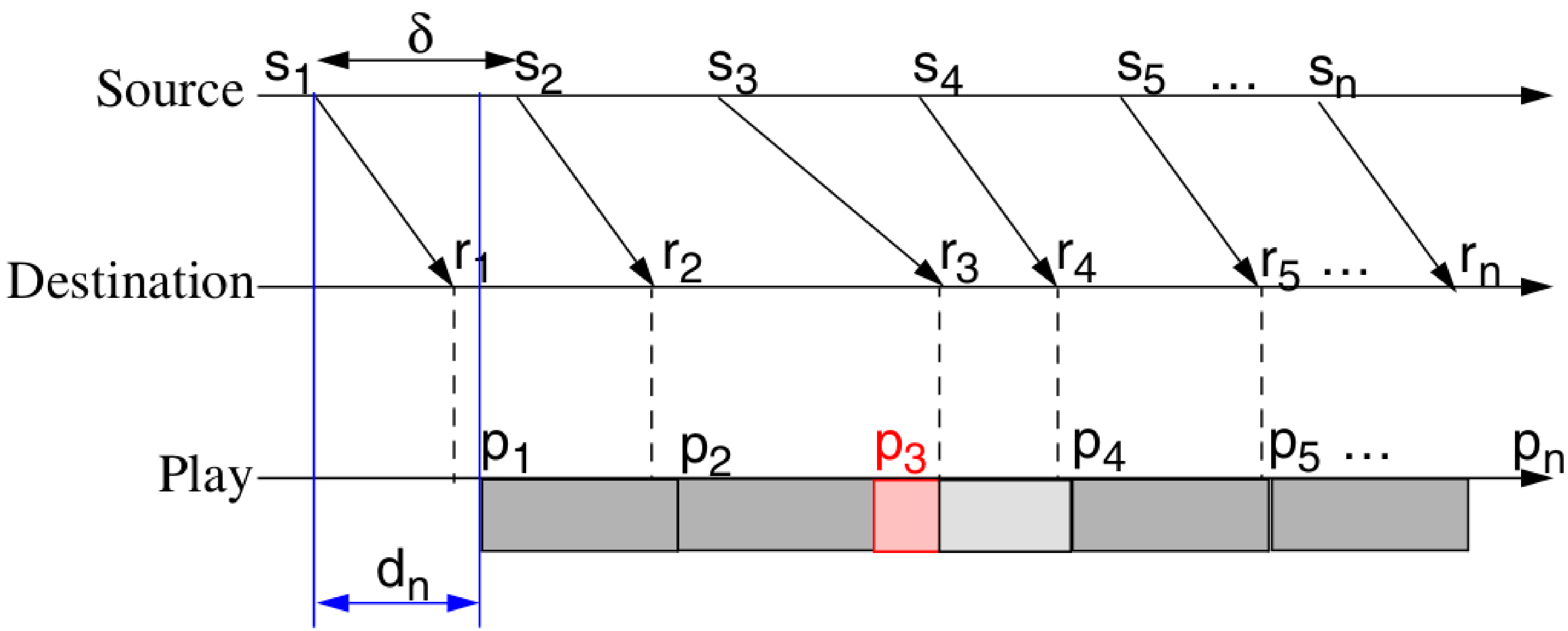


Packet Encapsulation



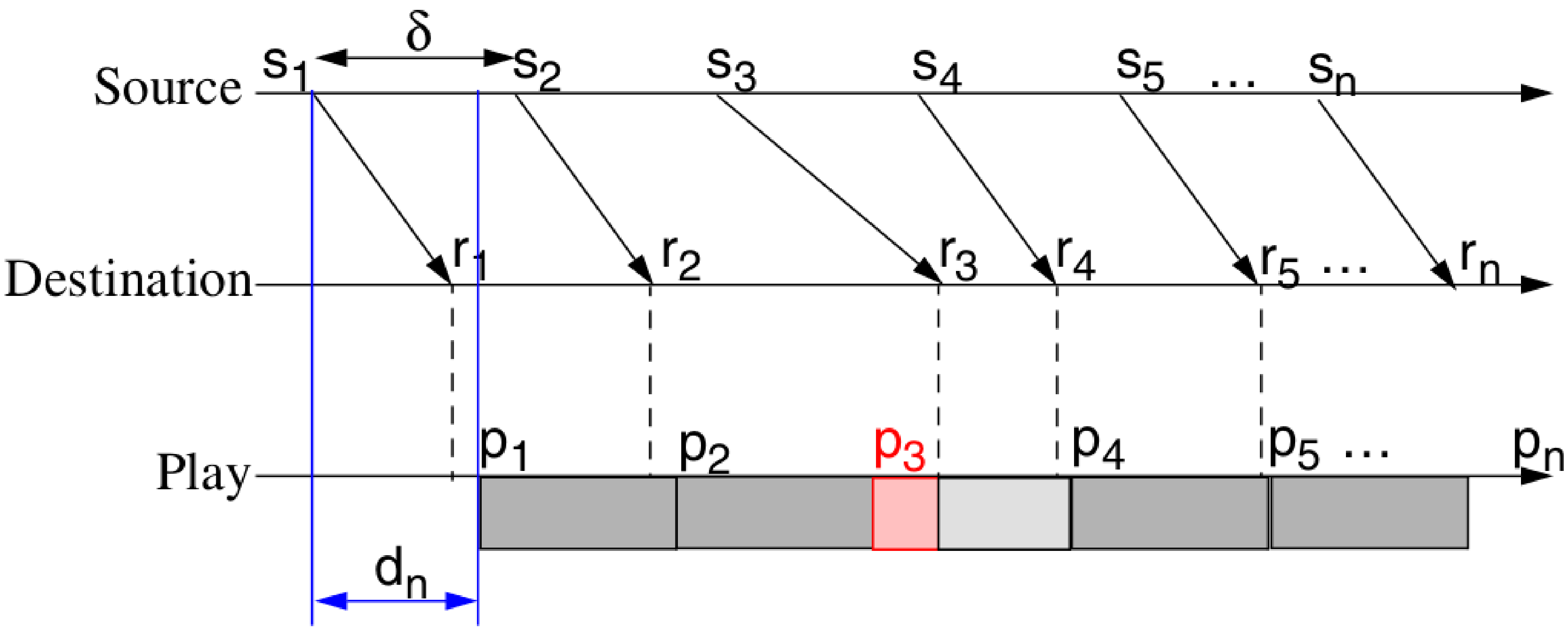
Realtime delivery

- Realtime app = maintain time relationship at receiver
  - Play in same order as original (sequence number)
  - Play time to reproduce original (time stamp)
  - Once decided p1, <u>all packets have deadlines</u>!



Jitter = $(r_i - r_{i-1}) - (s_i - s_{i-1})$

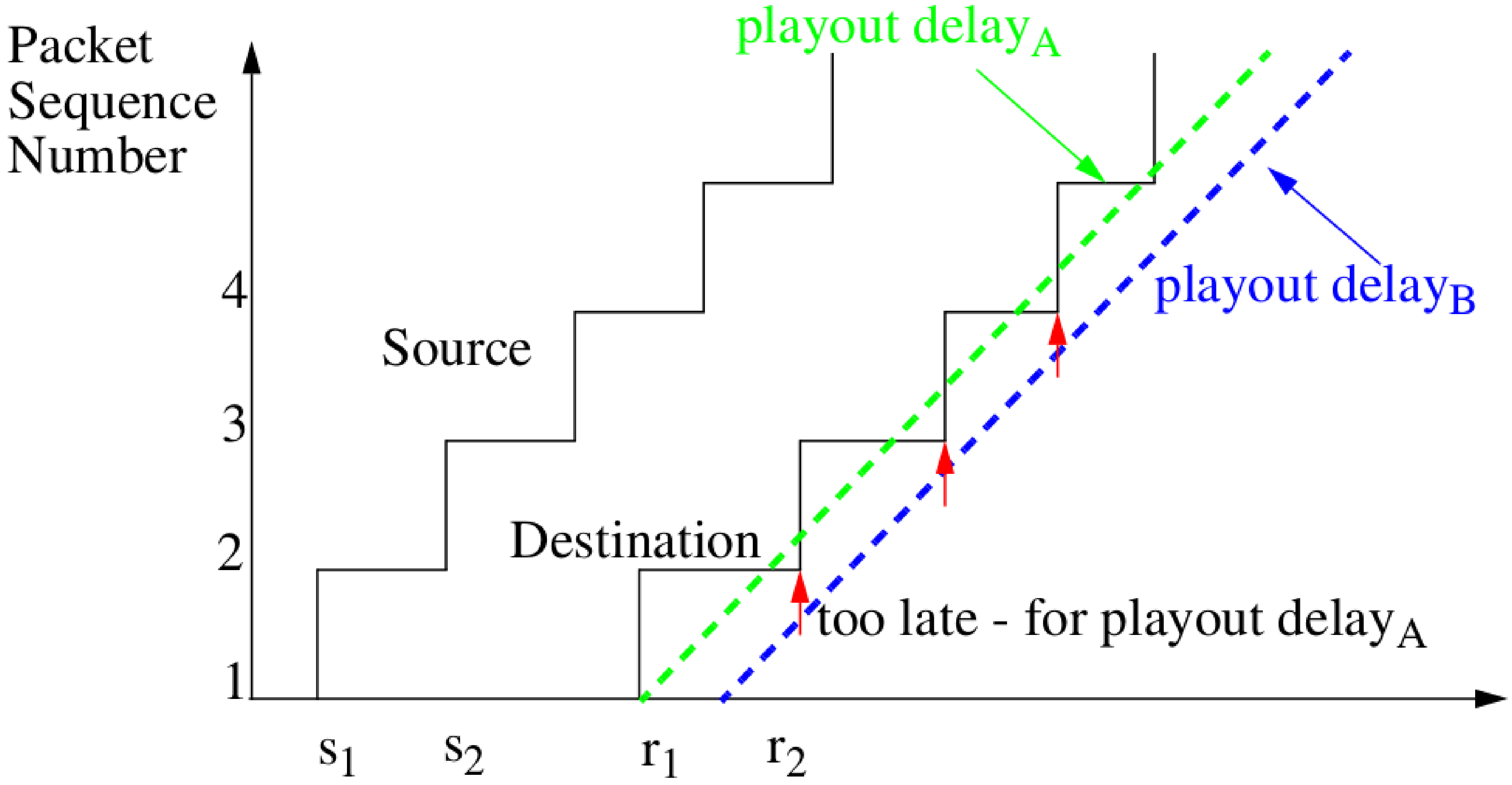
Realtime delivery



- **What is jitter?**
  - **Packet delay variance = $(r_i - r_{i-1}) - (s_i - s_{i-1})$**
  - **Negative jitter: late packet**
  - **Positive jitter: early packet**
- **How to shield listener from jitter?**
  - **Playout buffer (extra delay)**

Dealing with jitter

- **Playout buffer = delay at receiver to smooth jitter**



Delay and jitter

- Audio end-to-end delay components
  - encoding
  - packetization
  - transmission
  - propagation, switching/routing
  - receiving
  - decoding
  - dejittering
  - playing

sender

network

receiver

playout buffer ADDS delay

Voice quality metrics

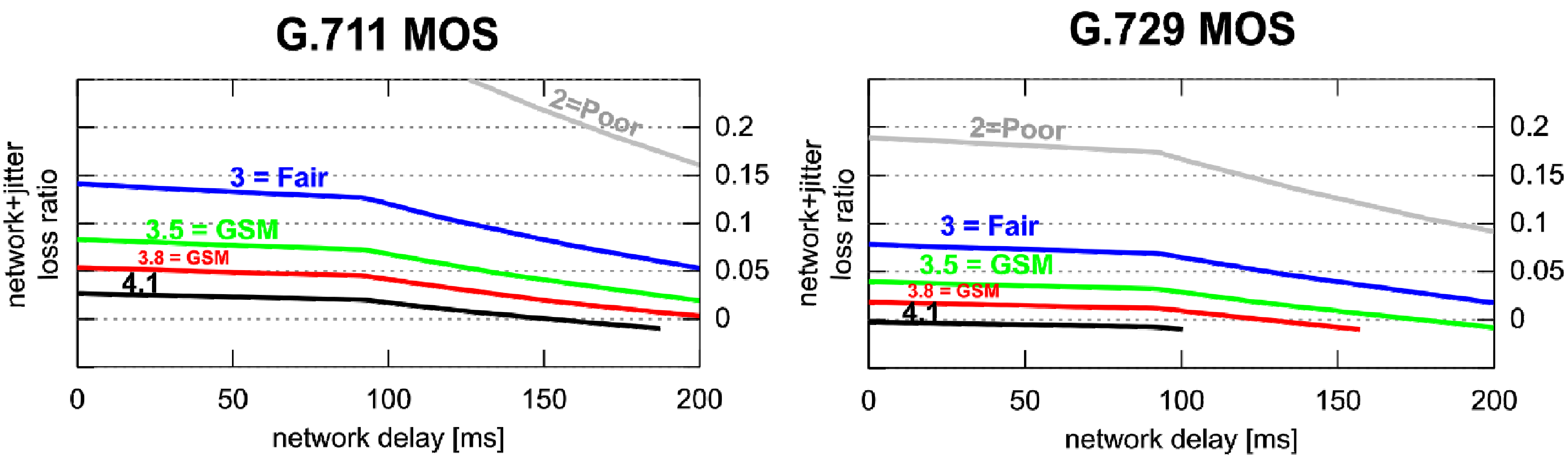
- Listening quality
- Conversational quality
- Network quality
  - Delay, loss, jitter
  - Delay limits
    - ← < 150ms acceptable
    - ← < 400ms tolerable
    - ← > 400ms unacceptable
- Mean Opinion Score (MOS)
  - Excellent = 5, Good = 4, Fair = 3, Poor = 2
  - Functions derived using human listeners to assign MOS to a given (loss,delay) conversation[1]

[1] Cole , Rosenbluth "Voice over IP Performance Monitoring", http://ccr.sigcomm.org/archive/2001/apr01/ccr-200104-cole.pdf

voice codecs

Codec	bitrate	Framesize [ms]	codec Delay[ms]	MOS ideal cond.
G.711	64kbps	10, 20,30	25	4.1
G.729	8kbps	10, 20,30	15, 25,35	3.92
GSM-FR	14kbps	22.5	20	3.5
SILK (skype)	6-40Kbps	20	?	5

MOS(delay, loss)



- Conditions: 25ms vocoder delay, 60ms playout buffer
- Used known MOS(delay,loss) functions to generate curves
- G.729 = high compression, less resilient to loss
- G.711 = needs more bandwidth, more loss resilient

Session Initiation Protocol (SIP)

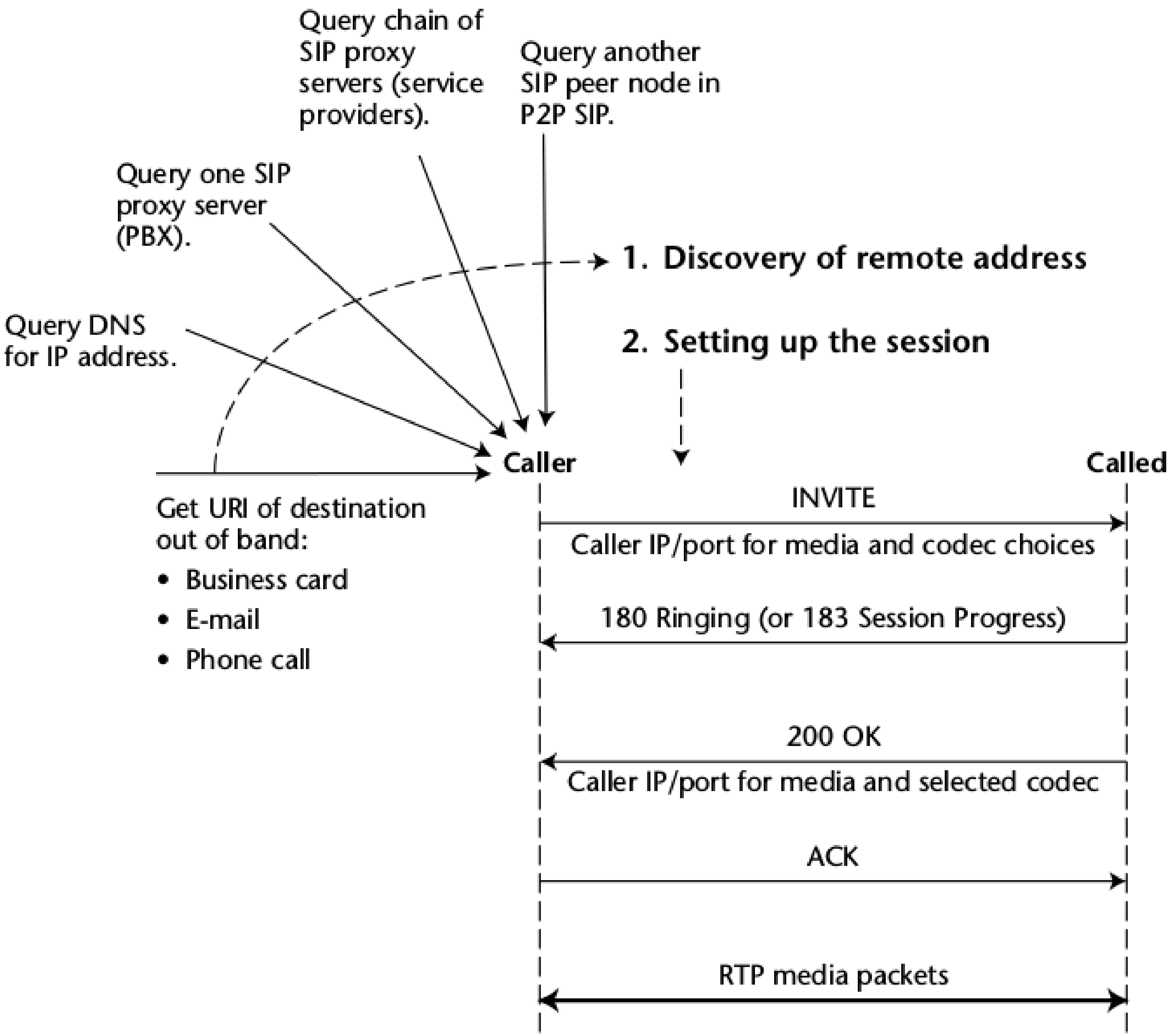
Sinnreich
Ch 6

- RFC 3261 , RFC 3853, RFC 4320
- Text based protocol, similar to HTTP, SMTP
- Locate user given email-style address
- Setup session
- negotiate call parameters
- Personal mobility: different terminal, same identifier
- SIP does not use RTP, a session does
- SDP (Session Description Protocol) – info about call, encoding, ports

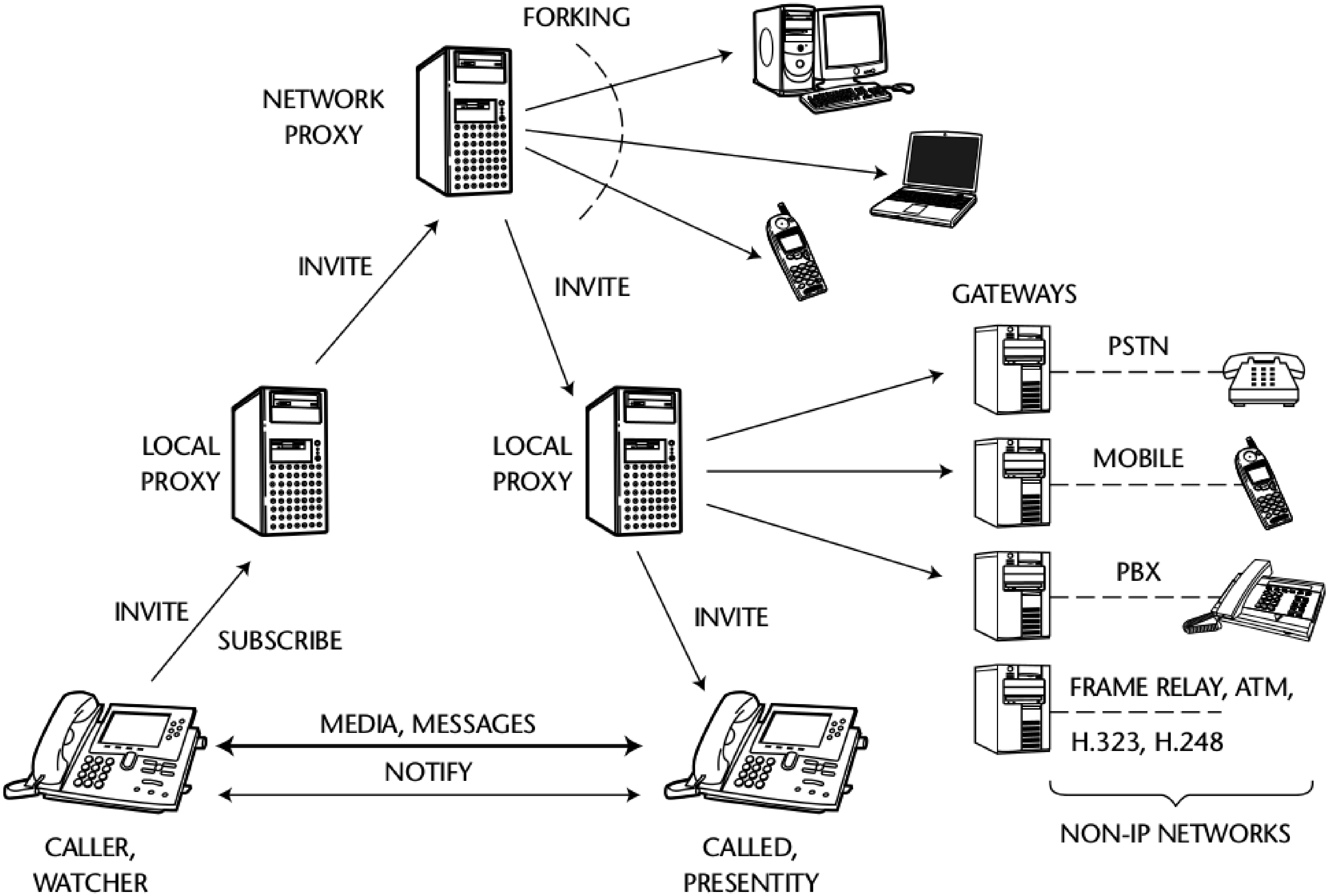
SIP components

- UAC: user-agent client (caller application)
- UAS: user-agent server: accept, redirect, refuse call
- redirect server: redirect requests
- proxy server: server + client
- registrar: track user locations
- user agent = UAC + UAS
- often combine registrar + (proxy or redirect server)

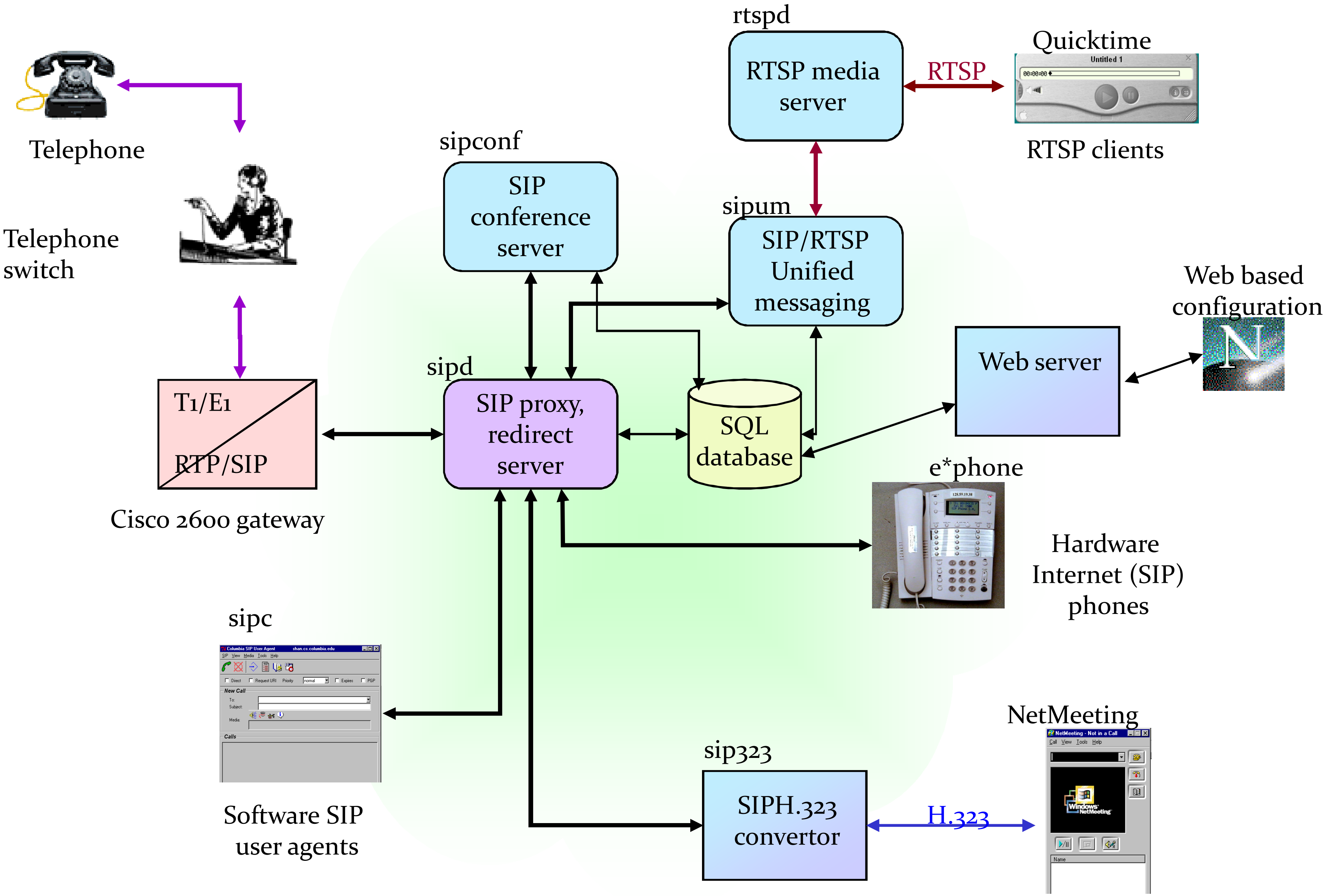
SIP in a nutshell



SIP enabled IP network



SIP-based Architecture

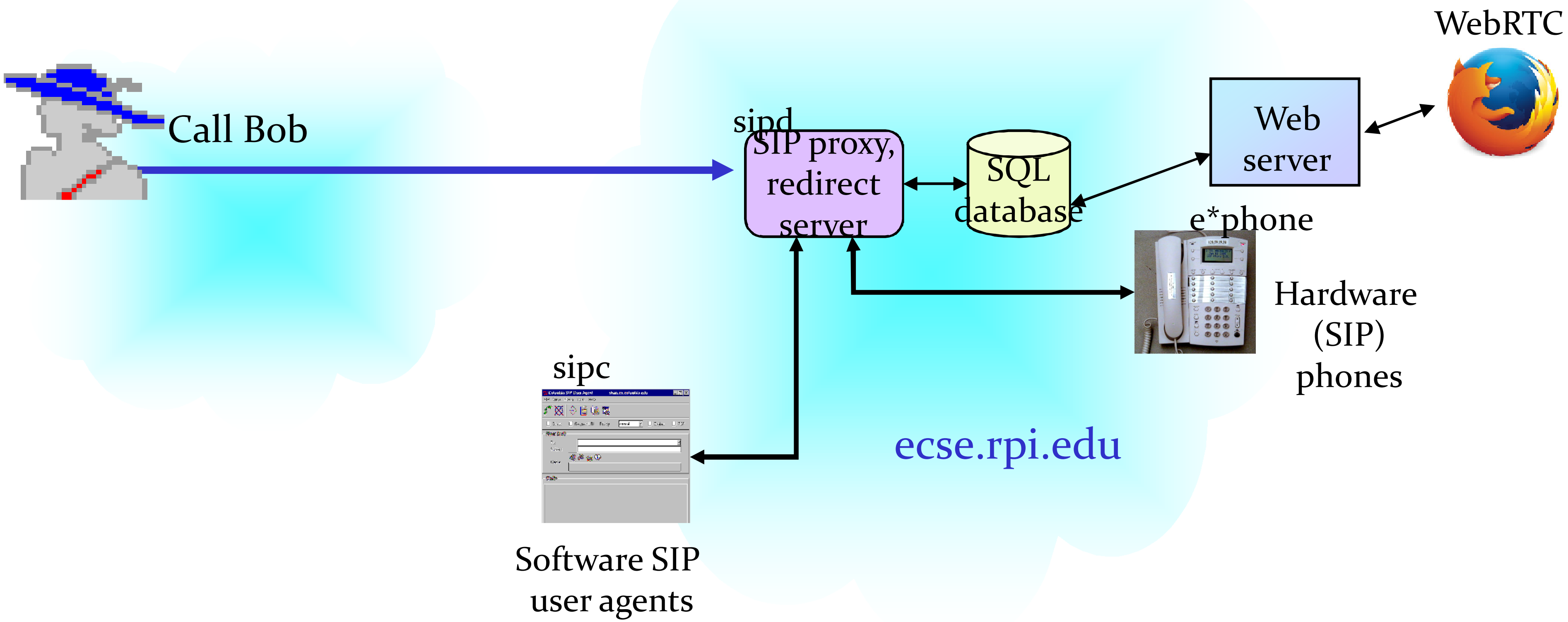


Example Call

- Bob signs up for the service from the web as "bob@ecse.rpi.edu"
- He registers from multiple phones

- Alice tries to reach Bob
  `INVITE ip:Bob.Wilson@ecse.rpi.edu`

- sipd canonicalizes the destination to `sip:bob@ecse.rpi.edu`
- sipd rings both e*phone and sipc
- Bob accepts the call from sipc and starts talking



SIP Sessions

- “Session”: exchange of data between an association of participants
- Users may move between endpoints
- Users may be addressable by multiple names
- Users may communicate in several different media
- SIP: enables internet endpoints to
  - Discover each other
  - Characterize the session
- Location infrastructure: proxy servers, invite/register…
  - Name mapping and redirection services
- Add/remove participants from session
- Add/remove media from session

SIP Capabilities

- User location: determination of the end system to be used for communication;
- User availability: determination of the willingness of the called party to engage in communications;
- User capabilities: determination of the media and media parameters to be used;
- Session setup: "ringing", establishment of session parameters at both called and calling party;
- Session management: including transfer and termination of sessions, modifying session parameters, and invoking services.

What SIP is not...

- SIP is not a vertically integrated communications system.
  - It is a component in a multimedia architecture.
- SIP does not provide services.
  - Rather, SIP provides primitives that can be used to implement different services.
  - For example, SIP can locate a user and deliver an opaque object to his current location.
- SIP does not offer conference control services
  - ... such as floor control or voting
  - SIP does not prescribe how a conference is to be managed.

SIP Structure

- 3 "layers", loosely coupled, fairly independent processing stages
- Lowest layer: *syntax*, encoding (augmented BNF)
- Second layer: *transport* layer.
  - Defines how a client sends requests and receives responses and how a server receives requests and sends responses over the network.
- Third layer: *transaction* layer.
  - A transaction is a request sent by a client transaction (using the transport layer) to a server transaction …
  - … along with all responses to that request sent from the server transaction back to the client.
  - The transaction layer handles application-layer retransmissions, matching of responses to requests, and application-layer timeouts

- The layer above the transaction layer is called the transaction user (TU).

SIP Design Choices

Transport protocol neutrality: run over reliable (TCP, SCTP) and unreliable (UDP) channels, with minimal assumptions

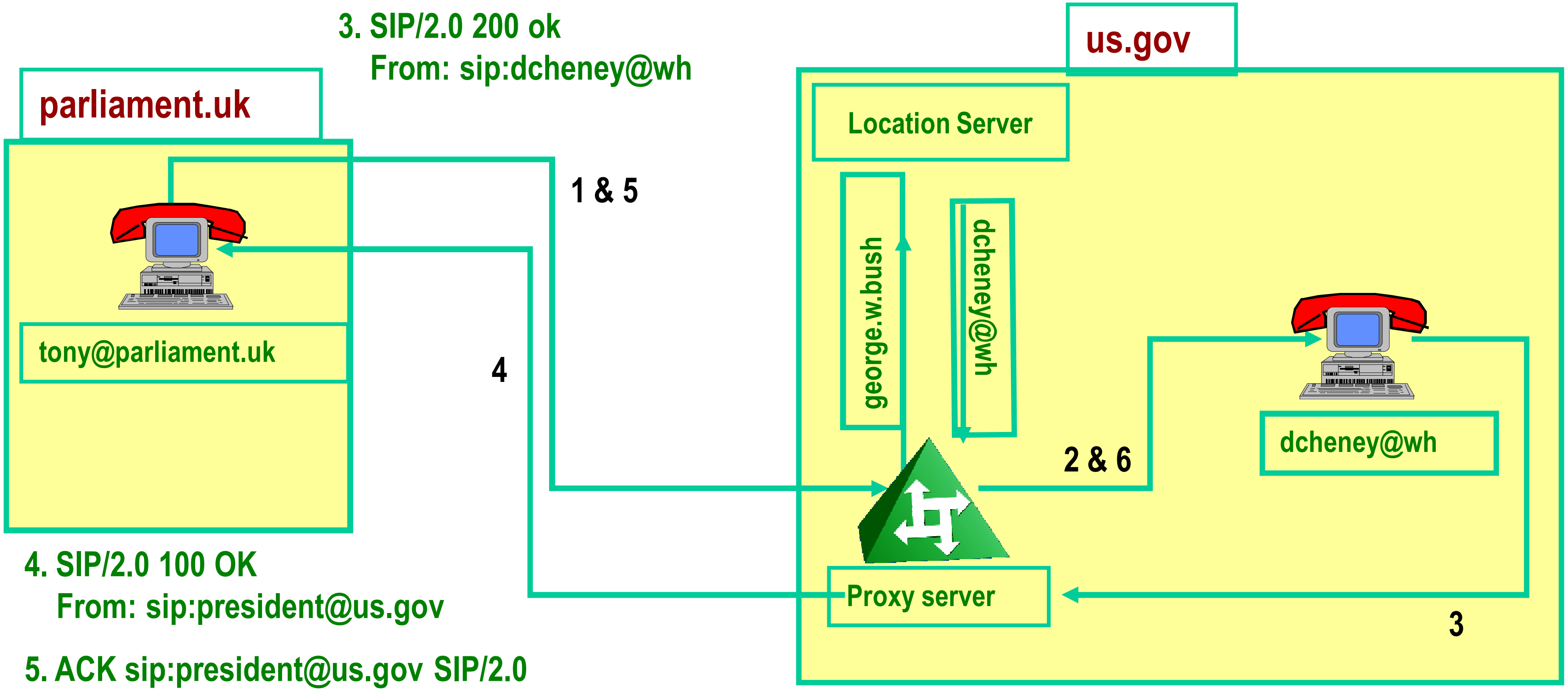
Request routing: direct (performance) or proxy-routed (control)

Separation signaling vs. media description: can add new applications or media types, SDP $\longrightarrow$ SDPng

Extensibility: indicate and require proxy and UA capabilities

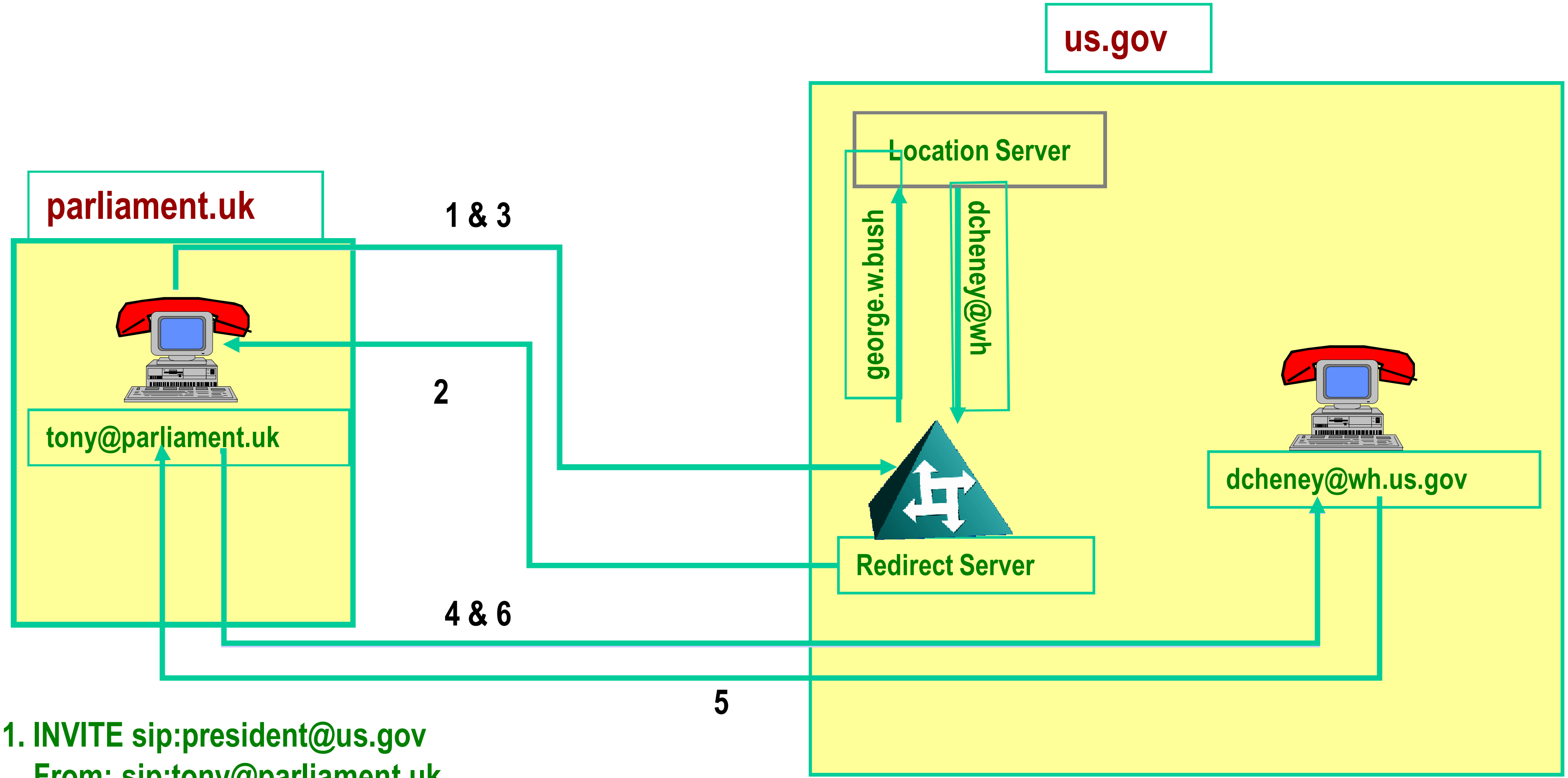
Proxy Server

1. INVITE sip:president@us.gov SIP/2.0
   From: sip:tony@parliament.uk
2. INVITE sip:dcheney@wh SIP/2.0
   From: sip:tony@parliament.uk
3. SIP/2.0 200 ok
   From: sip:dcheney@wh



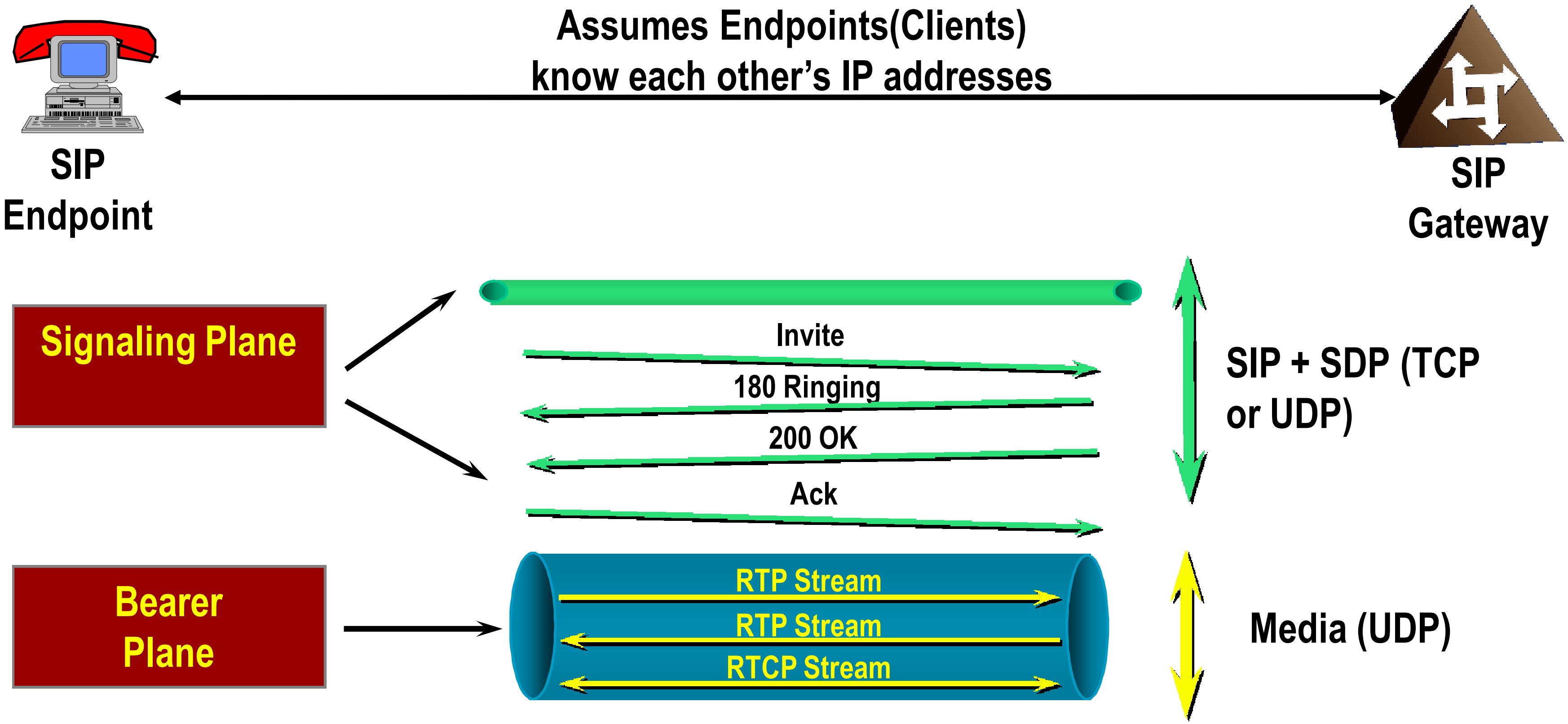
4. SIP/2.0 100 OK
   From: sip:president@us.gov
5. ACK sip:president@us.gov SIP/2.0
   From: sip:tony@parliament.uk
6. ACK sip:dcheney@wh SIP/2.0
   From: sip:tony@parliament.uk

Redirect Server

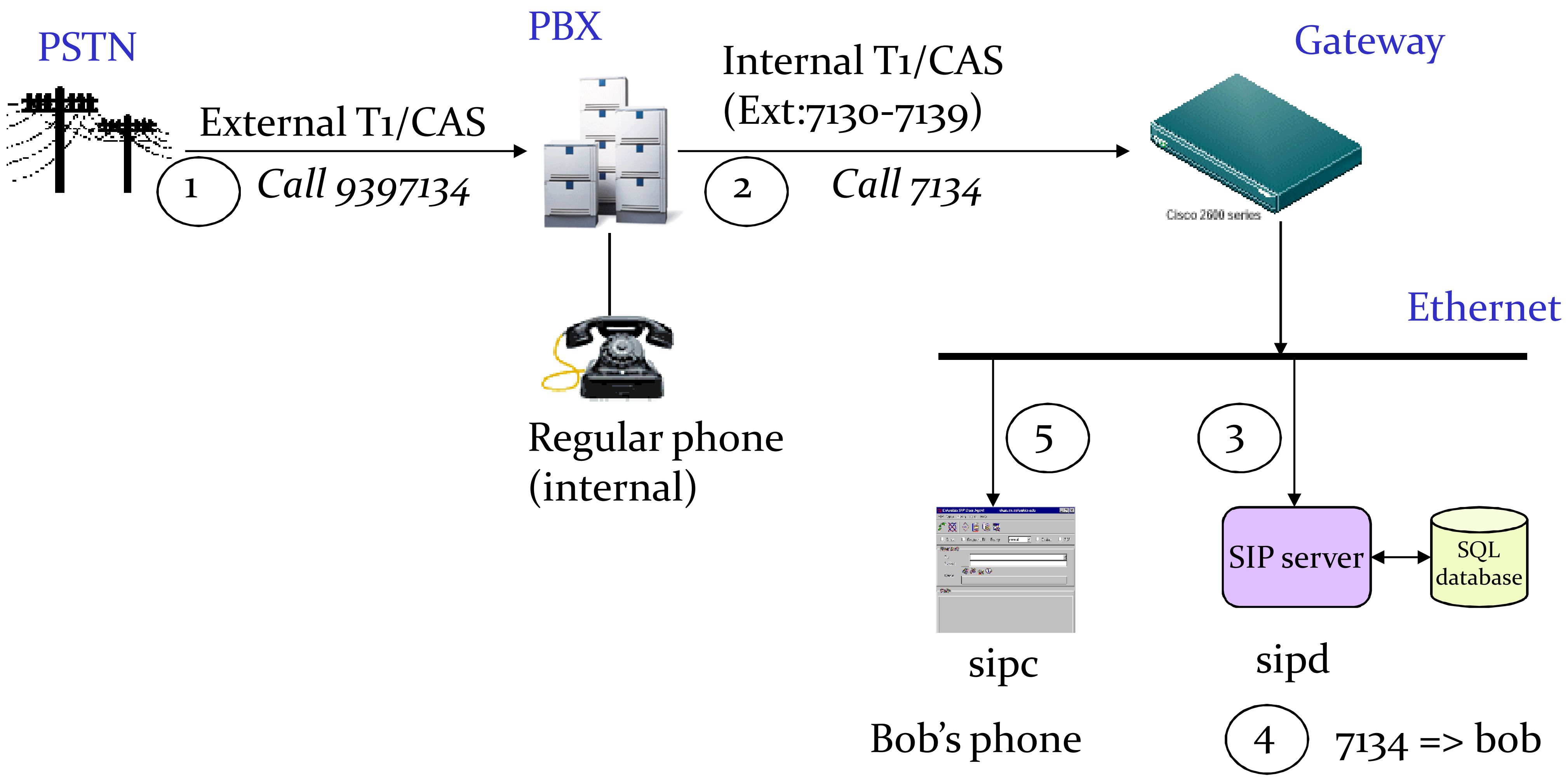


1. INVITE sip:president@us.gov
   From: sip:tony@parliament.uk
2. SIP/2.0 320 Moved temporarily
   Contact: sip:dcheney@wh.us.gov
3. ACK sip:president@us.gov
   From: sip:tony@parliament.uk
4. INVITE sip:dcheney@wh.us.gov
   From: tony@parliament.uk
5. SIP/2.0 200 OK
   To: tony@parliament.uk
6. ACK sip:dcheney@wh.us.gov
   From: sip:tony@parliament.uk

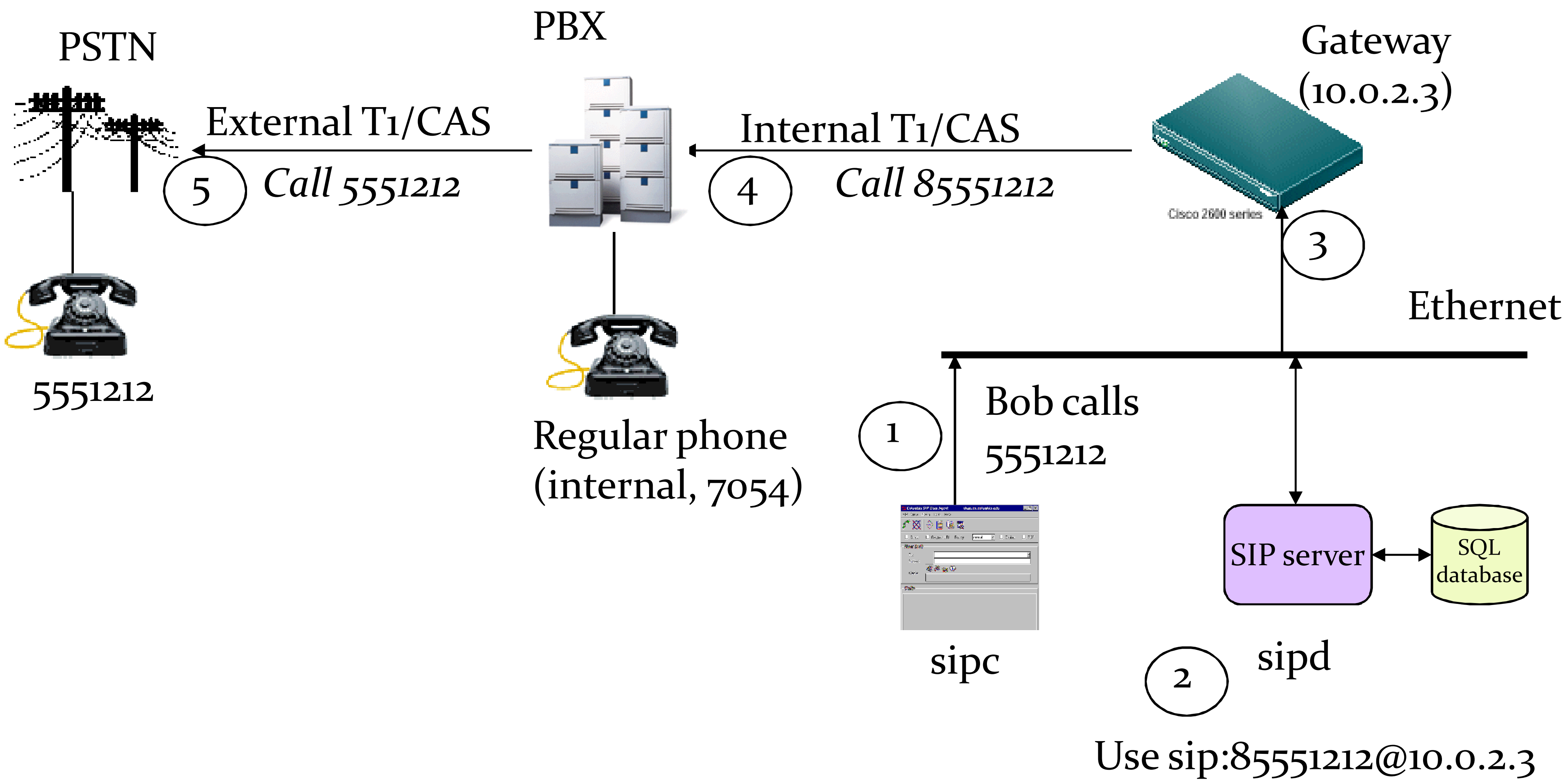
SIP Call Signaling



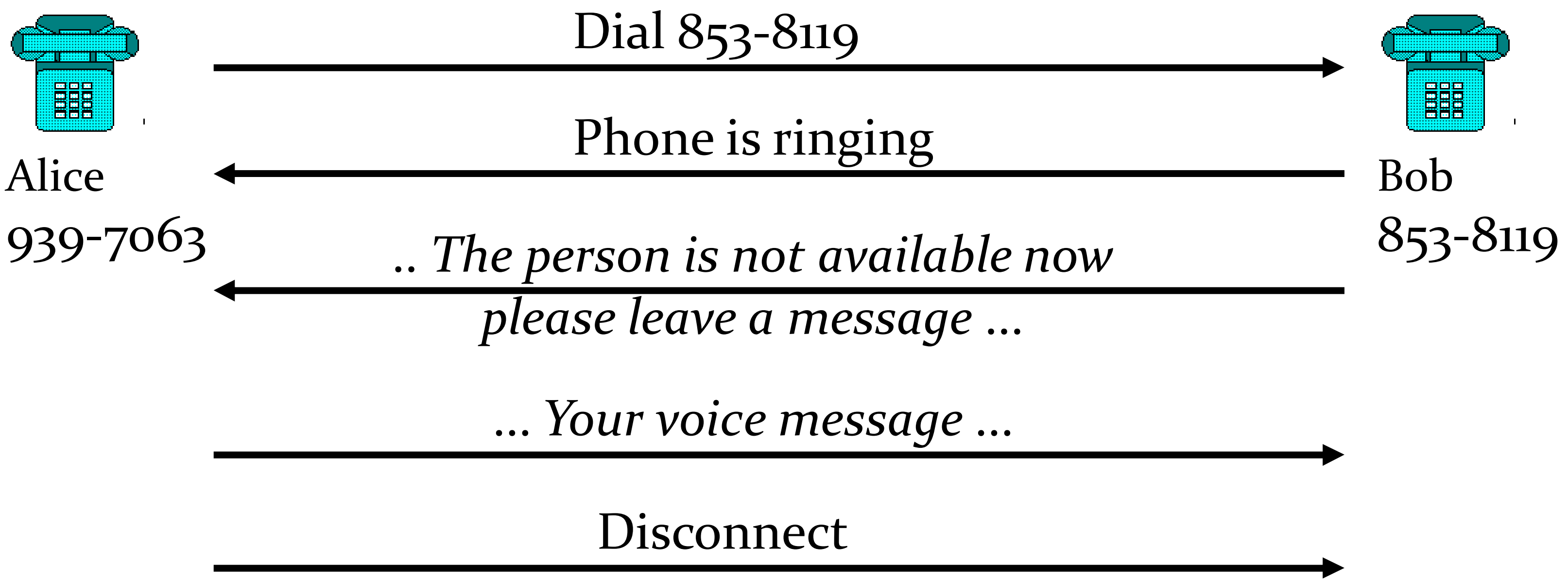
PSTN to IP Call



IP to PSTN Call

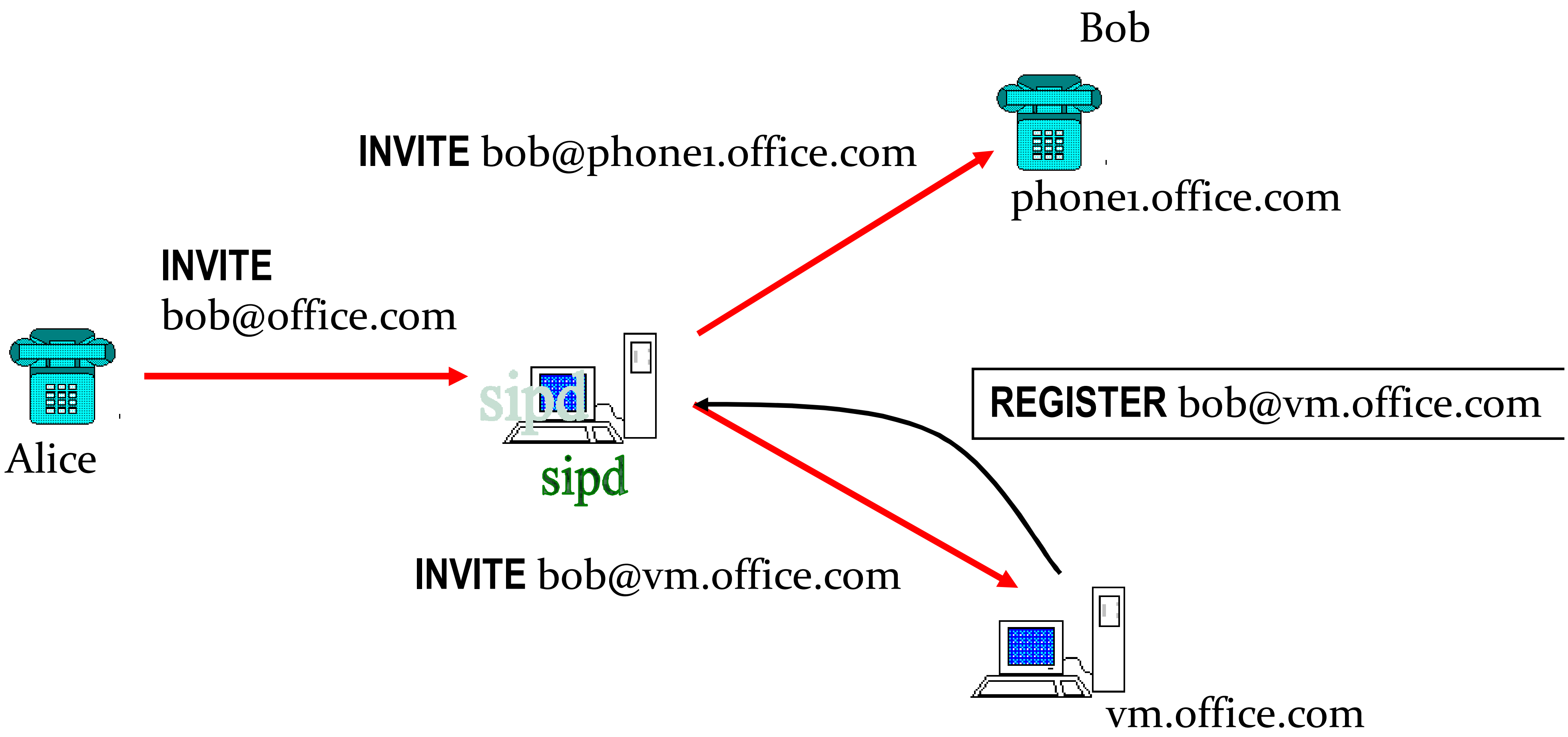


Traditional voice mail system



Bob can listen to his voice mails by dialing some number.

SIP-based Voicemail Architecture

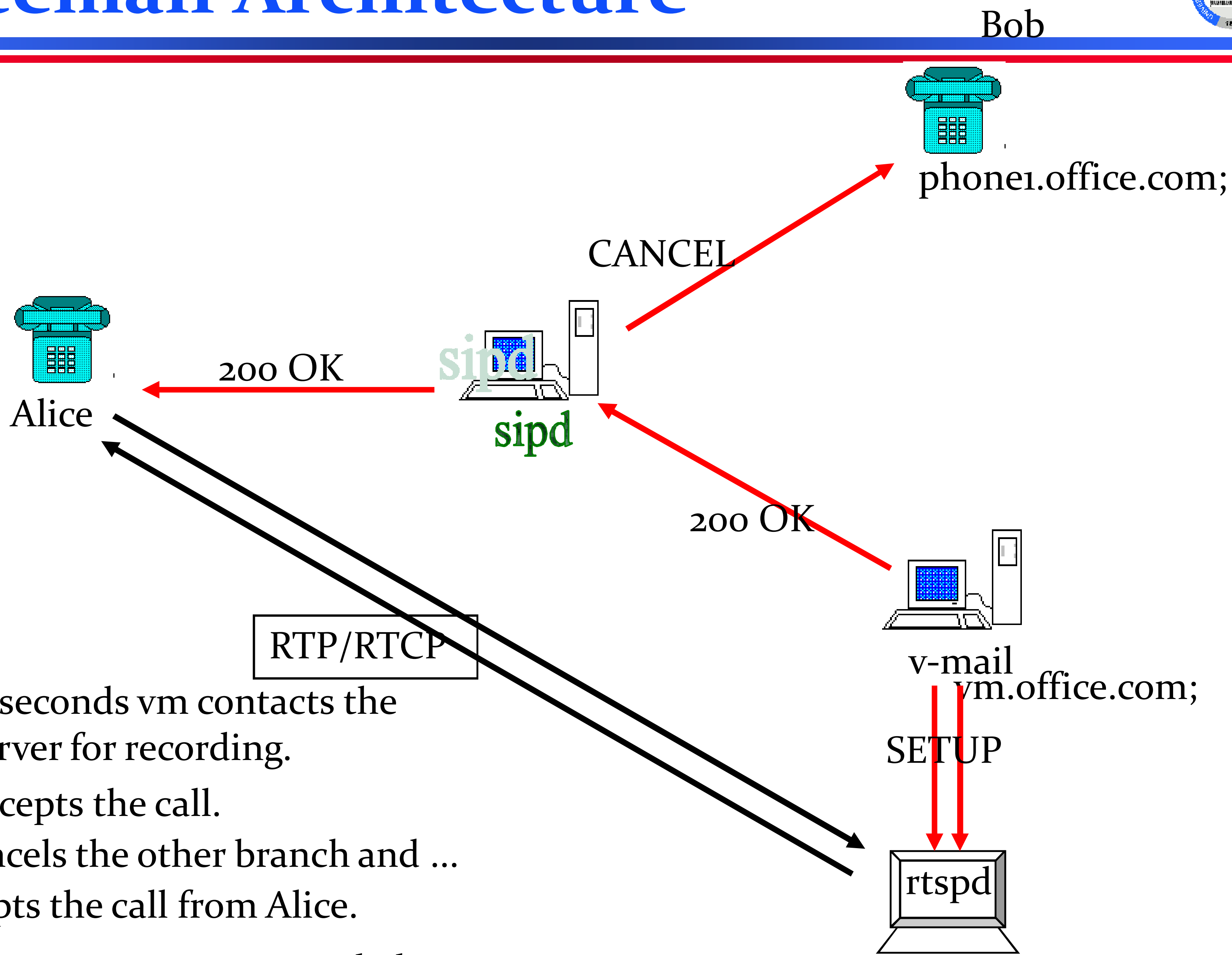


The voice mail server registers with the SIP proxy, sipd

Alice calls *bob@office.com* through SIP proxy.

SIP proxy forks the request to Bob's phone as well as to a voicemail server

Voicemail Architecture



After 10 seconds vm contacts the RTSP server for recording.

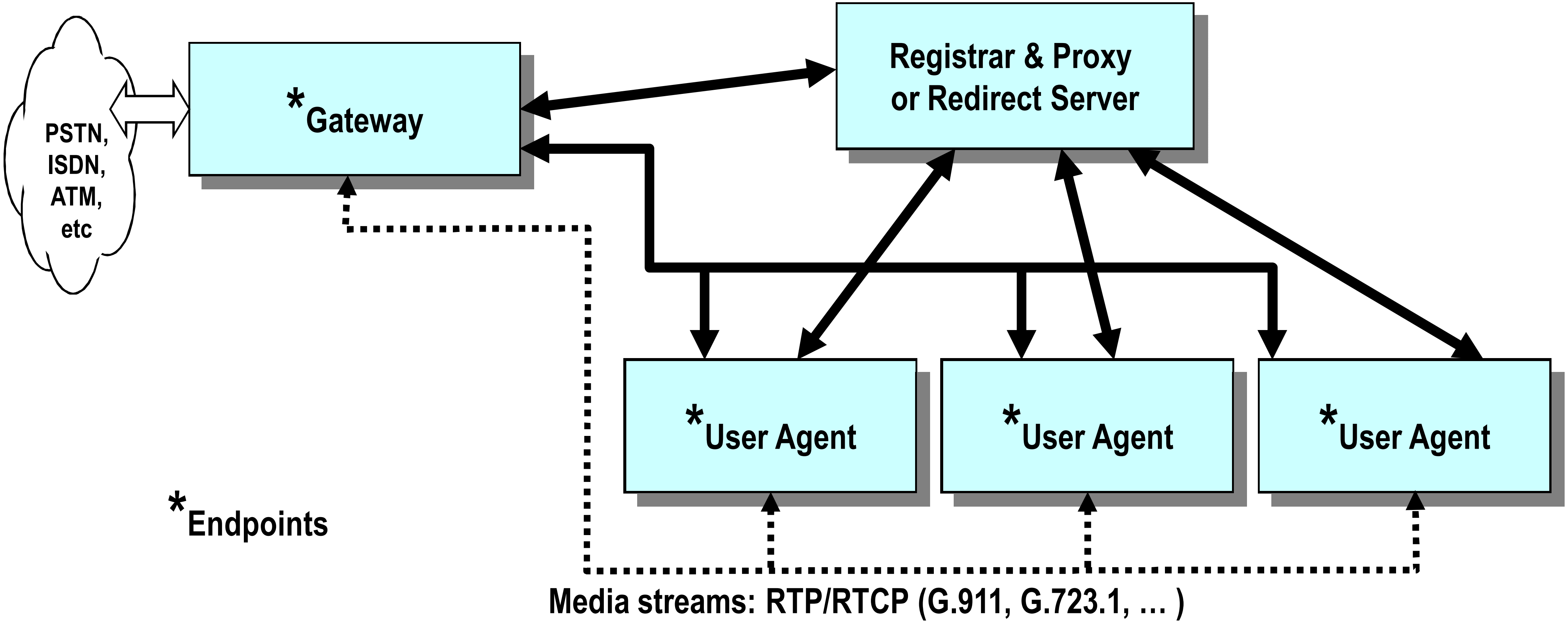
vm accepts the call.

Sipd cancels the other branch and ...

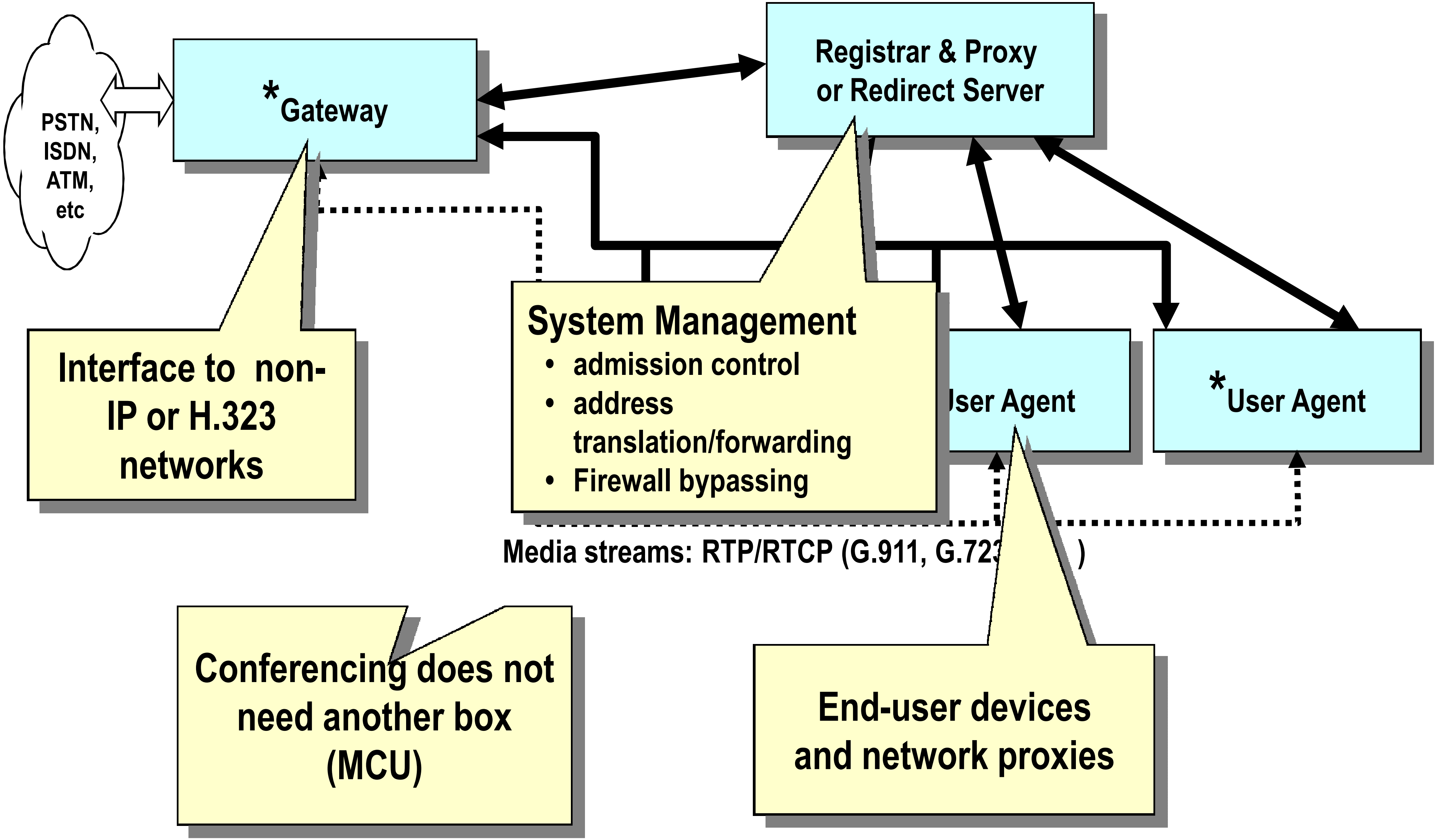
...accepts the call from Alice.

Now user message gets recorded

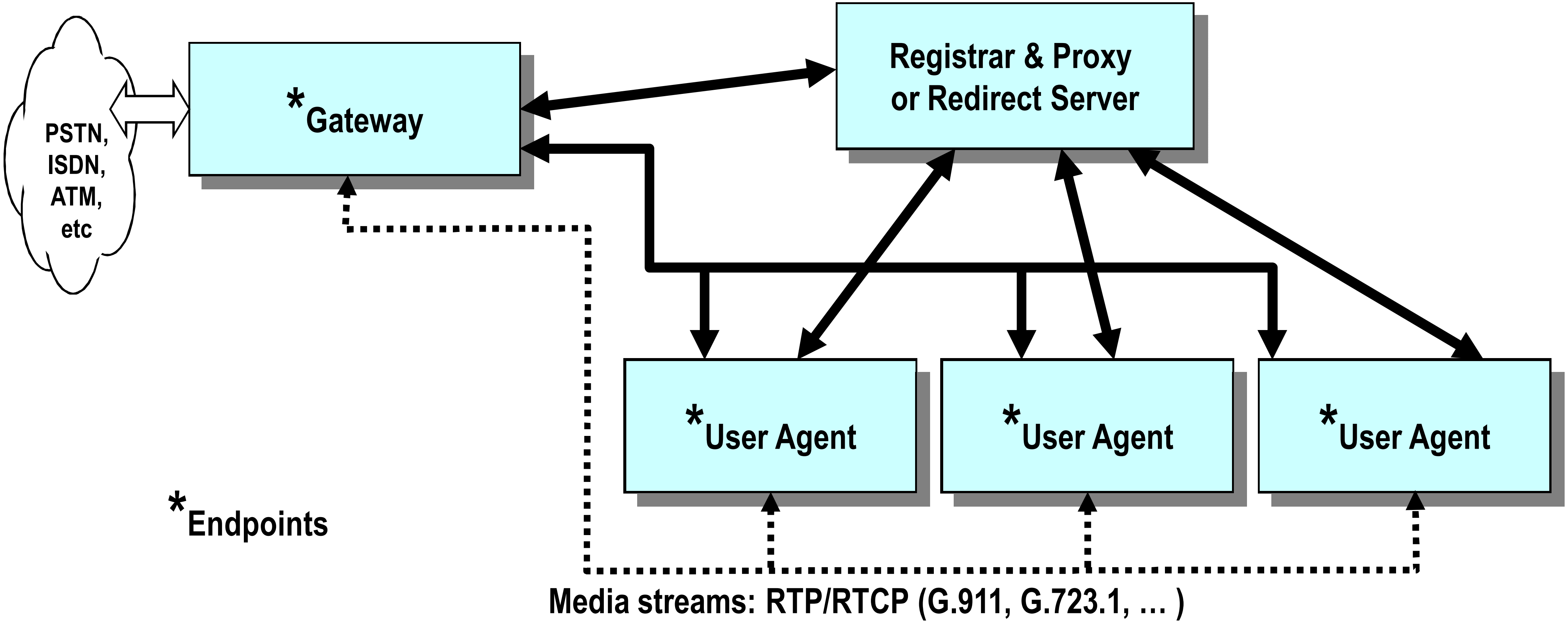
IETF SIP Architecture



IETF SIP Architecture Tour



IETF SIP Architecture Tour



Components of the SIP protocol suite:

- SIP = almost all signaling, optional services, etc.
- SDP = negotiation/capabilities
- DNS = address translation
- RSVP = QoS bandwidth guarantee

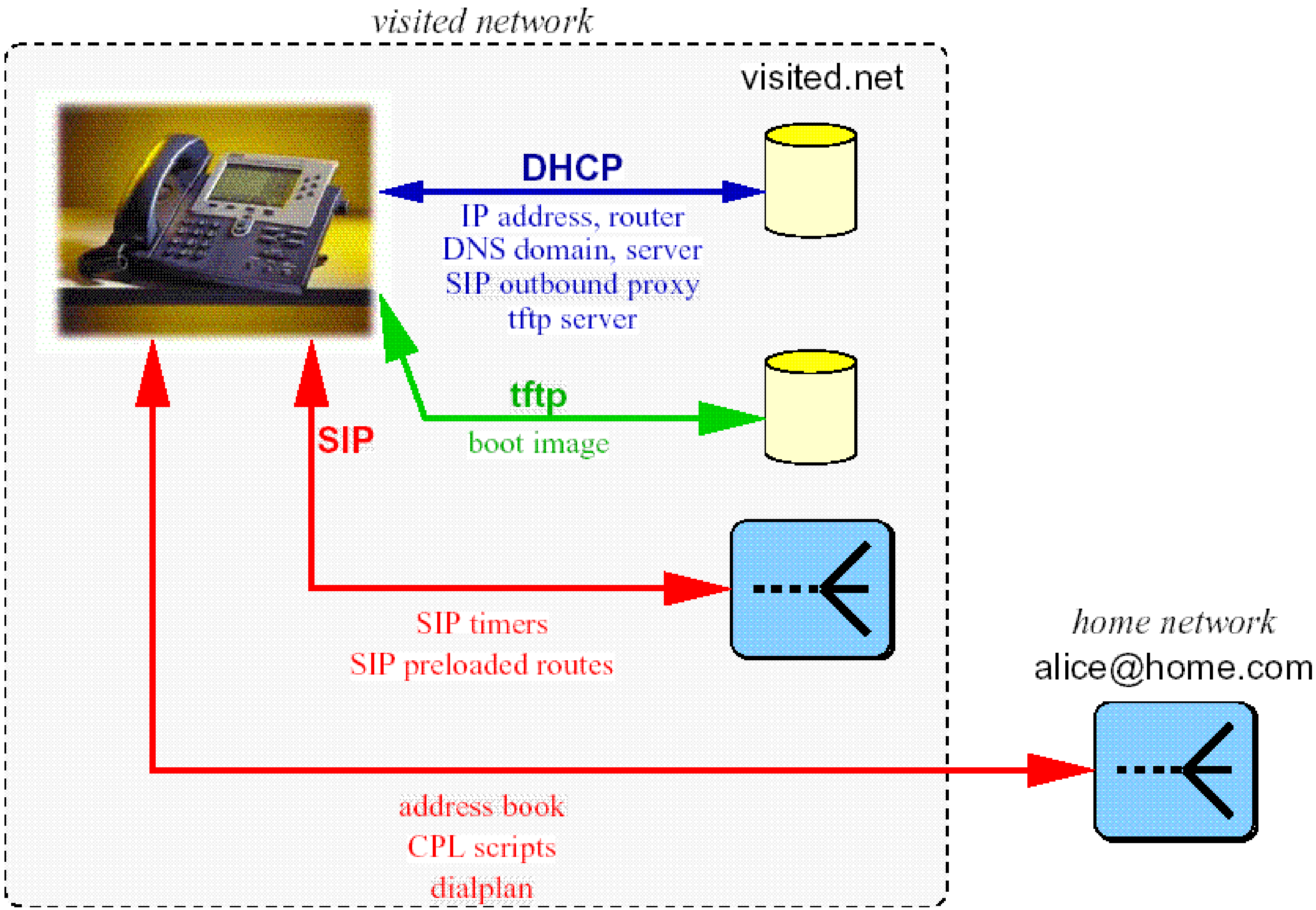
SIP for instant messaging: IM (RFC 3428)

- IM: transfer of (short) messages in near real-time, for conversational mode.
  - Current IM: proprietary, server-based and linked to buddy lists etc
- MESSAGE method: inherits SIP's request routing and security features
  - Message content as MIME body parts
  - Sent in the context of some SIP dialog
  - (note: slightly different from pager mode: asynchronous)
  - Sent over TCP (or congestion controlled transports): lots of messaging volumes…
- Allows IM applications to potentially interoperate and also provides SIP-based integration with other multimedia streams.

SIP compression (RFC 3486)

- Cannot use DNS SRV and NAPTR techniques: non-scalable (only useful for specifying transport protocol options)
- Use an application-level exchange to specify compression of signaling info
  - sip:alice@atlanta.com;comp=sigcomp
  - Via: SIP/2.0/UDP server1.foo.com:5060;branch=z9hG4bK87a7;comp=sigcomp
- SIGCOMP is the compression protocol

Device Configuration



SIP Scaling Issues

- SIP signaling primarily handled by SIP proxies, with associated registrars and location servers
- critical – common infrastructure for IM/presence, VoIP, conferences, mobile networks, . . .
- SIP proxies do not switch voice, but
  - route calls – mobility
  - implement policies
  - programmable logic
- far higher variability than classical switches: execute subscriber-defined code during call signaling:
  - sip-cgi scripts (similar to web cgi-bin scripts)
  - CPL scripts – XML-based call logic

SIP Scaling (contd)

Some metrics:

- BHCA – 750,000 to 2.5 million busy hour call attempts for large class-5 switches = 3.6 ms/call
- AT&T: 280 million calls a day = 0.3 ms/call
- Yahoo: 780 million page views/day
- AOL: 110 million emails/day
- AOL: 500 million IM/day
- web server: about 1,500 to 3,000 static requests/second

SIP Load Characteristics:

- not CPU-bound ➠ delay $\neq$ 1/throughput
- low byte volume ➠ easy to physically distribute for redundancy and load distribution
- servers can easily be shared among domains