Session Initiation Protocol (SIP)

Chapter 5

Introduction

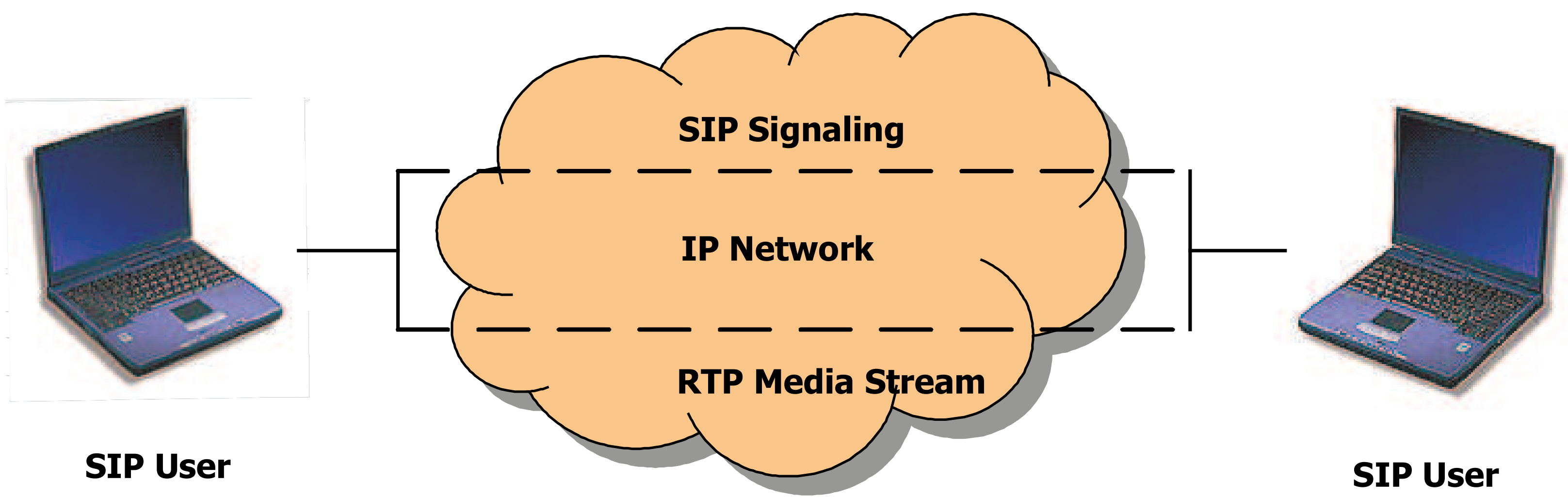
- A powerful alternative to H.323
- More flexible, simpler
- Easier to implement
  - Advanced features
- Better suited to the support of intelligent user devices
- A part of IETF multimedia data and control architecture
- SDP, RTSP (Real-Time Streaming Protocol), SAP (Session Announcement Protocol)

The Popularity of SIP

- Originally Developed in the MMUSIC (Multiparty Multimedia Session Control)
  - A separate SIP working group
  - RFC 2543
  - Many developers
  - The latest version: RFC 3261
- SIP + MGCP/MEGACO
  - The VoIP signaling in the future
- "bake-off"
  - Various vendors come together and test their products against each other
    - to ensure that they have implemented the specification correctly
    - to ensure compatibility with other implementations

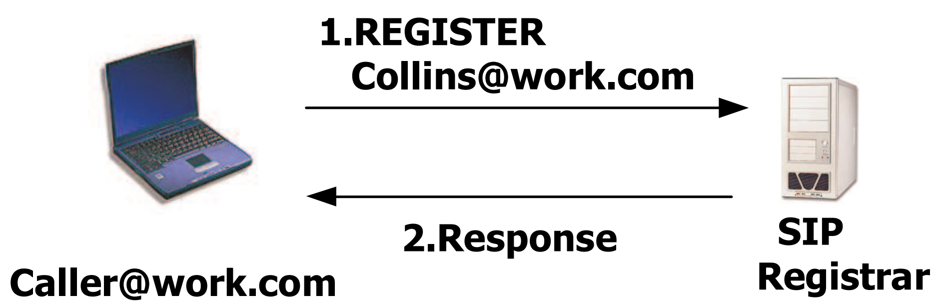
SIP Architecture

- A signaling protocol
  - The setup, modification, and tear-down of multimedia sessions
- SIP + SDP
  - Describe the session characteristics
- Separate signaling and media streams



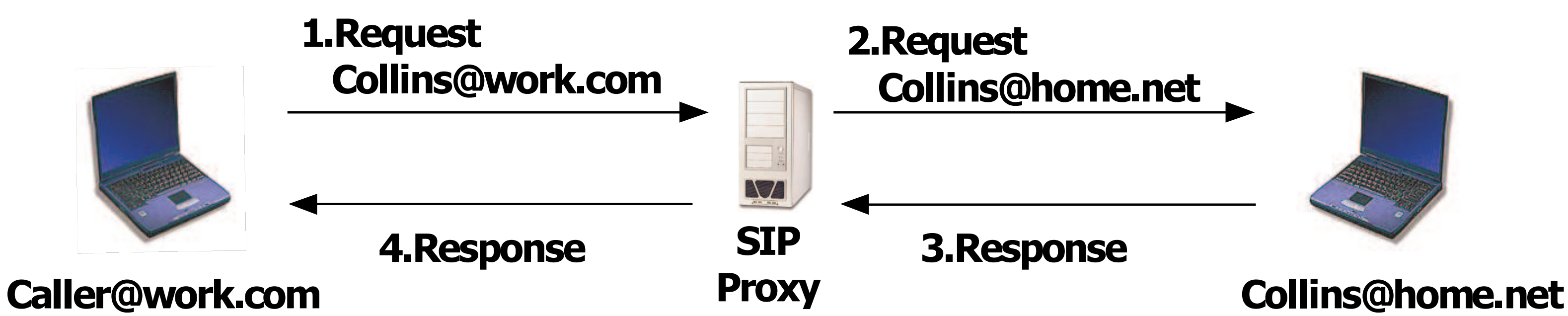
SIP Servers [1/3]

- Registrar
  - Accepts SIP REGISTER requests
    - Indicating that the user is at a particular address
    - Personal mobility
  - Typically combined with a proxy or redirect server



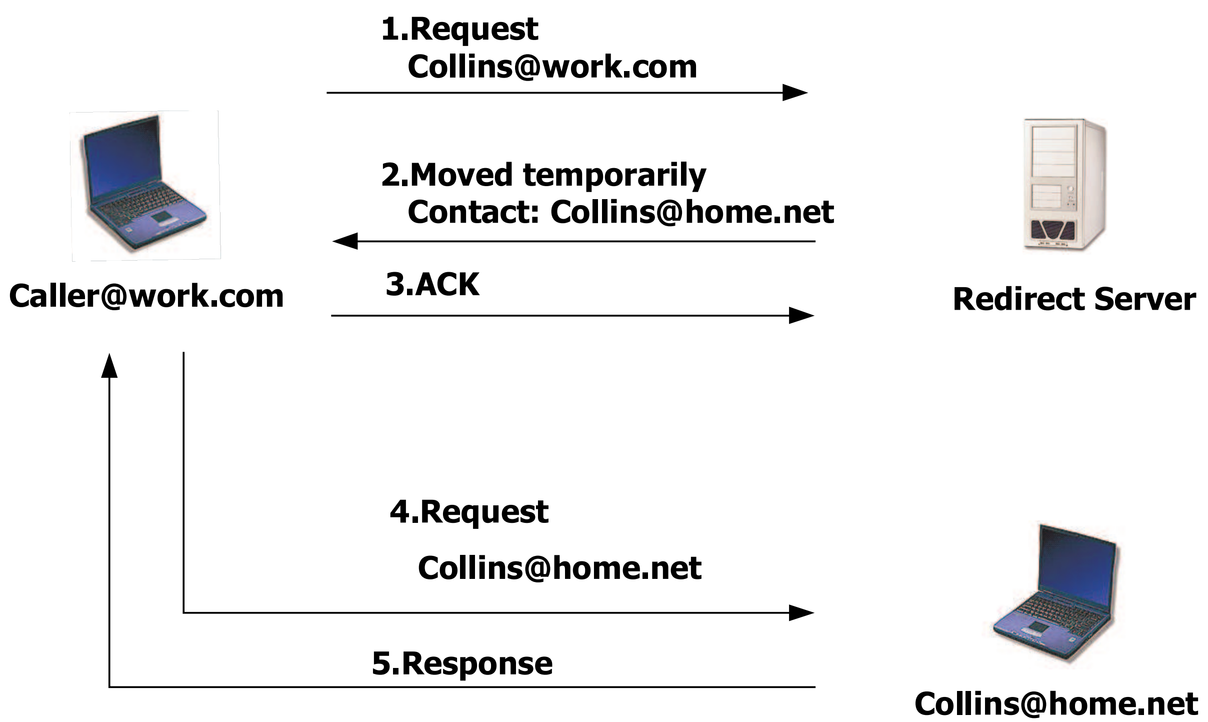
SIP Servers [2/3]

- Proxy servers
  - Handle requests or forward requests to other servers
  - Can be used for call forwarding, time-of-day routing, or follow-me services



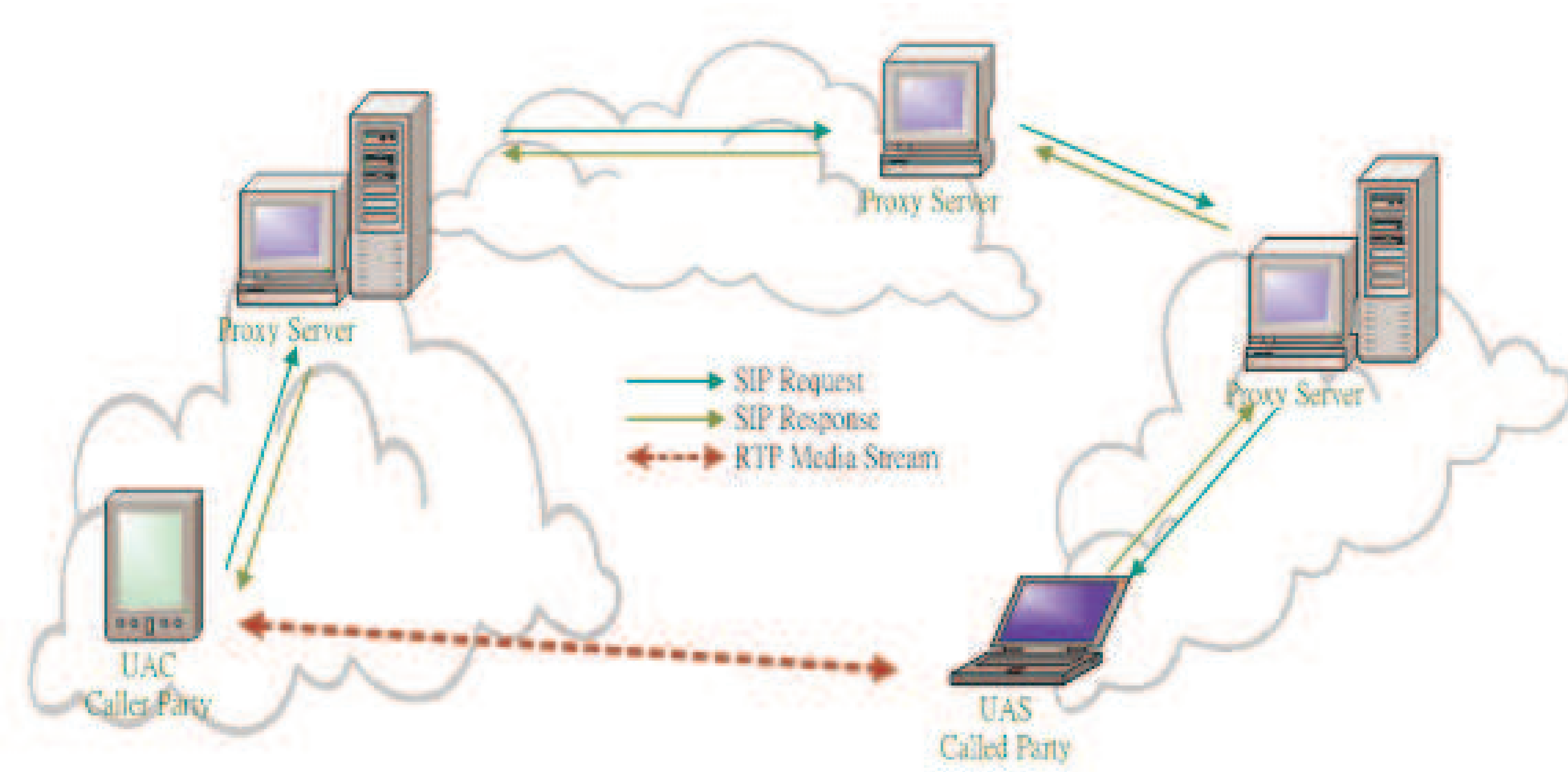
SIP Servers [3/3]

- Redirect servers
  - Map the destination address to zero or more new addresses



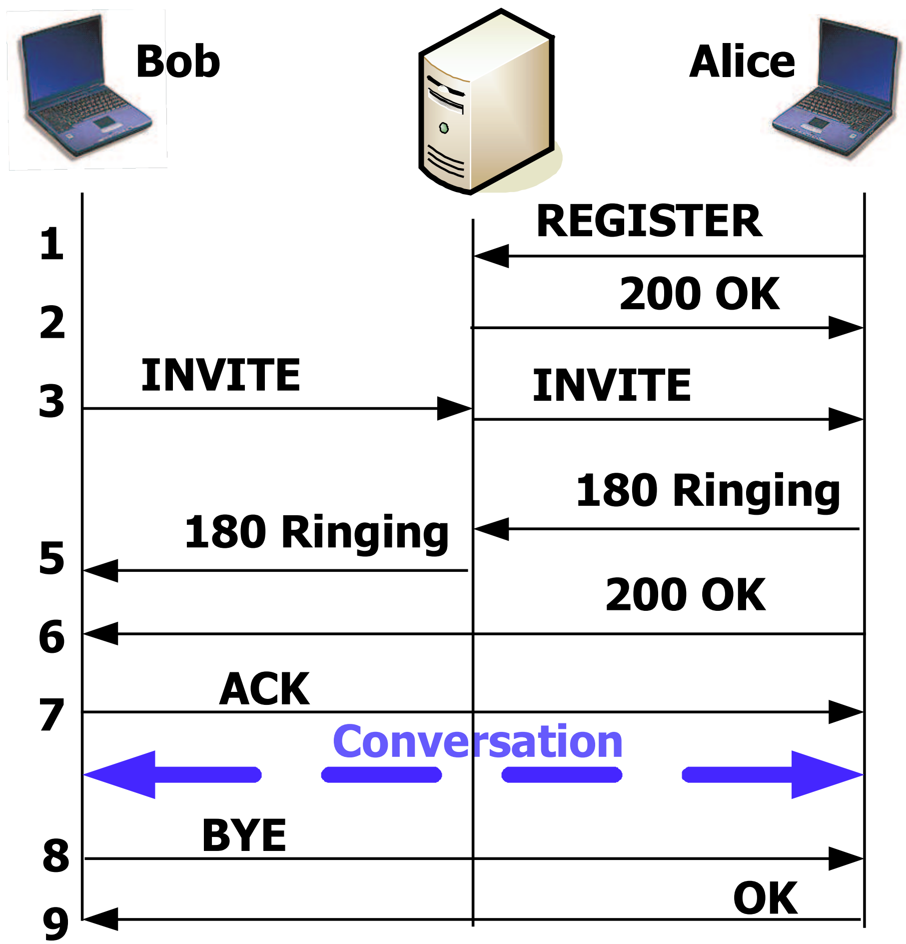
Redirect and proxy map to iterative and recursive lookup styles, respectively.

SIP Call Establishment [1/2]



SIP Call Establishment [2/2]

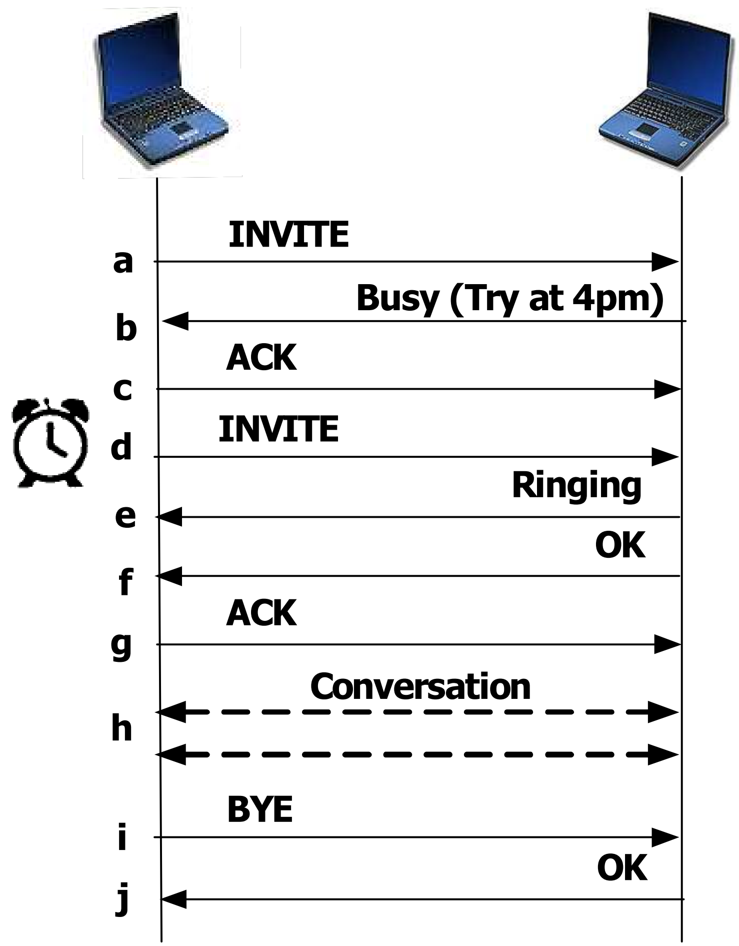
- It is simple, which contains a number of interim responses.



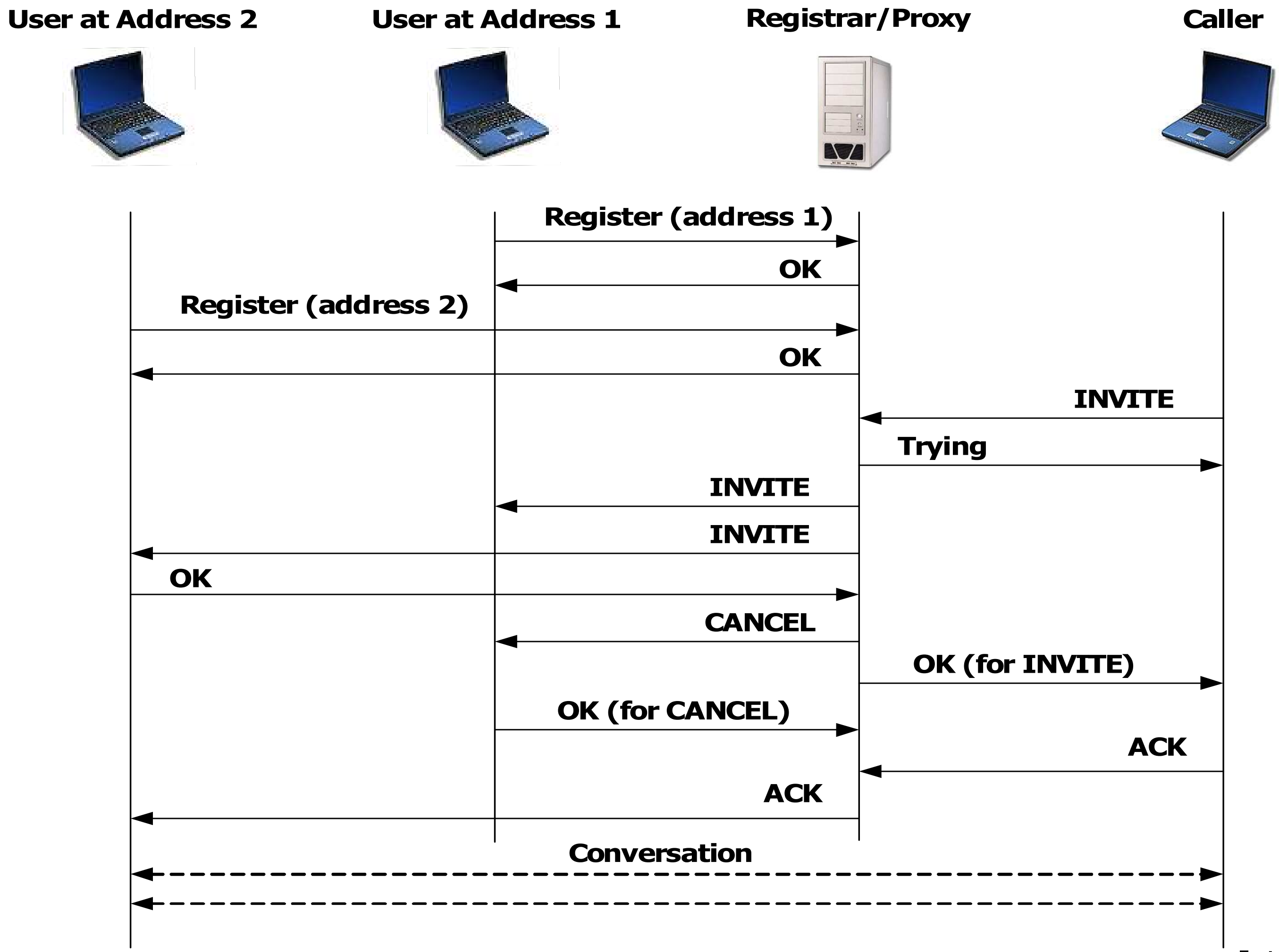
SIP Advantages

- Attempt to keep the signaling as simple as possible
- Offer a great deal of flexibility
  - Does not care what type of media is to be exchanged during a session or the type of transport to be used for the media
- Various pieces of information can be included within the messages
  - Including non-standard information
  - Enable the users to make intelligent decisions
    - The control of the intelligent features is placed in the hands of the customer, not the network operator.
  - E.g., SUBJECT header

Call Completion to A Busy User



"One number" service



Examples of SIP Messages

Collins@station1.work.com

Registrar

a

REGISTER sip:registrar.work.com SIP/2.0
Via: SIP/2.0/UDP station1.work.com; branch=z9hG4bK123
Max-Forwards: 70
From: sip:Collins@work.com; tag = 123456
To: sip:Collins@work.com
Call-ID: 123456@station1.work.com
CSeq: 1 REGISTER
Contact: sip:Collins@station1.work.com
Expires: 7200
Content-Length: 0

b

SIP/2.0 200 OK
Via: SIP/2.0/UDP station1.work.com; branch=z9hG4bK123
From: sip:Collins@work.com; tag = 123456
To: sip:Collins@work.com
Call-ID: 123456@station1.work.com
CSeq: 1 REGISTER
Contact: sip:Collins@station1.work.com
Expires: 3600
Content-Length: 0

Overview of SIP Messaging Syntax

- Text-based
  - Similar to HTTP
  - Disadvantage – more bandwidth consumption
- SIP messages
  - message = start-line
    *message-header CRLF
    [message-body]
  - start-line = request-line | status-line
- Request-line specifies the type of request
- The response line indicates the success or failure of a given request.

- Message headers
  - Additional information of the request or response
  - E.g.,
    - The originator and recipient
    - Retry-After header
    - Subject header
- Message body
  - Describe the type of session
  - The most common structure for the message body is SDP (Session Description Protocol).
  - Could include an ISDN User Part message
  - Examined only at the two ends

SIP Requests [1/2]

- Method SP Request-URI SP SIP-version CRLF
- Request-URI
  - The address of the destination
- Methods
  - INVITE, ACK, OPTIONS, BYE, CANCEL, REGISTER
  - INVITE
    - Initiate a session
    - Information of the calling and called parties
    - The type of media
    - ∼IAM (initial address message) of ISUP
    - ACK only when receiving the final response

SIP Requests [2/2]

- BYE
  - Terminate a session
  - Can be issued by either the calling or called party
- OPTIONS
  - Query a server as to its capabilities
    - A particular type of media
- CANCEL
  - Terminate a pending request
  - E.g., an INVITE did not receive a final response
- REGISTER
  - Log in and register the address with a SIP server
  - "all SIP servers" – multicast address (224.0.1.175)
  - Can register with multiple servers
  - Can have several registrations with one server

SIP Responses

- SIP Version SP Status Code SP Reason-Phrase CRLF
- Reason-Phrase
  - A textual description of the outcome
  - Could be presented to the user
- status code
  - A three-digit number
  - 1XX Informational
  - 2XX Success (only code 200 is defined)
  - 3XX Redirection
  - 4XX Request Failure
  - 5XX Server Failure
  - 6XX Global Failure
  - All responses, except for 1XX, are considered final
    - Should be ACKed

SIP Addressing

- SIP URLs (Uniform Resource Locators)
  - user@host:port
  - sip:collins@home.net
  - sip:3344556789@telco.net
  - sip:solomon@ncnu.edu.tw:5060

Message Headers

- Provide further information about the message
- E.g.,
  - To:header in an INVITE
    - The called party (callee)
  - From:header
    - The calling party (caller)
- Four main categories
  - General, Request, Response, and Entity headers

General Headers

- Used in both requests and responses
- Basic information
  - E.g., To:, From:, Call-ID: (uniquely identifies a specific invitation to a session), …
- Contact:
  - Provides a URL for use in future communication regarding a particular session
  - Examples 1: In a SIP INVITE, the Contact header might be different from the From header.
    - An third-party administrator initiates a multiparty session.
  - Example 2: Used in response, it is useful for directing further requests directly to the called user.
  - Example 3: It is used to indicate a more appropriate address if an INVITE issued to a given URI failed to reach the user.

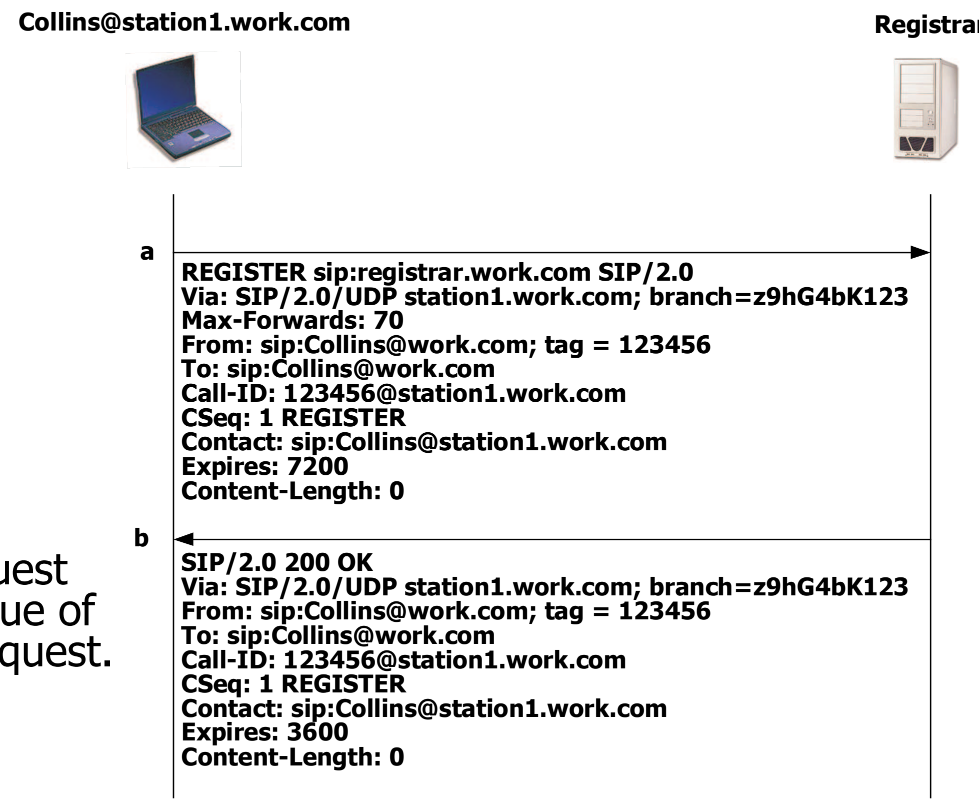
- Request Headers
  - Apply only to SIP requests
  - Addition information about the request or the client
  - E.g.,
    - Subject:
    - Priority:, urgency of the request (emergency, urgent, normal, or non-urgent)
- Response Headers
  - Further information about the response that cannot be included in the status line
  - E.g.,
    - Unsupported
    - Retry-After

Entity Headers

- Indicate the type and format of information included in the message body
- Content-Length: the length of the message body
- Content-Type: the media type of the message body
  - E.g., application/sdp
- Content-Encoding: for message compression
- Content Disposition: how a message part should be interpreted
  - session, alert, render …

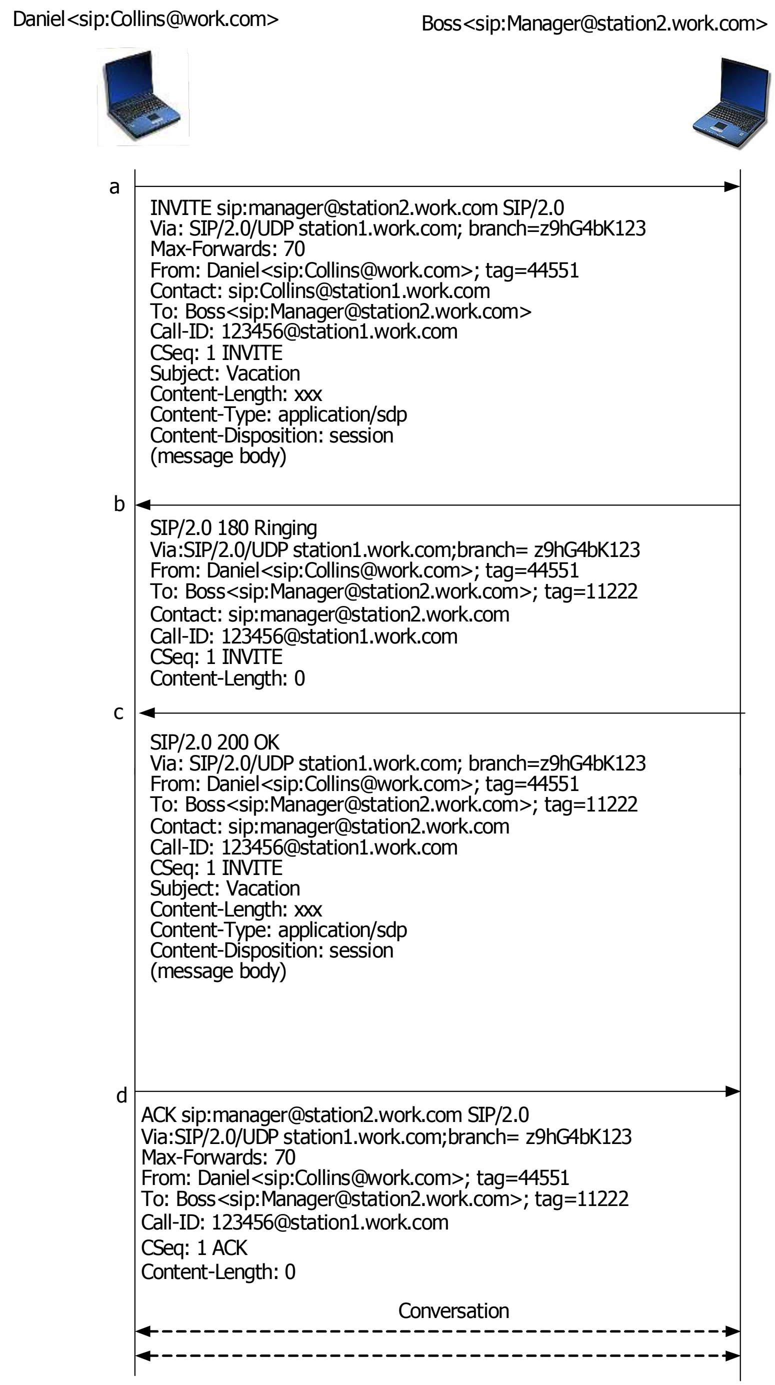
Examples of SIP Message Sequences

- Registration
  - Via:
  - From: and To:
  - Call-ID:
    - host-specific
  - Contact: (for future SIP message transmission)
  - Content-Length:
    - Zero, no msg body
  - CSeq:
    - A response to any request must use the same value of CSeq as used in the request.
  - Expires:
    - TTL (Time to live)
    - 0, un-register



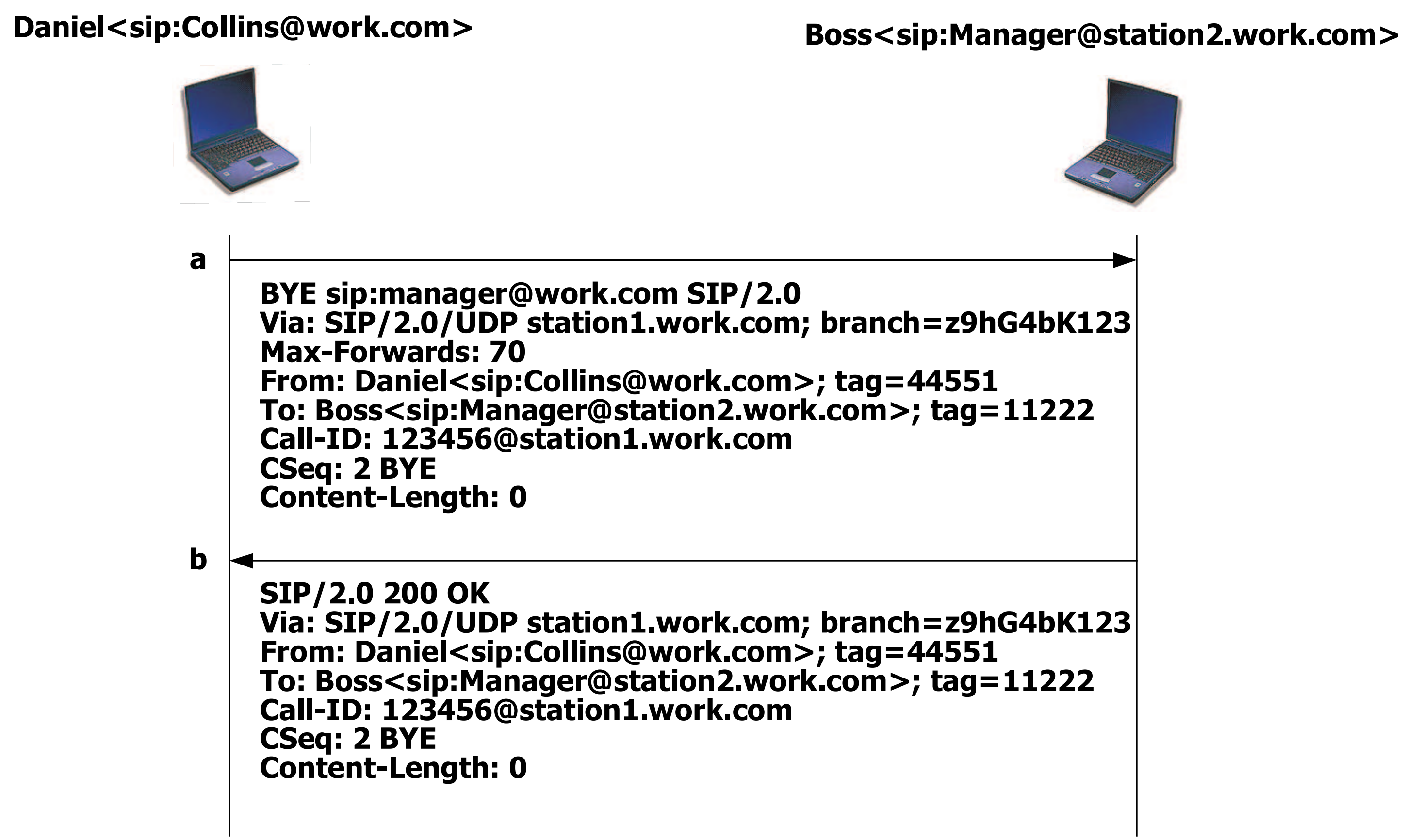
Invitation

- A two-party call
  - Subject:
    - optional
  - Content-Type:
    - application/sdp
  - A dialog ID
    - To identify a peer-to-peer relationship between two user agents
    - Tag in From
    - Tag in To
    - Call-ID



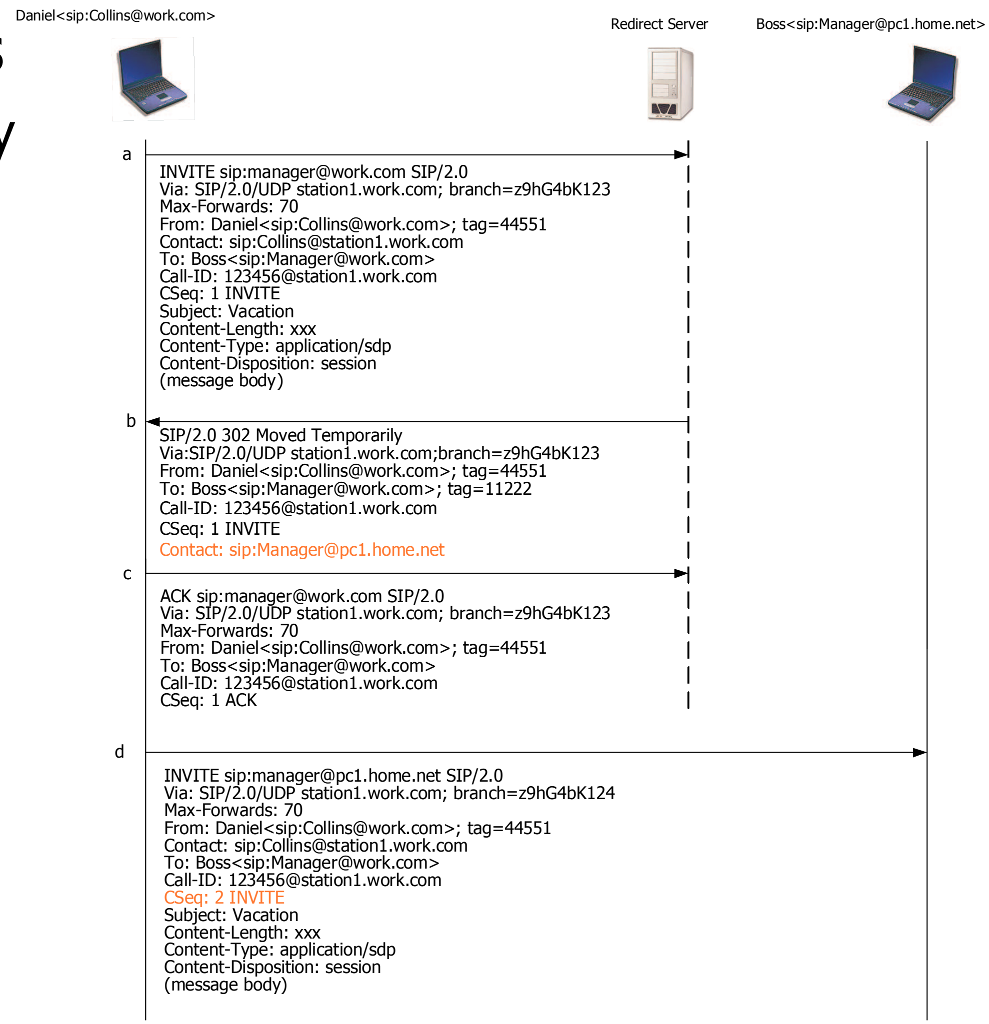
Termination of a Call

- Cseq:
  - Has changed



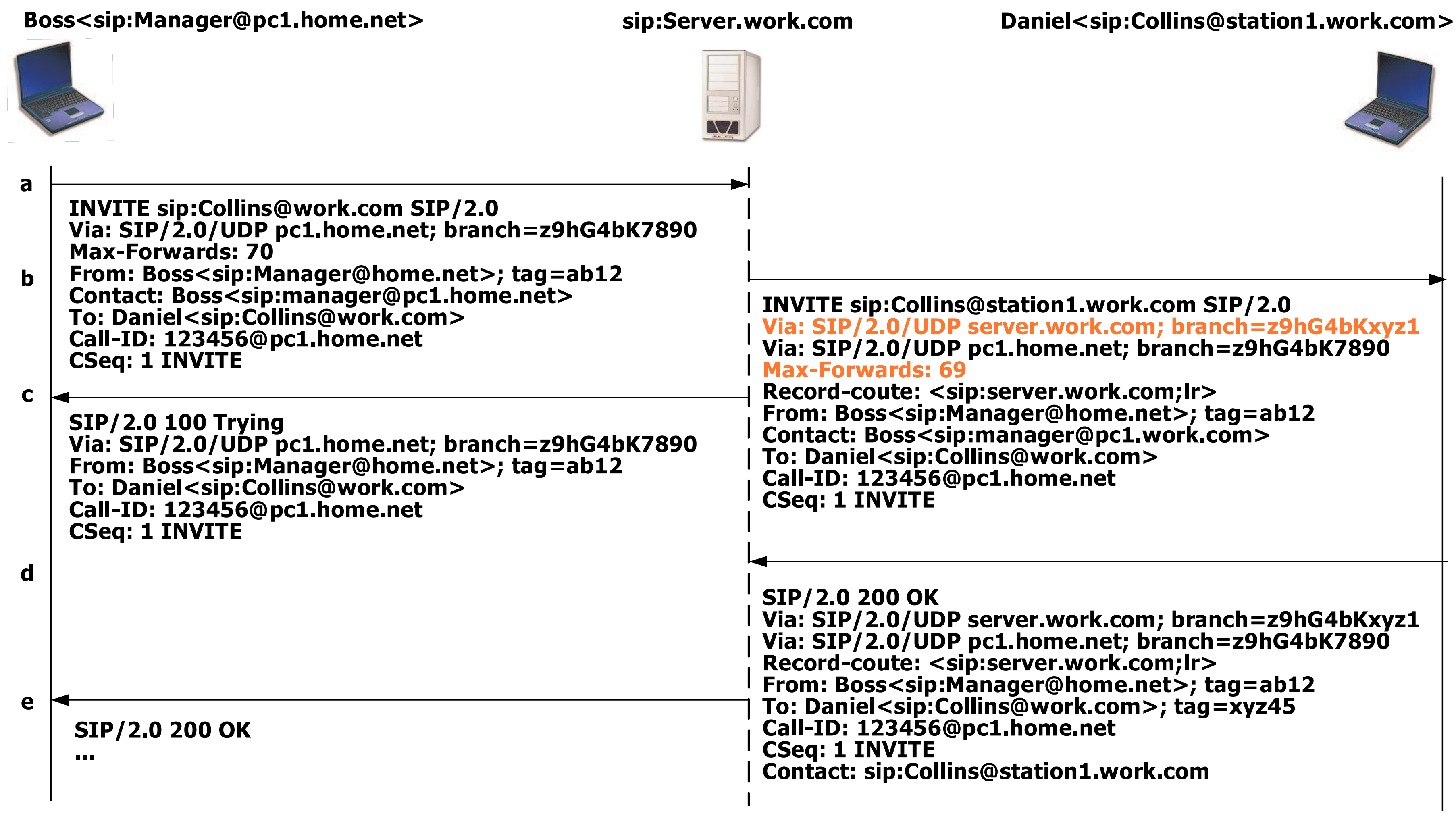
Redirect Servers

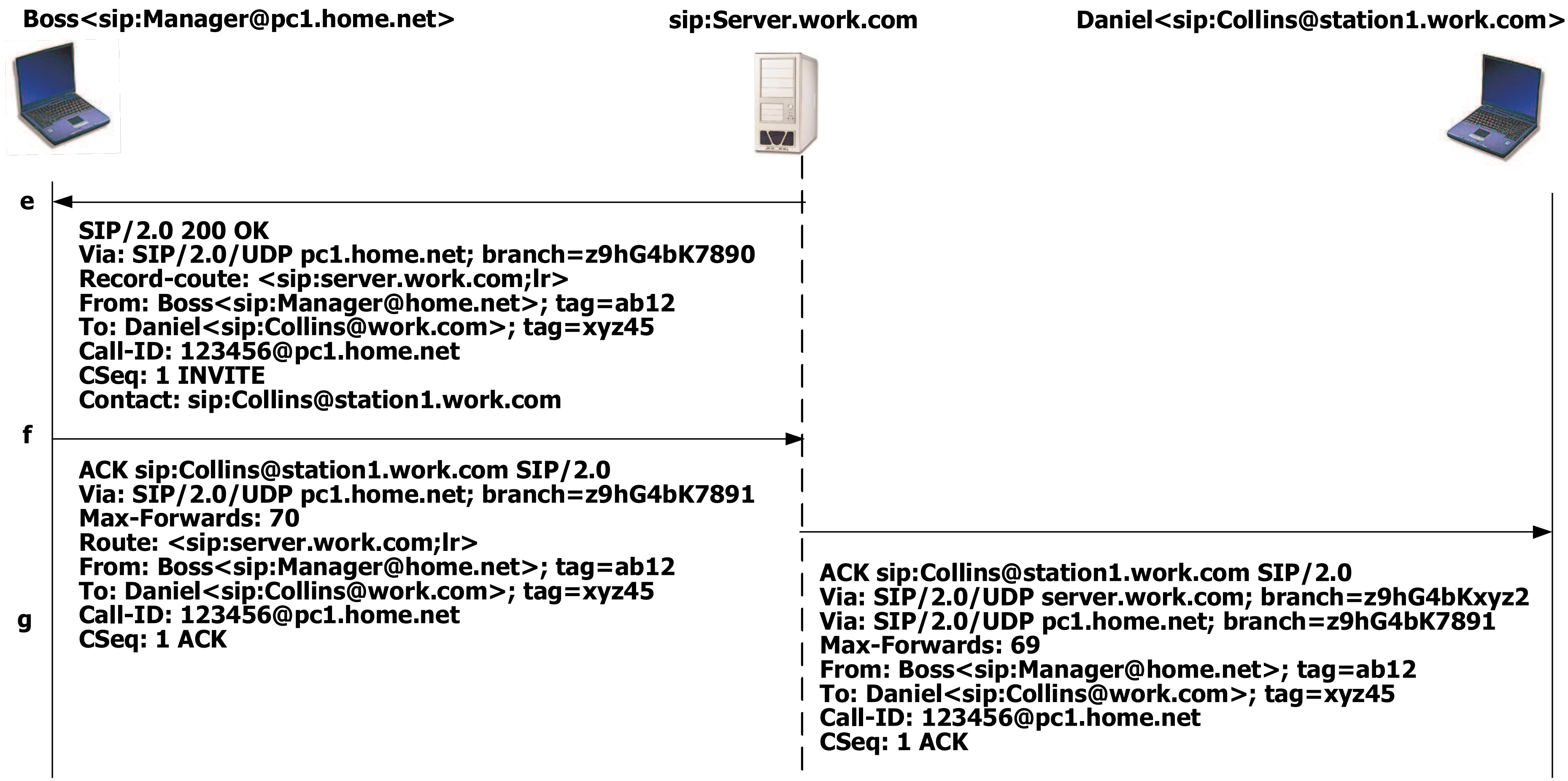
- An alternative address
  - 302, Moved temporarily
- Another INVITE
  - Same Call-ID
  - CSeq ++



Proxy Servers

- Sits between a user-agent client and the far-end user-agent server
- Numerous proxies can reside in a chain between the caller and callee.
- The last proxy may change the Request-URI.
- Via:
  - The path taken by a request
  - Loop detected, 482 (status code)
  - For a response
    - The 1st Via: header
    - Checked
    - Removed
  - Branch: used to distinguish between multiple responses to the same request
    - Forking Proxy: Issue a single request to multiple destinations


Boss<sip:Manager@pc1.home.net>
sip:Server.work.com
Daniel<sip:Collins@station1.work.com>
a
INVITE sip:Collins@work.com SIP/2.0
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
Max-Forwards: 70
From: Boss<sip:Manager@home.net>; tag=ab12
Contact: Boss<sip:manager@pc1.home.net>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
b
INVITE sip:Collins@station1.work.com SIP/2.0
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bKxyz1
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
Max-Forwards: 69
Record-coute: <sip:server.work.com;lr>
From: Boss<sip:Manager@home.net>; tag=ab12
Contact: Boss<sip:manager@pc1.work.com>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
c
SIP/2.0 100 Trying
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
From: Boss<sip:Manager@home.net>; tag=ab12
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
d
SIP/2.0 200 OK
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bKxyz1
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
Record-coute: <sip:server.work.com;lr>
From: Boss<sip:Manager@home.net>; tag=ab12
To: Daniel<sip:Collins@work.com>; tag=xyz45
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
Contact: sip:Collins@station1.work.com
e
SIP/2.0 200 OK
...

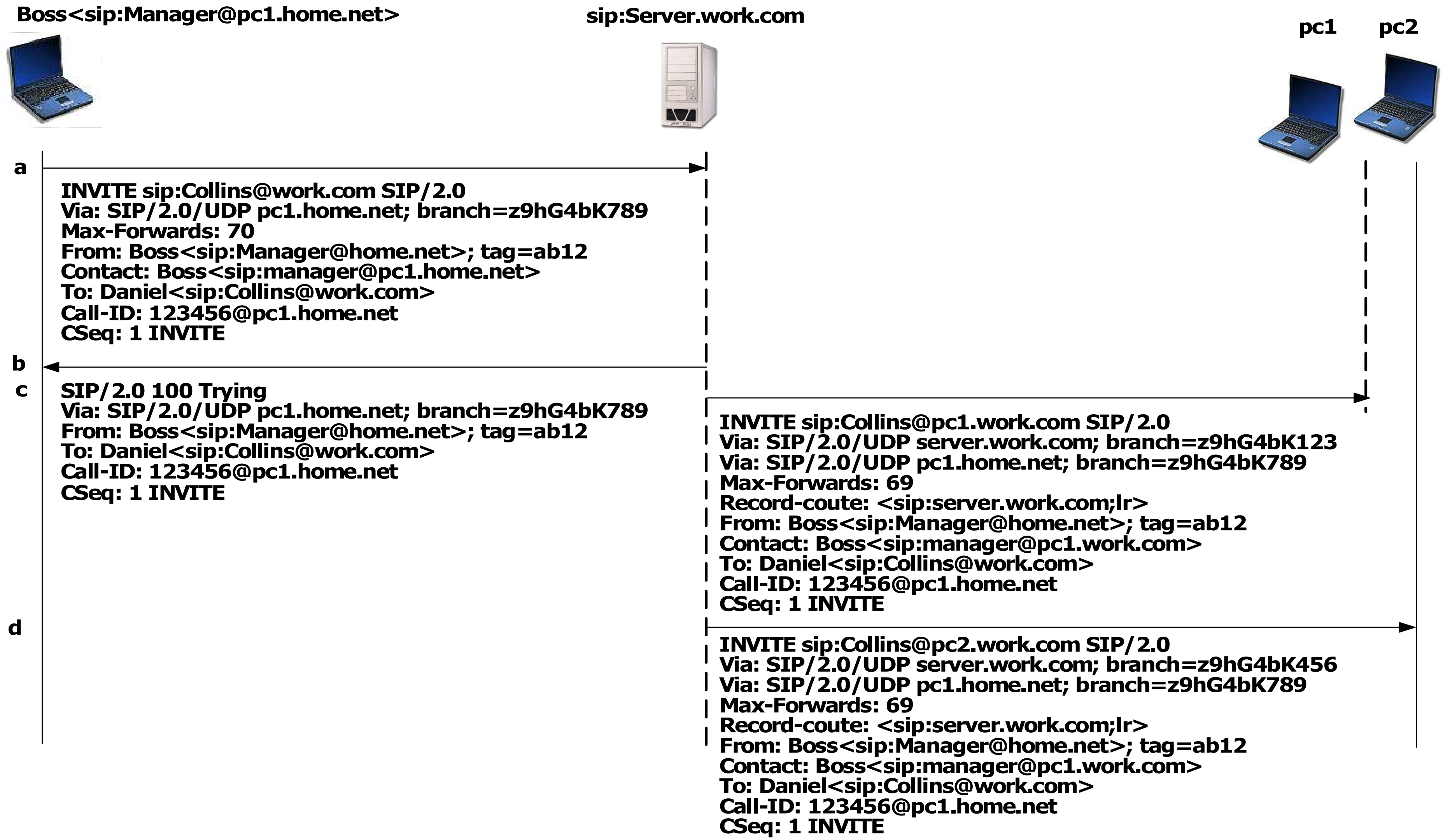

Boss<sip:Manager@pc1.home.net>
sip:Server.work.com
Daniel<sip:Collins@station1.work.com>
e
SIP/2.0 200 OK
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
Record-coute: <sip:server.work.com;lr>
From: Boss<sip:Manager@home.net>; tag=ab12
To: Daniel<sip:Collins@work.com>; tag=xyz45
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
Contact: sip:Collins@station1.work.com
f
ACK sip:Collins@station1.work.com SIP/2.0
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7891
Max-Forwards: 70
Route: <sip:server.work.com;lr>
From: Boss<sip:Manager@home.net>; tag=ab12
To: Daniel<sip:Collins@work.com>; tag=xyz45
Call-ID: 123456@pc1.home.net
CSeq: 1 ACK
g
ACK sip:Collins@station1.work.com SIP/2.0
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bKxyz2
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7891
Max-Forwards: 69
From: Boss<sip:Manager@home.net>; tag=ab12
To: Daniel<sip:Collins@work.com>; tag=xyz45
Call-ID: 123456@pc1.home.net
CSeq: 1 ACK

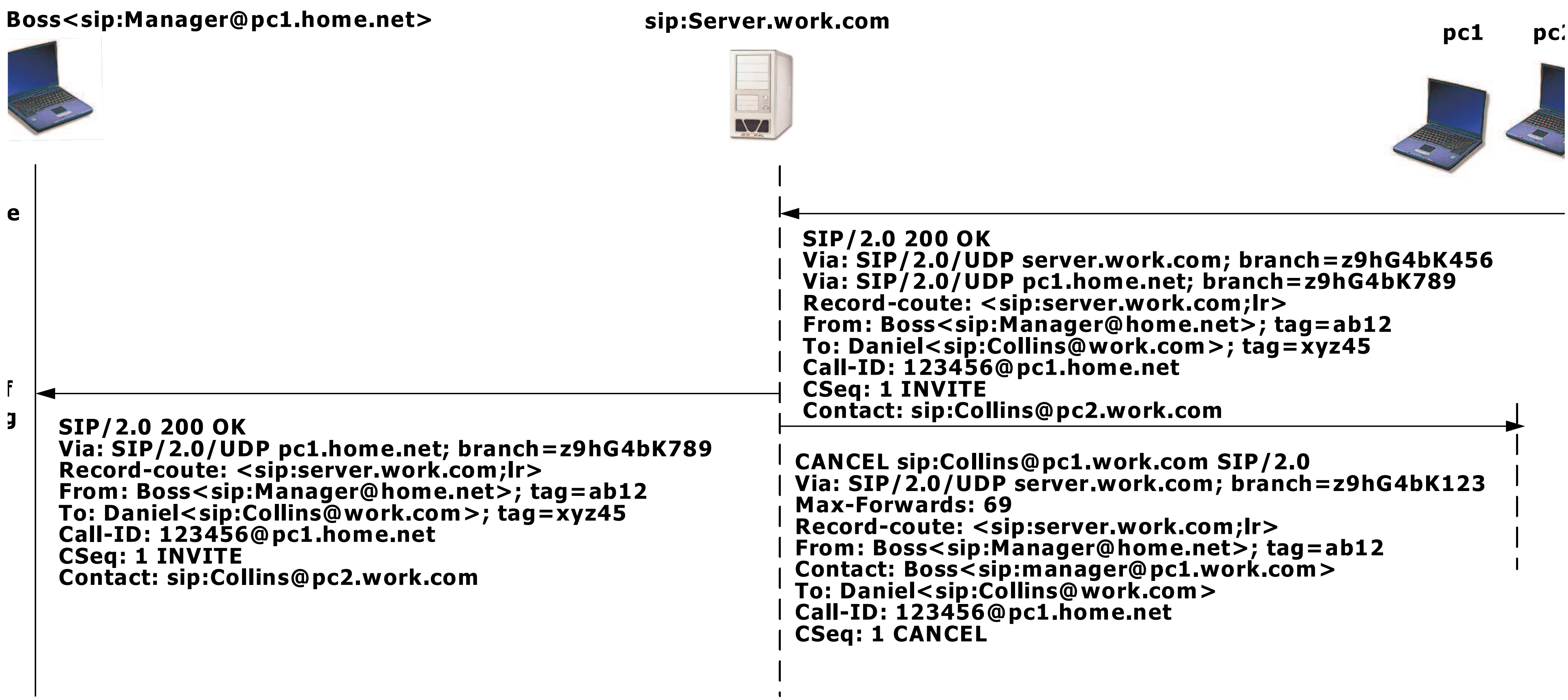
Proxy state

- Can be either stateless or stateful
- Record-Route:
  - With Contact header field, subsequent requests and responses may not pass through the same proxy.
  - A Proxy might require that it remains in the signaling path
    - In particular, for a stateful proxy
    - Each stateful proxy inserts its address into the Record-Route: header field
  - The 200 response from the final destination to the caller will include the Record-Route: header field
  - The caller will use the Record-Route: header field in sending the subsequent requests.
    - Take the information from the Record-Route: header field and place them in a Route: header field in reverse order
    - At each proxy along the path from caller to callee, the first entry in the Route header field is removed.

Forking Proxy

- “fork” requests
- A user is registered at several locations
  - ;branch=xxx
  - The purpose of the branch parameter is to distinguish between the various outgoing requests, and more importantly, the responses to those requests.
- In order to handle such forking, a proxy must be stateful.


Boss<sip:Manager@pc1.home.net>
sip:Server.work.com
pc1
pc2
a
INVITE sip:Collins@work.com SIP/2.0
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK789
Max-Forwards: 70
From: Boss<sip:Manager@home.net>; tag=ab12
Contact: Boss<sip:manager@pc1.home.net>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
b
SIP/2.0 100 Trying
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK789
From: Boss<sip:Manager@home.net>; tag=ab12
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
c
INVITE sip:Collins@pc1.work.com SIP/2.0
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bK123
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK789
Max-Forwards: 69
Record-coute: <sip:server.work.com;lr>
From: Boss<sip:Manager@home.net>; tag=ab12
Contact: Boss<sip:manager@pc1.work.com>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
d
INVITE sip:Collins@pc2.work.com SIP/2.0
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bK456
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK789
Max-Forwards: 69
Record-coute: <sip:server.work.com;lr>
From: Boss<sip:Manager@home.net>; tag=ab12
Contact: Boss<sip:manager@pc1.work.com>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE

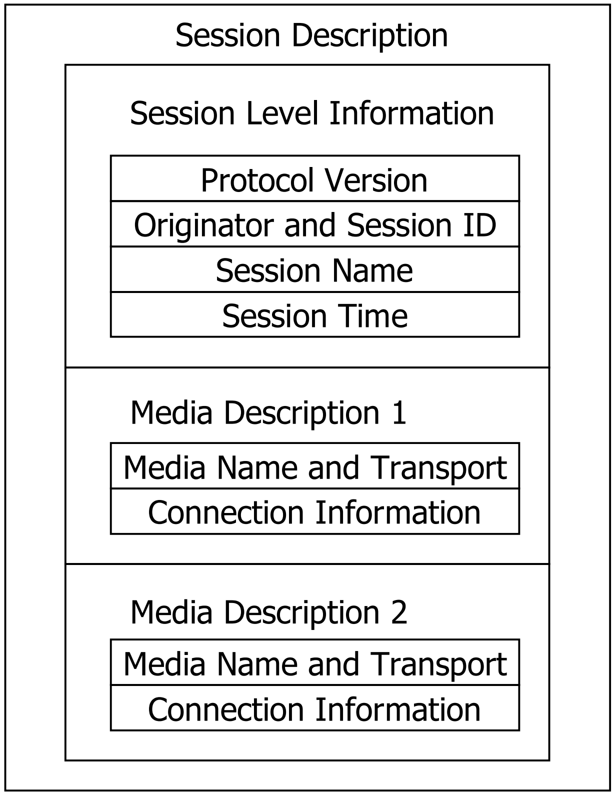

Boss<sip:Manager@pc1.home.net>
sip:Server.work.com
pc1
SIP/2.0 200 OK
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bK456
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK789
Record-coute: <sip:server.work.com;lr>
From: Boss<sip:Manager@home.net>; tag=ab12
To: Daniel<sip:Collins@work.com>; tag=xyz45
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
Contact: sip:Collins@pc2.work.com
SIP/2.0 200 OK
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK789
Record-coute: <sip:server.work.com;lr>
From: Boss<sip:Manager@home.net>; tag=ab12
To: Daniel<sip:Collins@work.com>; tag=xyz45
Call-ID: 123456@pc1.home.net
CSeq: 1 INVITE
Contact: sip:Collins@pc2.work.com
CANCEL sip:Collins@pc1.work.com SIP/2.0
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bK123
Max-Forwards: 69
Record-coute: <sip:server.work.com;lr>
From: Boss<sip:Manager@home.net>; tag=ab12
Contact: Boss<sip:manager@pc1.work.com>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 CANCEL

The Session Description Protocol

- The Most Common Message Body
  - Be session information describing the media to be exchanged between the parties
  - SDP, RFC 2327 (initial publication)
- SIP uses SDP in an answer/offer mode.
  - An agreement between the two parties as to the types of media they are willing to share
  - RFC 3264 (An Offer/Answer Model with SDP)
    - To describe how SDP and SIP should be used together

The Structure of SDP

- SDP simply provides a format for describing session information to potential session participants.
- Text-based Protocol
- The Structure of SDP
  - Session Level Info
    - Name of the session
    - Originator of the session
    - Time that the session is to be active
  - Media Level Info
    - Media type
    - Port number
    - Transport protocol
    - Media format



SDP Syntax

- A number of lines of text
- In each line
  - field=value
  - field is exactly one character (case-significant)
- Session-level fields
- Media-level fields
  - Begin with media description field (m=)

Mandatory Fields

- v=(protocol version)
- o=(session origin or creator)
- s=(session name), a text string
  - For multicast conference
- t=(time of the session), the start time and stop time
  - For pre-arranged multicast conference
- m=(media)
  - Media type
  - The transport port
  - The transport protocol
  - The media format, an RTP payload format

Optional Fields [1/3]

- Some optional fields can be applied at both session and media levels.
  - The value applied at the media level overrides that at the session level
- i=(session information)
  - A text description
  - At both session and media levels
  - It would be somewhat superfluous, since SIP already supports the Subject header.
- u=(URI of description)
  - Where further session information can be obtained
  - Only at session level

Optional Fields [2/3]

- e=(e-mail address)
  - Who is responsible for the session
  - Only at the session level
- p=(phone number)
  - Only at the session level
- c=(connection information)
  - Network type, address type and connection address
  - At session or media level
- b=(bandwidth information)
  - In kilobits per second
  - At session or media level

Optional Fields [3/3]

- r=(repeat times)
  - For regularly scheduled session a session is to be repeated
  - How often and how many times
- z=(timezone adjustments)
  - For regularly scheduled session
  - Standard time and daylight savings time
- k=(encryption key)
  - An encryption key or a mechanism to obtain it for the purposes of encrypting and decrypting the media
  - At session or media level
- a=(attributes)
  - Describe additional attributes

Ordering of Fields

- Session Level
  - Protocol version (v)
  - Origin (o)
  - Session name (s)
  - Session information (i)
  - URI (u)
  - E-mail address (e)
  - Phone number (p)
  - Connection info (c)
  - Bandwidth info (b)
  - Time description (t)
  - Repeat info (r)
  - Time zone adjustments (z)
  - Encryption key (k)
  - Attributes (a)
- Media level
  - Media description (m)
  - Media info (i)
  - Connection info (c)
    - Optional if specified at the session level
  - Bandwidth info (b)
  - Encryption key (k)
  - Attributes (a)

Subfields [1/3]

- Field = <value of subfield1> <value of subfield2> <value of subfield3>.
- Origin
  - Username, the originator's login id or "-"
  - session ID
    - A unique ID
    - Make use of NTP timestamp
  - version, a version number for this particular session
  - network type
    - A text string
    - IN refers to Internet
  - address type
    - IP4, IP6
  - address, a fully-qualified domain name or the IP address

Subfields [2/3]

- Connection Data
  - The network and address at which media data will be received
  - Network type
  - Address type
  - Connection address
- Media Information
  - Media type
    - Audio, video, data, or control
  - Port
  - Format
    - List the various types of media format that can be supported
    - According to the RTP audio/video profile
  - m= audio 45678 RTP/AVP 15 3 0
    - G.728, GSM, G.711

Subfields [3/3]

- Attributes
  - To enable additional information to be included
  - Property attribute
    - a=sendonly
    - a=recvonly
  - value attribute
    - a=orient:landscape used in a shared whiteboard session
  - rtpmap attribute
    - The use of dynamic payload type
    - a=rtpmap:<payload type> <encoding name>/<clock rate> [/<encoding parameters>].
    - m=video 54678 RTP/AVP 98
    - a=rtpmap 98 L16/16000/2
      - 16-bit linear encoded stereo (2 channels) audio sampled at 16kHz

Usage of SDP with SIP

- SIP and SDP make a wonderful partnership for the transmission of session information.
- SIP provides the messaging mechanism for the establishment of multimedia sessions.
- SDP provides a structured language for describing the sessions.
  - The entity headers identifies the message body.
    - Content-Length: 1280
    - Content-Type: application/sdp

Negotiation of Media

- Fig 5-15
  - G.728 is selected

Daniel<sip:Collins@station1.work.com>

Boss<sip:Manager@station2.work.com>

a

INVITE sip:Manager@station2.work.com SIP/2.0
From: Daniel<sip:Collins@station1.work.com>; tag = abcd1234
To: Boss<sip:Manager@station2.work.com>
CSeq: 1 INVITE
Content-Length: 230
Content-Type: application/sdp
Content-Disposition: session

v=0
o=collins 123456 001 IN IP4 station1.work.com
s=
c=IN IP4 station1.work.com
t=0 0
m=audio 4444 RTP/AVP 2 4 15
a=rtpmap 2 G726-32/8000
a=rtpmap 4 G723/8000
a=rtpmap 15 G728/8000

b

SIP/2.0 200 OK
...

b

```
SIP/2.0 200 OK
From: Daniel<sip:Collins@station1.work.com>; tag = abcd1234
To: Boss<sip:Manager@station2.work.com>; tag = xyz789
CSeq: 1 INVITE
Content-Length: 161
Content-Type: application/sdp
Content-Disposition: session

v=0
o=collins 45678 001 IN IP4 station2.work.com
s=
c=IN IP4 station2.work.com
t=0 0
m=audio 6666 RTP/AVP 15
a=rtpmap 15 G728/8000
```

c

```
ACK sip:manager@station2.work.com SIP/2.0
From: Daniel<sip:Collins@station1.work.com>; tag = abcd1234
To: Boss<sip:Manager@station2.work.com>; tag = xyz789
CSeq: 1 ACK
Content-Length: 0
```

d

Conversation

Multiple Media Streams

- INVITE with multiple media streams
  - Unsupported should also be returned with a port number of zero
- An alternative
  - INVITE

    m=audio 4444 RTP/AVP 2 4 15

    a=rtpmap 2 G726-32/8000

    a=rtpmap 4 G723/8000

    a=rtpmap 15 G728/8000

    m=video 6666 RTP/AVP 34

    a=rtpmap 34 H263/9000/2

Daniel<sip:Collins@station1.work.com> **Boss<sip:Manager@station2.work.com>**

a

INVITE sip:Manager@station2.work.com SIP/2.0
From: Daniel<sip:Collins@station1.work.com>; tag = abcd1234
To: Boss<sip:Manager@station2.work.com>
CSeq: 1 INVITE
Content-Length: 213
Content-Type: application/sdp
Content-Disposition: session

v=0
o=collins 123456 001 IN IP4 station1.work.com
s=
c=IN IP4 station1.work.com
t=0 0
m=audio 4444 RTP/AVP 2 4 15
a=rtpmap 2 G726-32/8000
a=rtpmap 4 G723/8000
a=rtpmap 15 G728/8000
m=video 6666 RTP/AVP 34
a=rtpmap 34 H263/9000/2

b

SIP/2.0 200 OK
…

b

SIP/2.0 200 OK
From: Daniel<sip:Collins@station1.work.com>; tag = abcd1234
To: Boss<sip:Manager@station2.work.com>; tag = xyz789
CSeq: 1 INVITE
Content-Length: 163
Content-Type: application/sdp
Content-Disposition: session

v=0
o=collins 45678 001 IN IP4 station2.work.com
s=
c=IN IP4 station2.work.com
t=0 0
m=audio 9000 RTP/AVP 15
a=rtpmap 15 G728/8000
m=video 0 RTP/AVP 34

c

ACK sip:manager@station2.work.com SIP/2.0
From: Daniel<sip:Collins@station1.work.com>; tag = abcd1234
To: Boss<sip:Manager@station2.work.com>; tag = xyz789
CSeq: 1 ACK
Content-Length: 0

d

Conversation

SIP and SDP Offer/Answer Model

- Re-INVITE is issued when the callee replies with more than one codec.
  - With the same dialog identifier (To and From headers, including tag values), Call-ID and Request-URI
  - The session version is increased by 1 in **o=** line of message body.

- RFC 3264 - An Offer/Answer Model with SDP

Daniel<sip:Collins@station1.work.com> **Boss<sip:Manager@station2.work.com>**

a

INVITE sip:manager@station2.work.com SIP/2.0
CSeq: 1 INVITE
Content-Length: 183
Content-Type: application/sdp
Content-Disposition: session

v=0
o=collins 123456 001 IN IP4 station1.work.com
s=
c=IN IP4 station1.work.com
t=0 0
m=audio 4444 RTP/AVP 2 4 15
a=rtpmap 2 G726-32/8000
a=rtpmap 4 G723/8000
a=rtpmap 15 G728/8000
a=inactive

b

SIP/2.0 200 OK
CSeq: 1 INVITE
Content-Length: 157
Content-Type: application/sdp
Content-Disposition: session

v=0
o=collins 45678 001 IN IP4 station2.work.com
s=
c=IN IP4 station2.work.com
t=0 0
m=audio 6666 RTP/AVP 4
a=rtpmap 4 G723/8000

Daniel<sip:Collins@station1.work.com> **Boss<sip:Manager@station2.work.com>**

c

ACK sip:manager@station2.work.com SIP/2.0
From: Daniel<sip:Collins@station1.work.com>; tag = abcd1234
To: Boss<sip:Manager@station2.work.com>; tag = xyz789
CSeq: 1 ACK
Content-Length: 0

d

INVITE sip:manager@station2.work.com SIP/2.0
CSeq: 2 INVITE
Content-Length: 126
Content-Type: application/sdp
Content-Disposition: session

v=0
o=collins 123456 002 IN IP4 station1.work.com
s=
c=IN IP4 station1.work.com
t=0 0
m=audio 4444 RTP/AVP 15
a=rtpmap 15 G728/8000

OPTIONS Method

- The SIP method OPTIONS allows a UA to query another UA or a proxy server as to its capabilities.
- This allows a client to discover information about the supported methods, content types, extensions, codecs, etc. without "ringing" the other party.
- Accept Header Field
  - Indicate the type of information that the sender hopes to receive
  - e.g. Accept: application/sdp, text/html
- Allow Header Field
  - Indicate the SIP methods
  - e.g. Allow: INVITE, ACK, OPTIONS, CANCEL, BYE, INFO
- Supported Header Field
  - Indicate the SIP extensions that can be supported

Daniel<sip:Collins@station1.work.com>

Boss<sip:Manager@station2.work.com>

a

OPTIONS sip:manager@station2.work.com SIP/2.0
Via: SIP/2.0/UDP Station1.work.com; branch=z9hG4bK7890123
From: Daniel<sip:Collins@work.com>; tag=lmnop123
To: Boss<sip:Manager@station2.work.com>
Call-ID: 123456@station1.work.com
Contact: Daniel<sip:Collins@station1.work.com>
CSeq: 1 OPTIONS
Accept: application/sdp
Content-Length: 0

b

SIP/2.0 200 OK
Via: SIP/2.0/UDP Station1.work.com; branch=z9hG4bK7890123
From: Daniel<sip:Collins@work.com>; tag=lmnop123
To: Boss<sip:Manager@station2.work.com>; tag=xyz5678
Call-ID: 123456@station1.work.com
CSeq: 1 OPTIONS
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Length: 146
Content-Type: application/sdp

v=0
o=manager 45678 001 IN IP4 station2.work.com
s=
c=IN IP4 station2.work.com
t=0 0
m=audio 0 RTP/AVP 4 15
a=rtpmap 4 G723/8000
a=rtpmap 15 G728/8000

SIP Extensions and Enhancements

- RFC 2543, March 1999
  - SIP has attracted enormous interest.
  - Traditional telecommunications companies, cable TV providers and ISP
- A large number of extensions to SIP have been proposed.
  - SIP will be enhanced considerably before it becomes an Internet standard.

The Supported Header

- The Require Header
  - In request, a UA client indicates that a UA server must support certain extension.
- How does the server require the client to support some extension?
  - If the request did not indicate that the client suports the extension, send a response
    - 421, Extension Required
- The Supported header
  - Supported: Felix

SIP INFO Method

- Specified in RFC 2976
  - For transferring information during an ongoing session
- Some example uses of the INFO method
  - DTMF digits
  - Account balance information in a Pre-paid service
  - mid-call signaling information

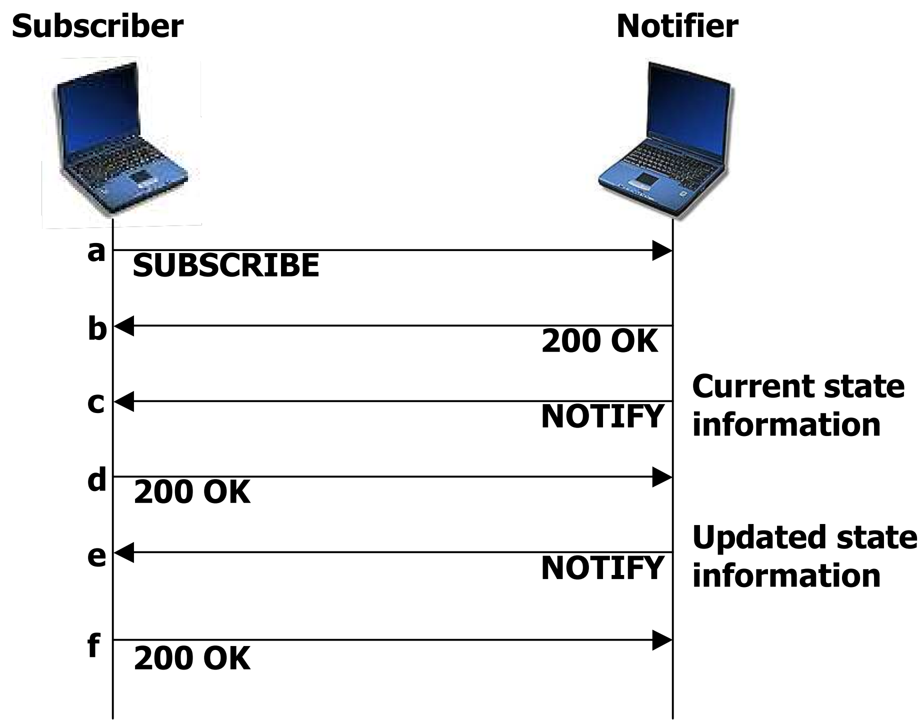
DTMF

- Dual-Tone Multi-frequency (DTMF) digits

Digit	1	2	3	4	5	6
Low Frequency	697Hz	697	697	770	770	770
High Frequency	1209Hz	1336	1477	1209	1336	1477
Digit	7	8	9	*	0	#
Low Frequency	852	852	852	941	941	941
High Frequency	1209	1336	1477	1209	1336	1477

SIP Event Notification

- Several SIP-based applications have been devised based on the concept of a user being informed of some event.
  - E.g., Presence
- RFC 3265 has addressed the issue of event notification.
  - SUBSCRIBE and NOTIFY
  - The Event header



SIP for Instant Messaging

- The IETF working group – SIP for Instant Messaging and Presence Leveraging Extensions (SIMPLE)
- A new SIP method – MESSAGE (RFC 3428)
  - This request carries the actual message in a message body.
  - A MESSAGE request does not establish a SIP dialog.

Boss<sip:Manager@pc1.home.com> **sip:Server.work.com** **Daniel<sip:Collins@station1.work.com>**



a (Boss → Server)

MESSAGE sip:Collins@work.com SIP/2.0
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
Max-Forwards: 70
From: Boss<sip:Manager@home.net>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 MESSAGE
Content-Type: text/plain
Content-Length: 19
Content-Disposition: render

Hello. How are you?

b (Server → Daniel)

MESSAGE sip:Collins@work.com SIP/2.0
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bKxyz1
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
Max-Forwards: 69
From: Boss<sip:Manager@home.net>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 MESSAGE
Content-Type: text/plain
Content-Length: 19
Content-Disposition: render

Hello. How are you?

c (Daniel → Server)

SIP/2.0 200 OK
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bKxyz1
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
From: Boss<sip:Manager@home.net>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 MESSAGE
Content-Length: 0

d (Server → Boss)

SIP/2.0 200 OK
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
From: Boss<sip:Manager@home.net>
To: Daniel<sip:Collins@work.com>
Call-ID: 123456@pc1.home.net
CSeq: 1 MESSAGE
Content-Length: 0

Boss<sip:Manager@pc1.home.com> sip:Server.work.com Daniel<sip:Collins@station1.work.com>



e

MESSAGE sip:Manager@home.net SIP/2.0
Via: SIP/2.0/UDP station1.work.com; branch=z9hG4bK123
Max-Forwards: 70
From: Daniel<sip:Collins@work.com>
To: Boss<sip:Manager@home.net>
Call-ID: 456789@station1.work.com
CSeq: 1101 MESSAGE
Content-Type: text/plain
Content-Length: 22
Content-Disposition: render

I'm fine. How are you?

f

MESSAGE sip:Manager@home.net SIP/2.0
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bKabcd
Via: SIP/2.0/UDP station1.work.com; branch=z9hG4bK123
Max-Forwards: 69
From: Daniel<sip:Collins@work.com>
To: Boss<sip:Manager@home.net>
Call-ID: 456789@station1.work.com
CSeq: 1101 MESSAGE
Content-Type: text/plain
Content-Length: 22
Content-Disposition: render

I'm fine. How are you?

g

SIP/2.0 200 OK
Via: SIP/2.0/UDP server.work.com; branch=z9hG4bKabcd
Via: SIP/2.0/UDP station1.work.com; branch=z9hG4bK123
From: Daniel<sip:Collins@work.com>
To: Boss<sip:Manager@home.net>
Call-ID: 456789@station1.work.com
CSeq: 1101 MESSAGE
Content-Length: 0

h

SIP/2.0 200 OK
Via: SIP/2.0/UDP station1.work.com; branch=z9hG4bK123
From: Daniel<sip:Collins@work.com>
To: Boss<sip:Manager@home.net>
Call-ID: 456789@station1.work.com
CSeq: 1101 MESSAGE
Content-Length: 0

RFC 3515 - SIP REFER Method

- To enable the sender of the request to instruct the receiver to contact a third party
  - With the contact details for the third party included within the REFER request
  - For Call Transfer applications
- The Refer-to: and Referred-by: Headers
- The dialog between Mary and Joe remains established.
  - Joe could return to the dialog after consultation with Susan.


sip:Mary@station1.work.com
sip:Joe@station2.work.com
sip:Susan@station3.work.com
a
REFER sip:Joe@station2.work.com SIP/2.0
Via: SIP/2.0/UDP station1.work.com; branch=z9hG4bK789
Max-Forwards: 70
From: Mary<sip:Mary@work.com>; tag=123456
To: Joe<sip:Joe@work.com>; tag=67890
Contact: Mary<Mary@station1.work.com>
Refer-To: Sussan<sip:Sussan@station3.work.com>
Call-ID: 123456@station1.work.com
CSeq: 123 REFER
Content-Length: 0
b
SIP/2.0 202 Accepted
Via: SIP/2.0/UDP station1.work.com; branch=z9hG4bK789
From: Mary<sip:Mary@work.com>; tag=123456
To: Joe<sip:Joe@work.com>; tag=67890
Contact: Joe<Joe@station2.work.com>
Call-ID: 123456@station1.work.com
CSeq: 123 REFER
Content-Length: 0
c
Invite sip:Susan@station3.work.com SIP/2.0
Via: SIP/2.0/UDP station2.work.com; branch=z9hG4bKxyz1
Max-Forwards: 70
From: Joe<sip:Joe@work.com>; tag=abcxyz
To: Susan<sip:Susan@station3.work.com>
Contact: Joe<Joe@station2.work.com>
Call-ID: 67890@station2.work.com
CSeq: 567 INVITE
Content-Type: application/sdp
Content-Length: xx
Content-Disposition: session
{message body}

sip:Mary@station1.work.com **sip:Joe@station2.work.com** **sip:Susan@station3.work.com**

e

SIP/2.0 200 OK
Via: SIP/2.0/UDP station2.work.com; branch=z9hG4bKxyz1
From: Joe<sip:Joe@work.com>; tag=abcxyz
To: Susan<sip:Susan@station3.work.com>; tag=123xyz
Call-ID: 67890@station2.work.com
CSeq: 567 INVITE
Content-Type: application/sdp
Content-Length: xx
Content-Disposition: session
{message body}

f

ACK sip:Susan@station3.work.com SIP/2.0
Via: SIP/2.0/UDP station2.work.com; branch=z9hG4bKxyz1
Max-Forwards: 70
From: Joe<sip:Joe@work.com>; tag=abcxyz
To: Susan<sip:Susan@station3.work.com>; tag=123xyz
Call-ID: 67890@station2.work.com
CSeq: 567 ACK
Content-Length: 0

g

NOTIFY sip:Mary@station1.work.com SIP/2.0
Via: SIP/2.0/UDP station2.work.com; branch=z9hG4bK123
Max-Forwards: 70
To: Joe<sip:Joe@work.com>; tag=67890
From: Mary<sip:Mary@work.com>; tag=123456
Contact: Joe<Joe@station2.work.com>
Call-ID: 123456@station1.work.com
CSeq: 124 NOTIFY
Content-Type: message/sipfrag;version=2.0
Content-Length: 15

SIP/2.0 200 OK

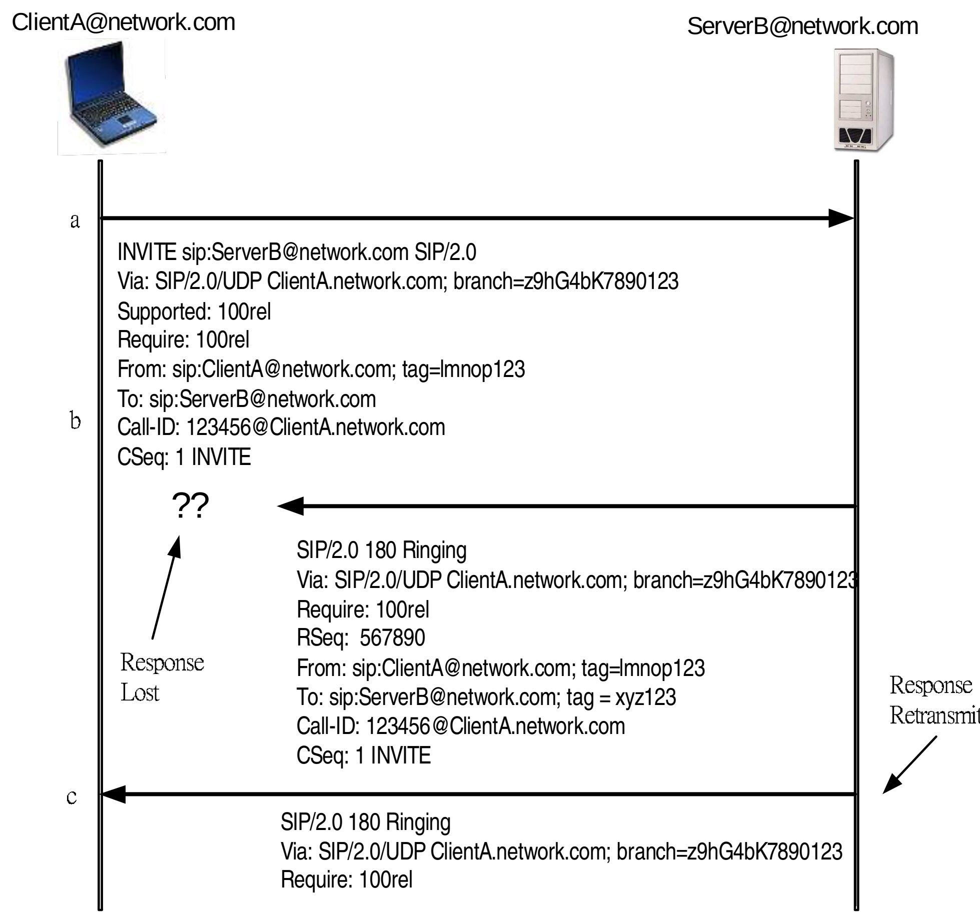
h

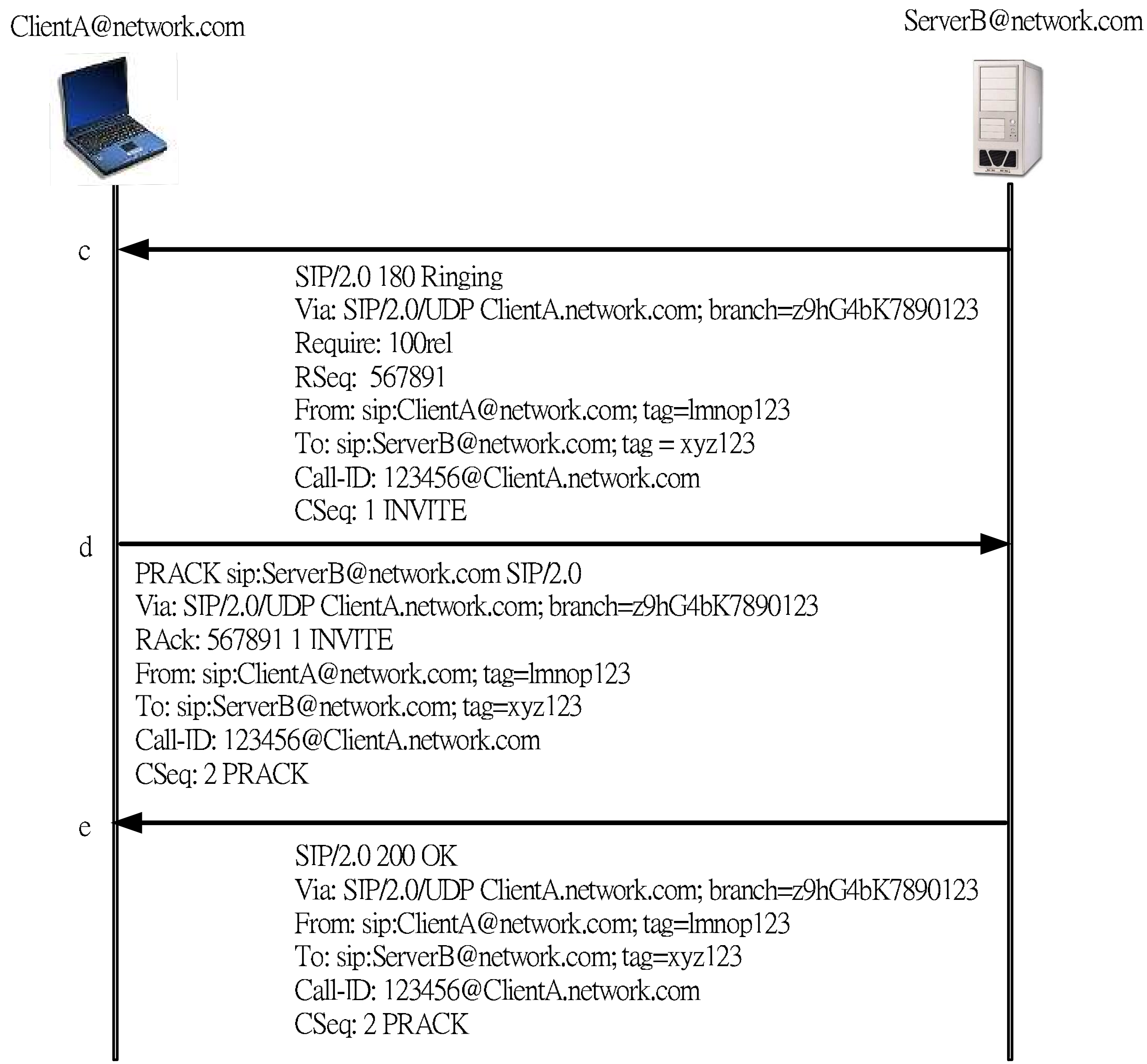
SIP/2.0 200 OK
Via: SIP/2.0/UDP station2.work.com; branch=z9hG4bK123
To: Joe<sip:Joe@work.com>; tag=67890
From: Mary<sip:Mary@work.com>; tag=123456
Call-ID: 123456@station1.work.com
CSeq: 124 NOTIFY
Content-Length: 0

Reliability of Provisional Responses

- Provisional Responses
  - 100 (trying), 180 (ringing), 183 (session in progress)
  - Are not answered with an ACK
- If the messages is sent over UDP
  - Unreliable
- Lost provisional response may cause problems when interoperating with other network
  - 180, 183 → Q.931 alerting or ISUP ACM
  - To drive a state machine
  - E.g., a call to an unassigned number
    - ACM to create a one-way path to relay an announcement such as "The number you have called has been changed"
    - If the provisional response is lost, the caller is left in the dark and does not understand why the call did not connect.

- RFC 3262
  - Reliability of Provisional Responses in SIP
- Supported: 100rel
- RSeq Header
  - Response Seq
  - +1, when retxm
- RAck Header
  - Response ACK
  - In PRACK
  - RSeq+CSeq
- PRACK
  - Prov. Resp. ACK
- Should not
  - Apply to 100
- Default timer value = 0.5 s




ClientA@network.com
ServerB@network.com
c
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ClientA.network.com; branch=z9hG4bK7890123
Require: 100rel
RSeq: 567891
From: sip:ClientA@network.com; tag=lmnop123
To: sip:ServerB@network.com; tag = xyz123
Call-ID: 123456@ClientA.network.com
CSeq: 1 INVITE
d
PRACK sip:ServerB@network.com SIP/2.0
Via: SIP/2.0/UDP ClientA.network.com; branch=z9hG4bK7890123
RAck: 567891 1 INVITE
From: sip:ClientA@network.com; tag=lmnop123
To: sip:ServerB@network.com; tag=xyz123
Call-ID: 123456@ClientA.network.com
CSeq: 2 PRACK
e
SIP/2.0 200 OK
Via: SIP/2.0/UDP ClientA.network.com; branch=z9hG4bK7890123
From: sip:ClientA@network.com; tag=lmnop123
To: sip:ServerB@network.com; tag=xyz123
Call-ID: 123456@ClientA.network.com
CSeq: 2 PRACK

Integration of SIP Signaling and Resource Management

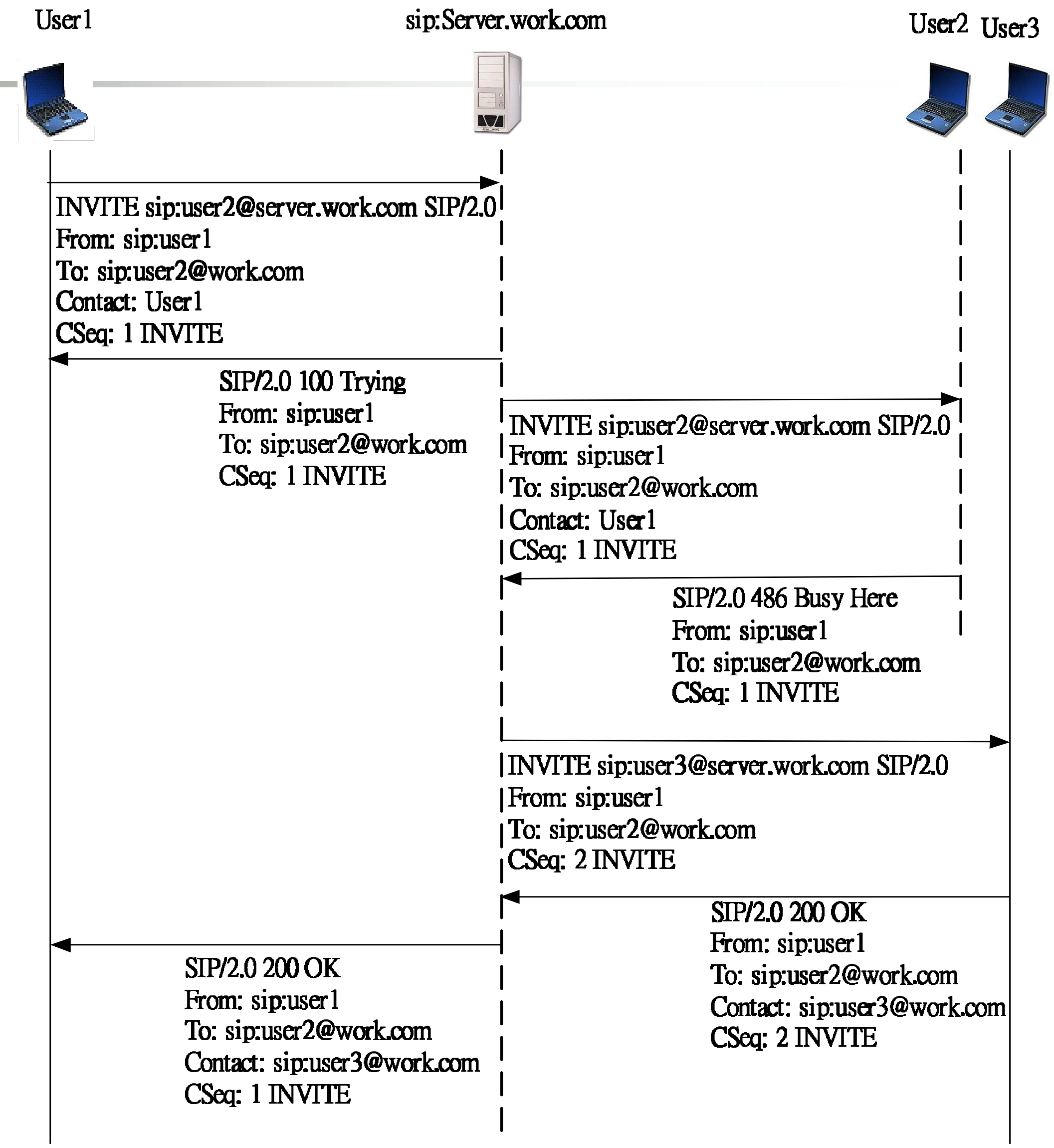
- Ensuring that sufficient resources are available to handle a media stream is a very important.
  - To provide a high-quality service for a carrier-grade network
- The signaling might take a different path from the media.
  - The successful transfer of signaling messages does not imply to a successful transfer of media.
- Assume resource-reservation mechanisms are available (Chapter 8)
  - On a per-session basis
    - End-to-end network resources are reserved as part of session establishment.
  - On an aggregate basis
    - A certain amount of network resources are reserved in advance for a certain type of usage.
    - Policing functions at the edge of the network

Usage of SIP for Features/Services

- Call-transfer application (with REFER method)
- Personal Mobility through the use of registration
- One number service through forking proxy
- Call-completion services by using Retry-After: header
- To carry MIME content as well as an SDP description
- SIP address is a URL
  - Click-to-call applications
- The existing supplementary services in traditional telephony
  - Call waiting, call forwarding, multi-party calling, call screening
- Proxy invokes various types of advanced feature logic.
  - Policy server (call-routing, QoS)
  - Authentication server
  - Use the services of an IN SCP over INAP

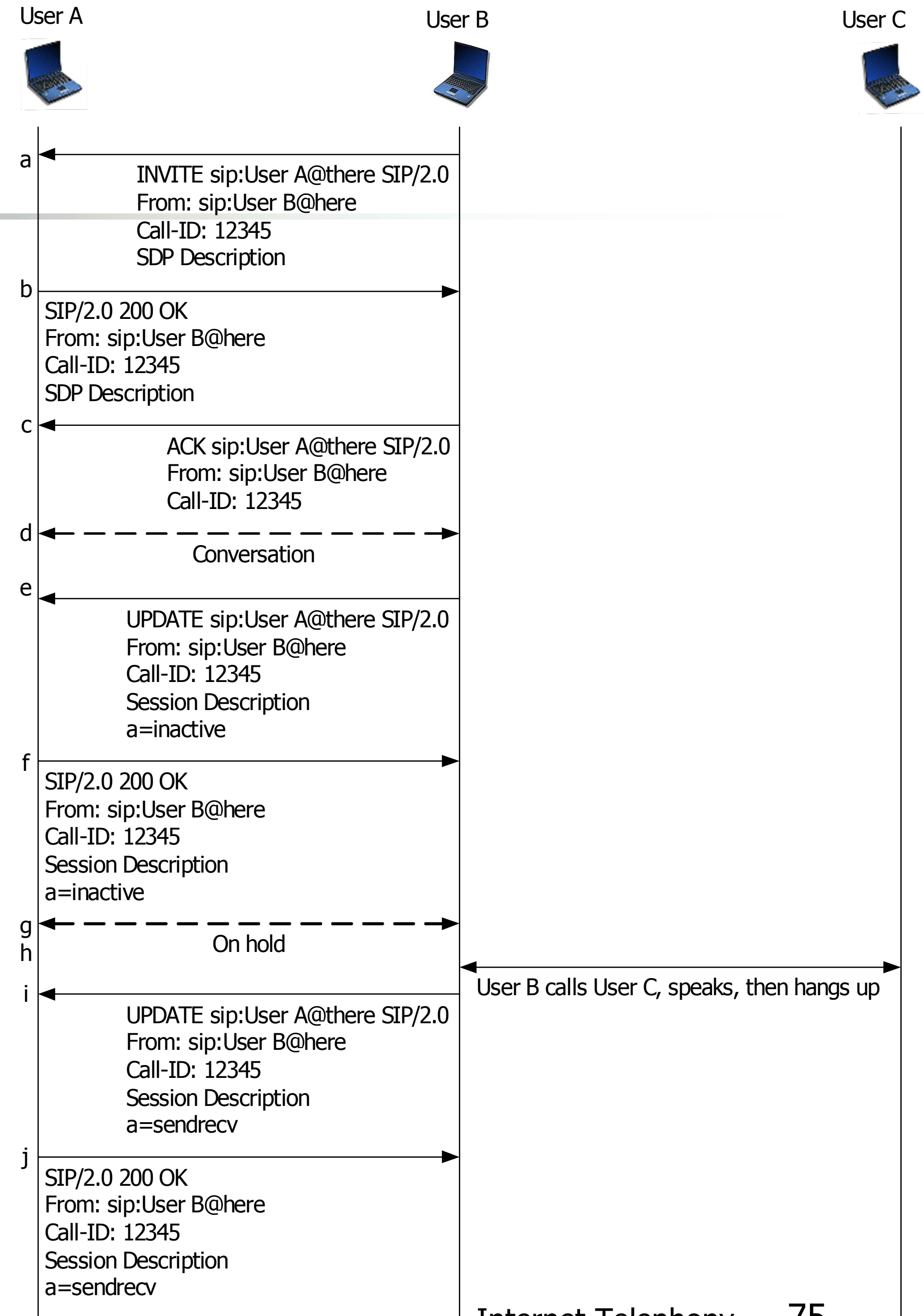
Call Forwarding

- On busy
- 486, busy here
- With the same To, User 3 can recognize that this call is a forwarded call, originally sent to User 2.
- Contact: header in 200 response
- Call-forwarding-on-no-answer
  - Timeout
  - CANCEL method



Consultation Hold

- User A asks User B a question, and User B need to check with User C for the correct answer.
- User B could use the REFER method to transfer the call to User C.



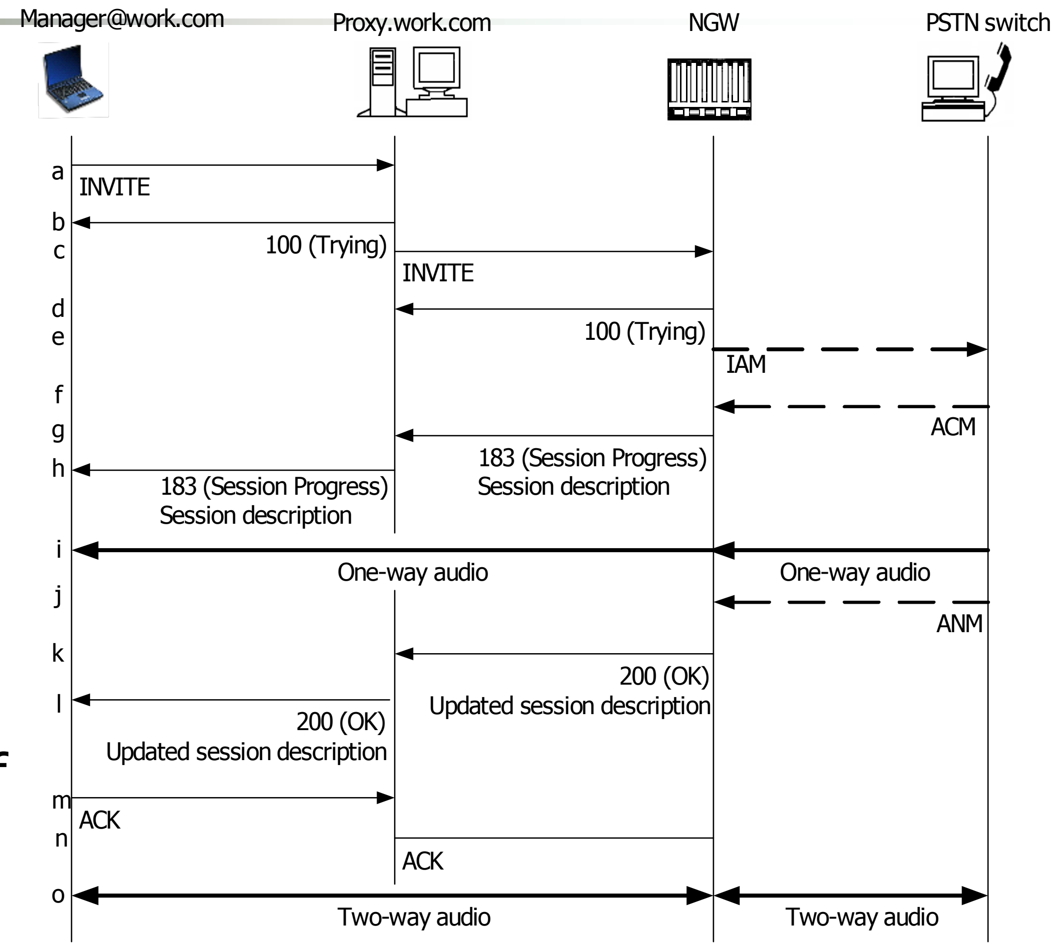
Music on Hold

183 Session Progress

- It has been included within the revised SIP spec.
  - To open one-way audio path from called end to calling end
    - From the called party to calling party
    - Enable in-band call progress information to be transmitted
      - Tones or announcements
  - Interworking with SS7 network
    - ACM (Address Complete Message)
    - For SIP-PSTN-SIP connections

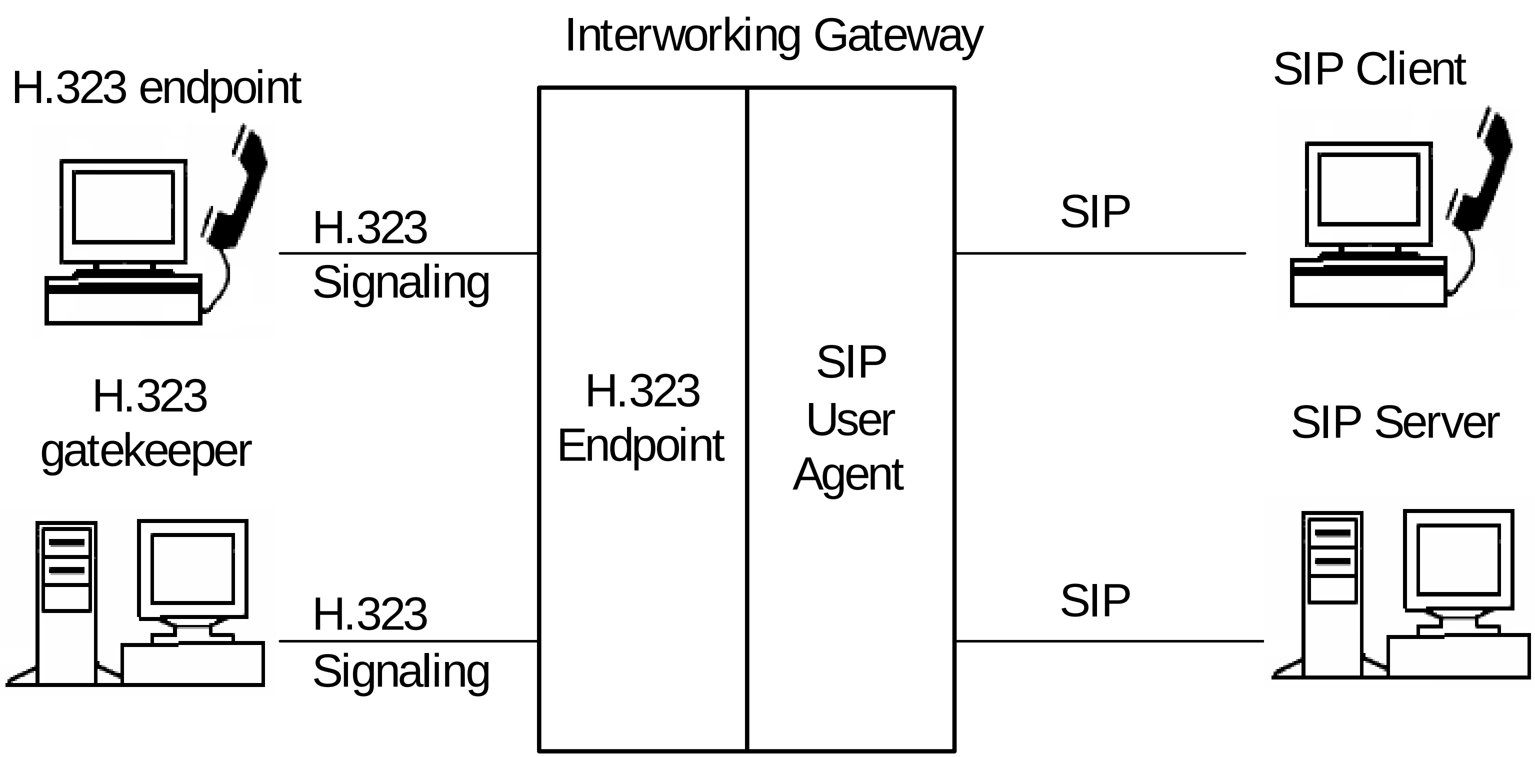
PSTN Interworking

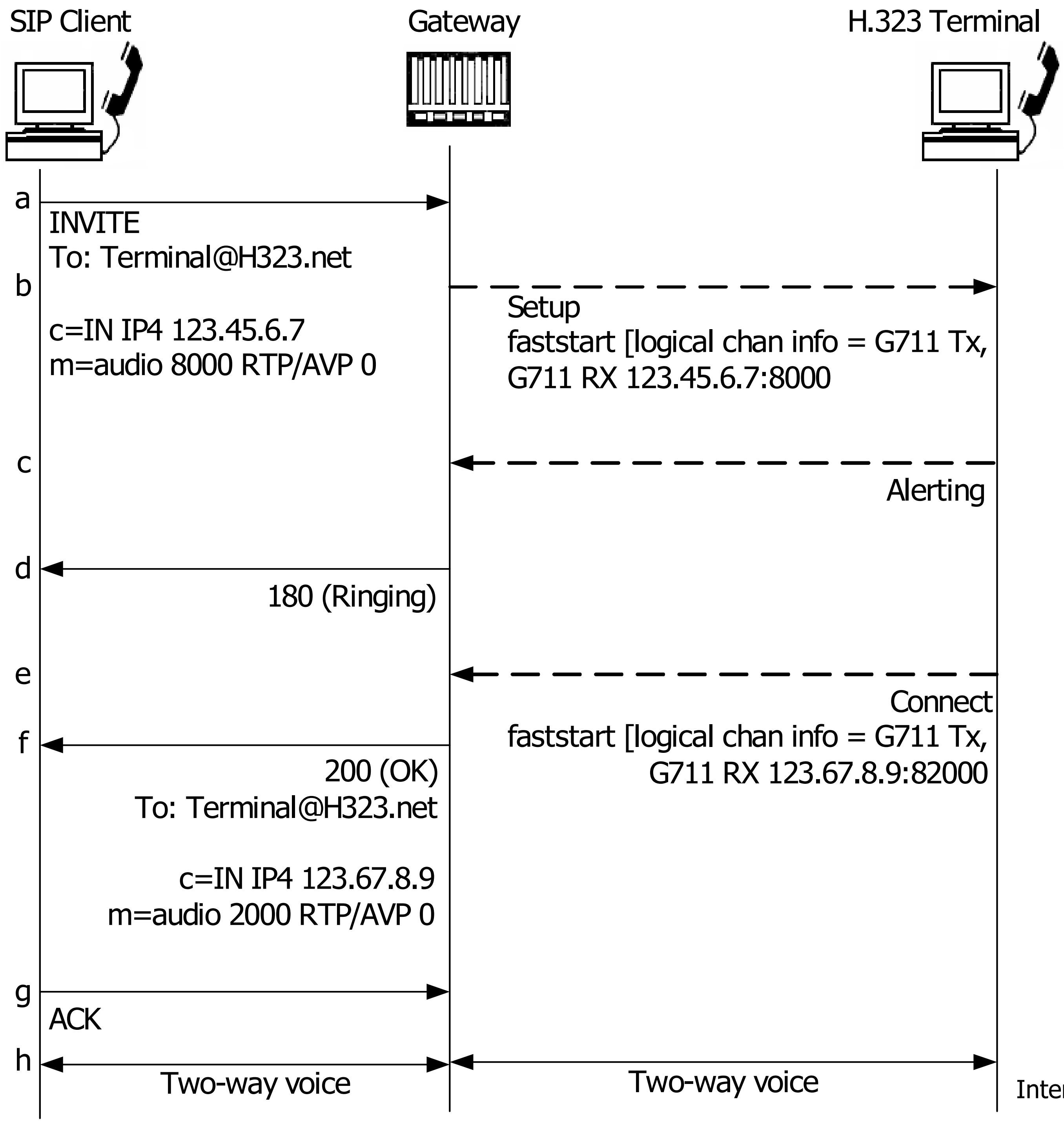
- PSTN Interworking
  - A SIP URL to a telephone number
  - A network gateway
- PSTN – SIP – PSTN
  - MIME media types
  - For ISUP
- SIP for Telephony (SIP-T)
- The whole issue of interworking with SS7 is fundamental to the success of VoIP in the real world.

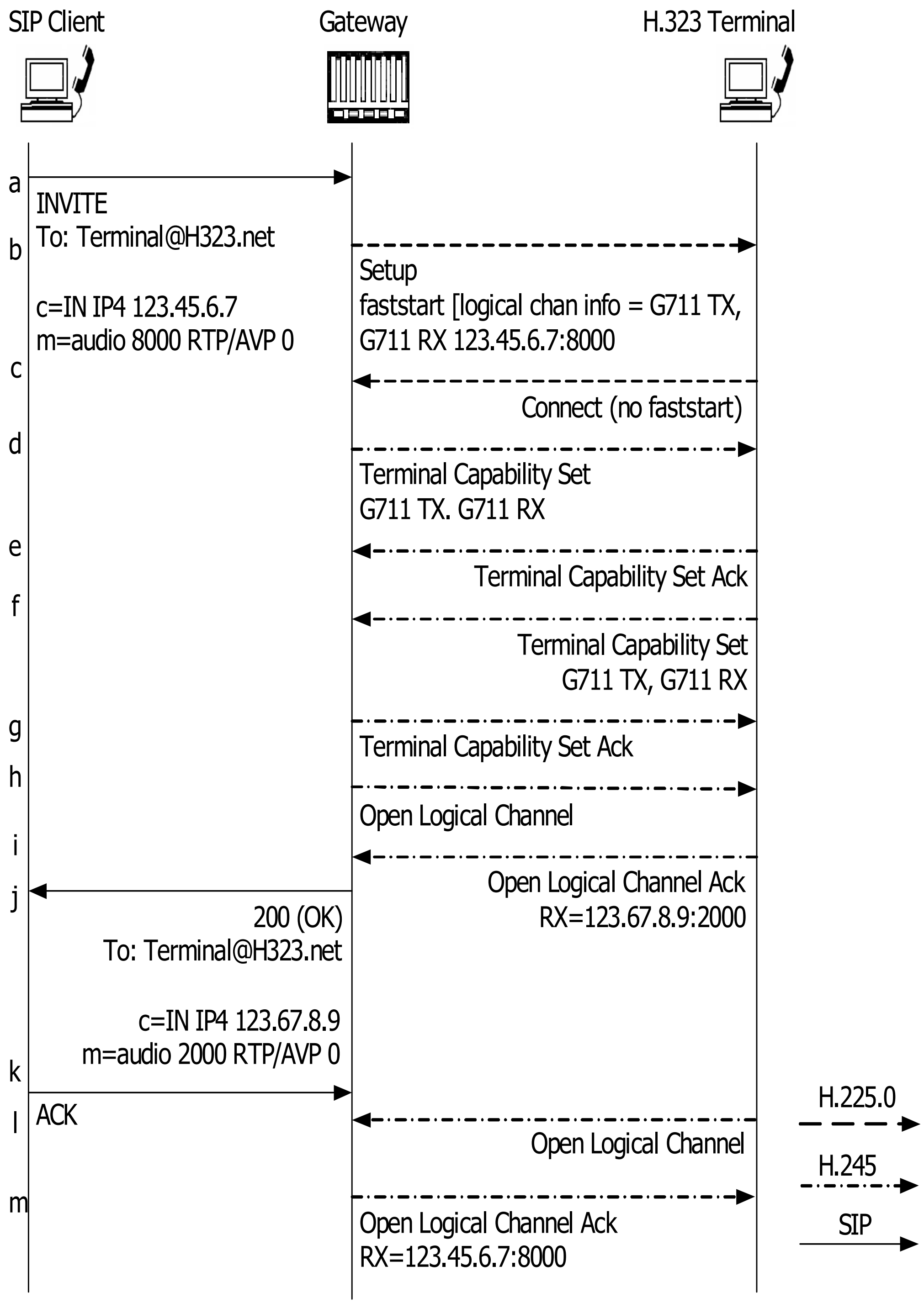


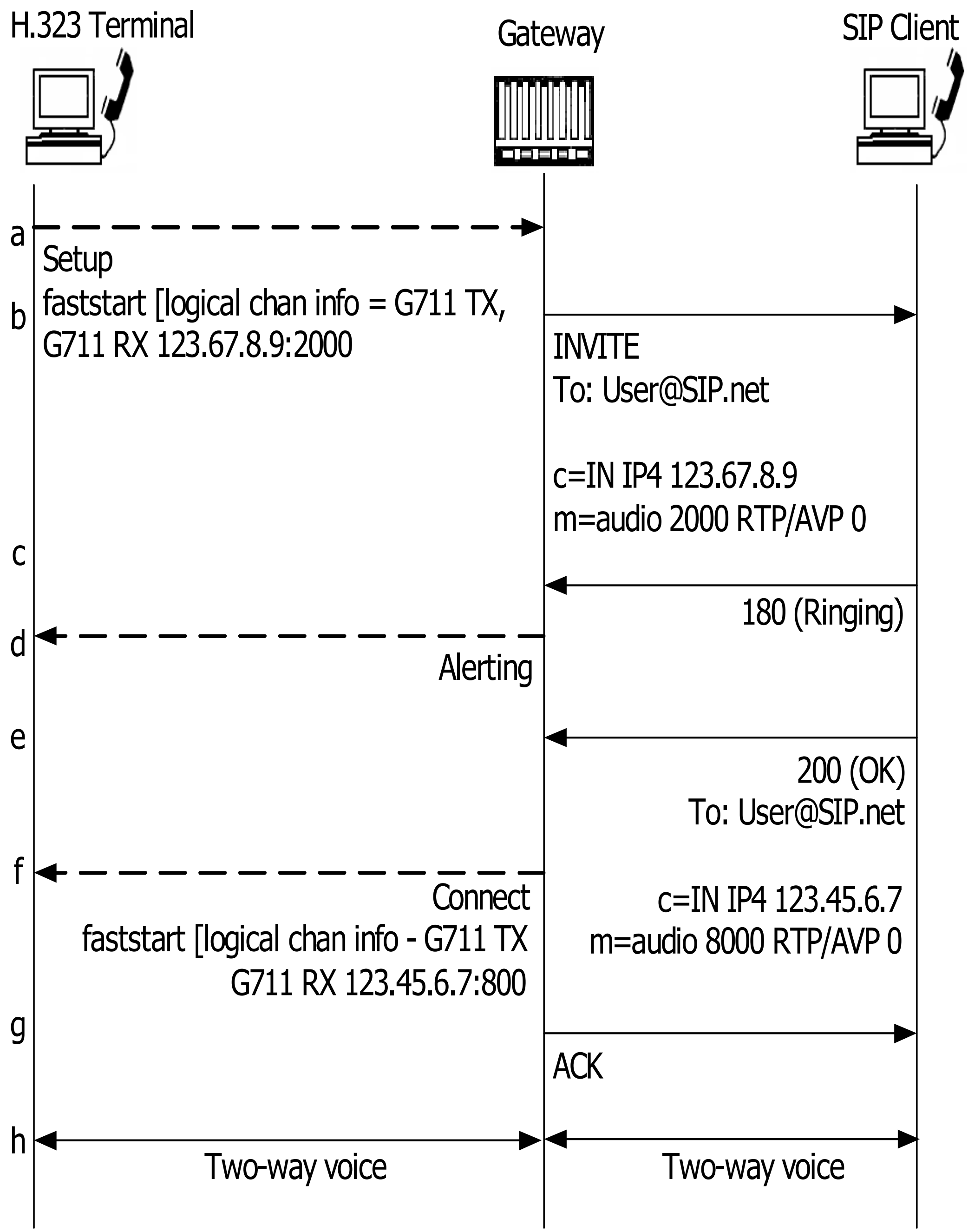
Interworking with H.323

- SIP-H.323 interworking gateway




SIP Client
Gateway
H.323 Terminal
a
INVITE
To: Terminal@H323.net
b
c=IN IP4 123.45.6.7
m=audio 8000 RTP/AVP 0
Setup
faststart [logical chan info = G711 Tx,
G711 RX 123.45.6.7:8000
c
Alerting
d
180 (Ringing)
e
Connect
faststart [logical chan info = G711 Tx,
G711 RX 123.67.8.9:82000
f
200 (OK)
To: Terminal@H323.net
c=IN IP4 123.67.8.9
m=audio 2000 RTP/AVP 0
g
ACK
h
Two-way voice
Two-way voice


SIP Client
Gateway
H.323 Terminal
a
INVITE
To: Terminal@H323.net
b
c=IN IP4 123.45.6.7
m=audio 8000 RTP/AVP 0
Setup
faststart [logical chan info = G711 TX,
G711 RX 123.45.6.7:8000
c
Connect (no faststart)
d
Terminal Capability Set
G711 TX. G711 RX
e
Terminal Capability Set Ack
f
Terminal Capability Set
G711 TX, G711 RX
g
Terminal Capability Set Ack
h
Open Logical Channel
i
Open Logical Channel Ack
RX=123.67.8.9:2000
j
200 (OK)
To: Terminal@H323.net
c=IN IP4 123.67.8.9
m=audio 2000 RTP/AVP 0
k
ACK
l
Open Logical Channel
m
Open Logical Channel Ack
RX=123.45.6.7:8000
H.225.0
H.245
SIP


H.323 Terminal
Gateway
SIP Client
a
Setup
faststart [logical chan info = G711 TX,
G711 RX 123.67.8.9:2000
b
INVITE
To: User@SIP.net
c=IN IP4 123.67.8.9
m=audio 2000 RTP/AVP 0
c
180 (Ringing)
d
Alerting
e
200 (OK)
To: User@SIP.net
c=IN IP4 123.45.6.7
m=audio 8000 RTP/AVP 0
f
Connect
faststart [logical chan info - G711 TX
G711 RX 123.45.6.7:800
g
ACK
h
Two-way voice
Two-way voice

Summary

- The future for signaling in VoIP networks
  - Simple, yet flexible
  - Easier to implement
  - Fit well with the media gateway control protocols
- SIP is the protocol of choice for the evolution of third-generation wireless networks.
  - SIP-based mobile devices will become available
  - SIP-based network elements will be introduced within mobile networks.

Presentation List

- 11/17
  - RFC 3960 - Early Media and Ringing Tone Generation in SIP (朱恆德)
  - RFC 3711 - Secure RTP (SRTP) (黃泰榮)
  - RFC 3361 - DHCP Option for SIP Servers. (吳菖育)
- 11/24
  - RFC 3486 - Compressing the Session Initiation Protocol (SIP) (曾朝弘)
  - RFC 3824 - Using E.164 numbers with SIP (陳瑞邦)
  - RFC 4168 - SCTP as a Transport for SIP (謝應能)
- 12/1
  - RFC 4028 - Session Timers in SIP (林穎舜)
  - RFC 4123 - SIP-H.323 Interworking Requirements (王彥翔)
  - RFC 3372 - Session Initiation Protocol for Telephones (SIP-T): Context and Architectures. (郭致宏)

Available Articles

- RFC 3050 - Common Gateway Interface for SIP
- RFC 3329 - Security Mechanism Agreement for SIP
- RFC 3351 - User Requirements for SIP in Support of Deaf, Hard of Hearing and Speech-impaired Individuals.
- RFC 3581 - An Extension to SIP for Symmetric Response Routing
- RFC 3702 - AAA Requirements for SIP
- RFC 3903 - SIP Extension for Event State Publication