# Session Initiation Protocol (SIP)

# Introduction

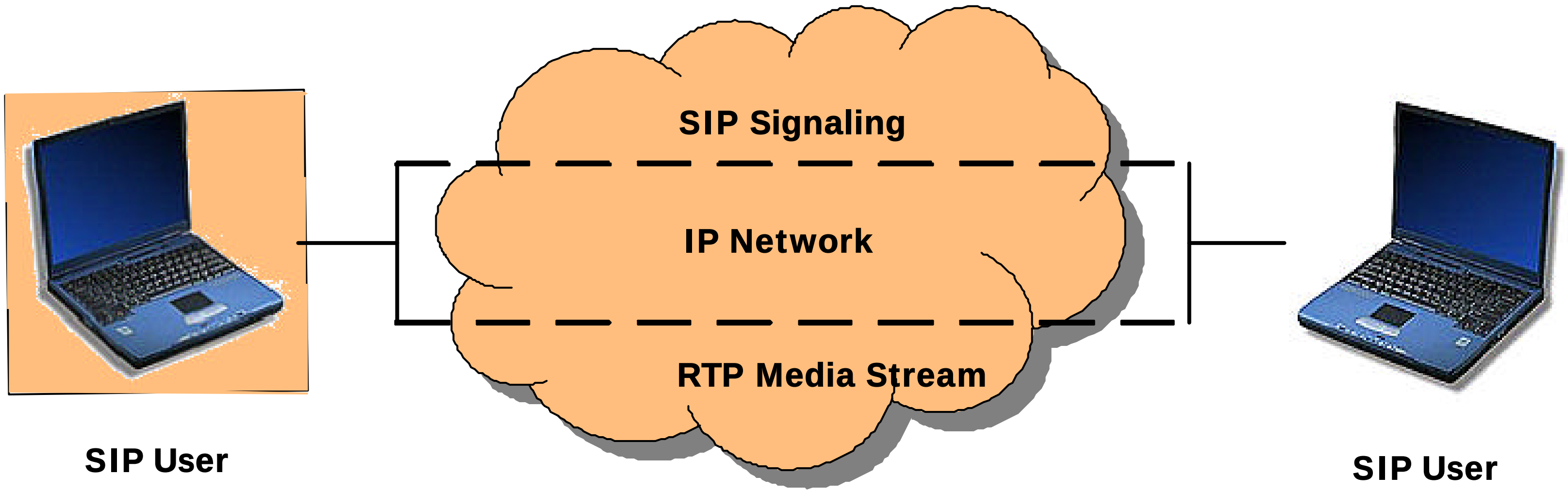
- A powerful alternative to H.323
- More flexible, simpler
- Easier to implement
  - Advanced features
- Better suited to the support of intelligent user devices
- A part of IETF multimedia data and control architecture
  - SDP, RTSP (Real-Time Streaming Protocol), SAP (Session Announcement Protocol)

# 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 (June 2002 )
- SIP + MGCP/MEGACO
  - The VoIP signaling in the future
- “bake-offs” or SIP Interoperability Tests
  - The development of SIP and its implementation by system developers has involved a number of events.
  - 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 to potential session participants
- Separate signaling and media streams
  - Signaling may pass via one or more proxy or redirect servers
  - Media stream takes a more direct path.

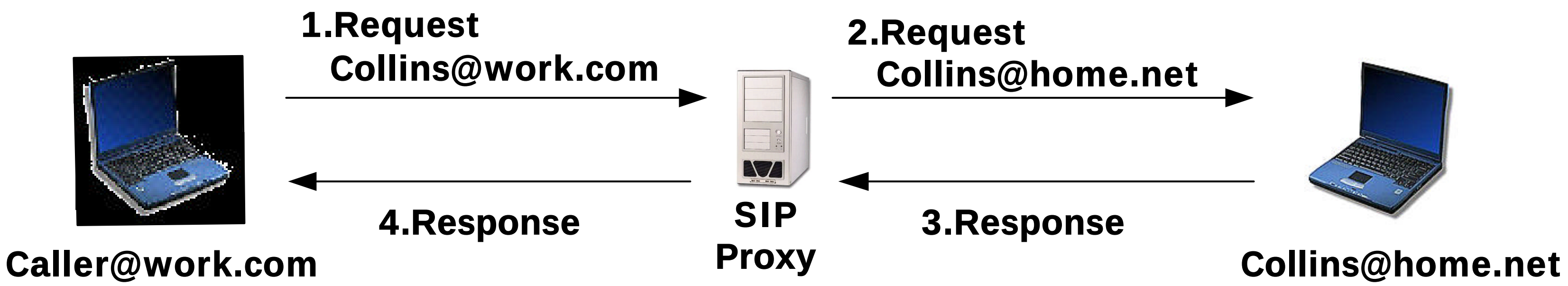


# SIP Network Entities [1/4]

- Clients
  - User agent clients
  - Application programs sending SIP requests
- Servers
  - Responds to clients' requests
- Clients and servers may be in the same platform.
  - Proxy acts as both clients and servers

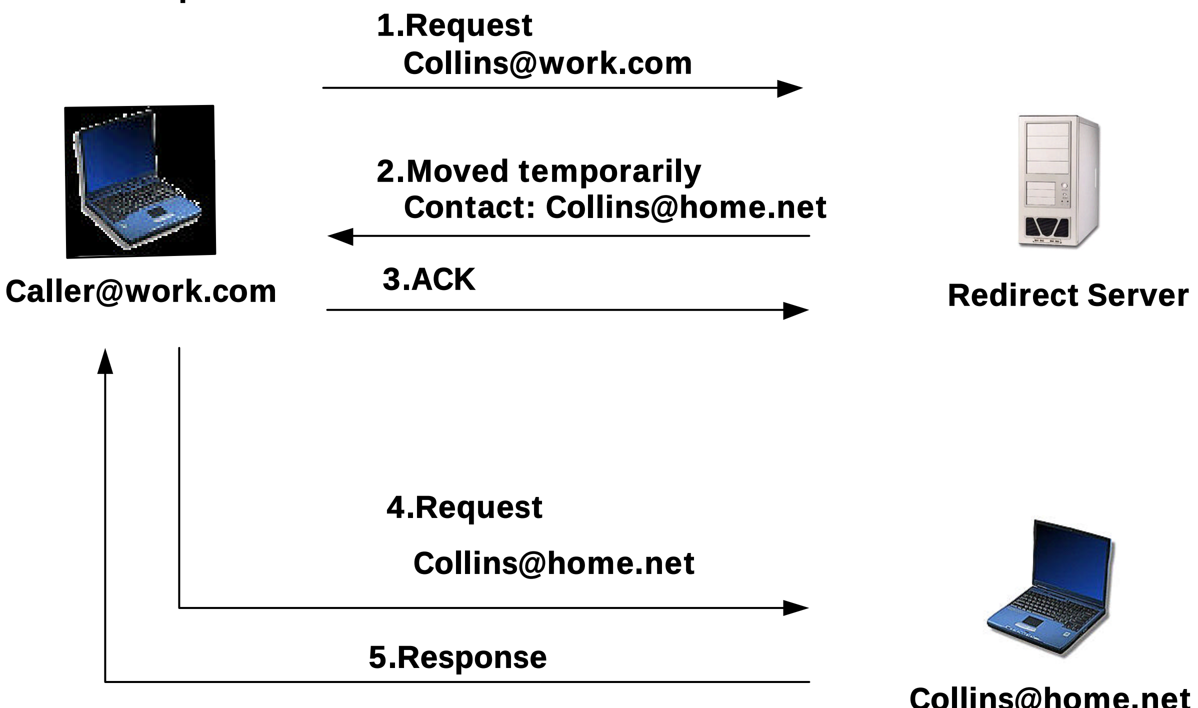
# SIP Network Entities [2/4]

- Four types of servers
  - Proxy servers
    - Act in a similar way to a proxy server used for web access
    - Handle requests or forward requests to other servers after some translation
    - Can be used for call forwarding, time-of-day routing, or follow-me services



# SIP Network Entities [3/4]

- Redirect servers
  - Accept SIP requests
  - Map the destination address to zero or more new addresses
  - Return the new address(es) to the originator of the request

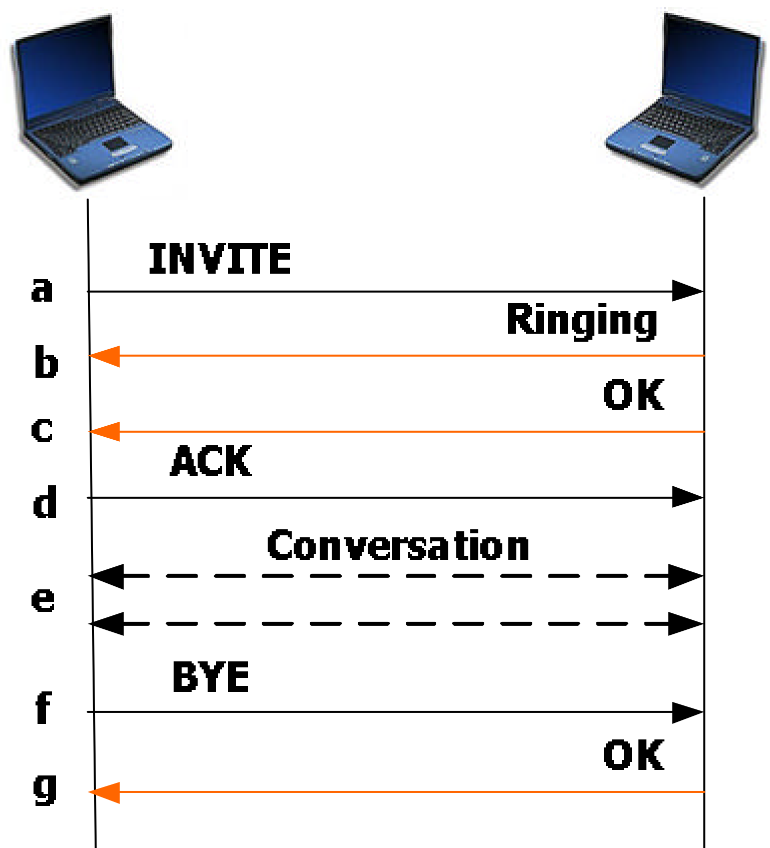


# SIP Network Entities [4/4]

- A user agent server
  - Accepts SIP requests and contacts the user
  - The user responds → an SIP response
  - A SIP device
    - E.g., a SIP-enabled telephone
- A registrar (location server)
  - Accepts SIP REGISTER requests
    - Indicating that the user is at a particular address
    - Personal mobility
  - Typically combined with a proxy or redirect server

# SIP Call Establishment

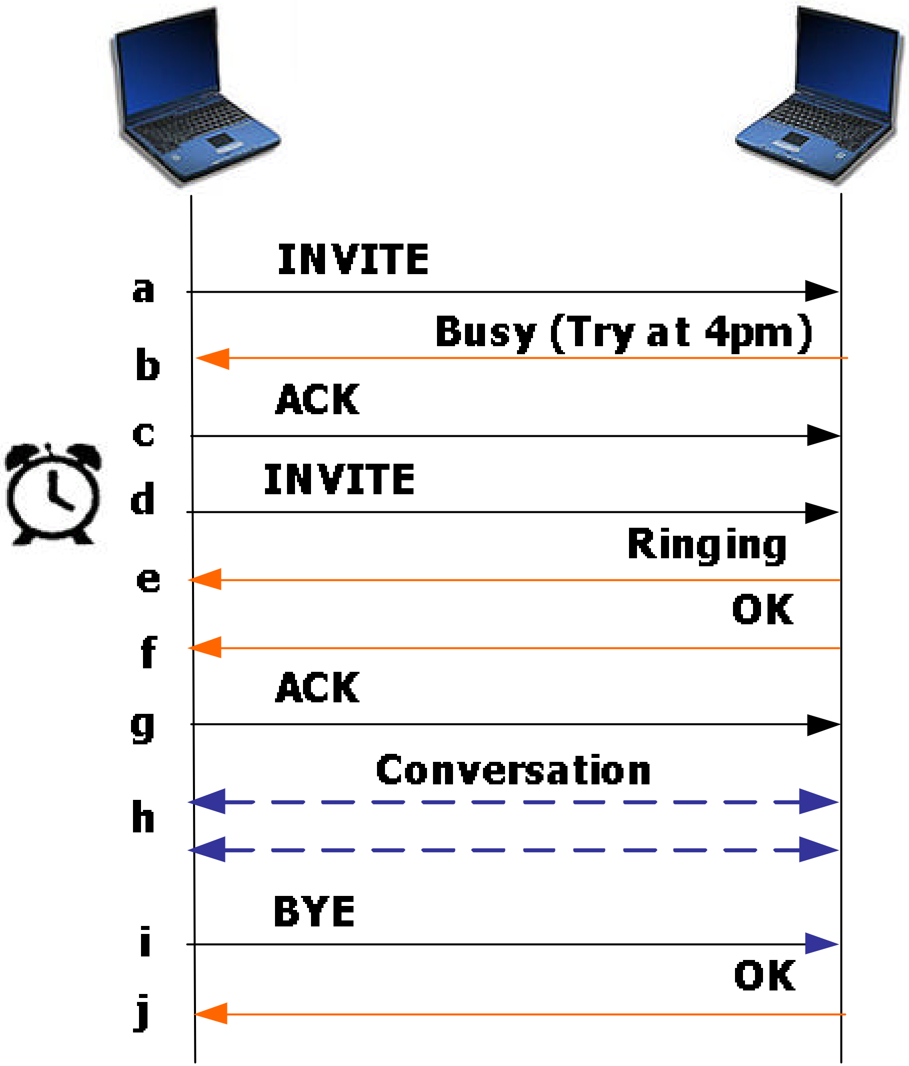
- A SIP call establishment is simple.
- A number of interim responses may be made to the INVITE prior to the called party accepting the call.



# 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
    - Text-based encoding
  - 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 Busy Subscriber Service



# 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

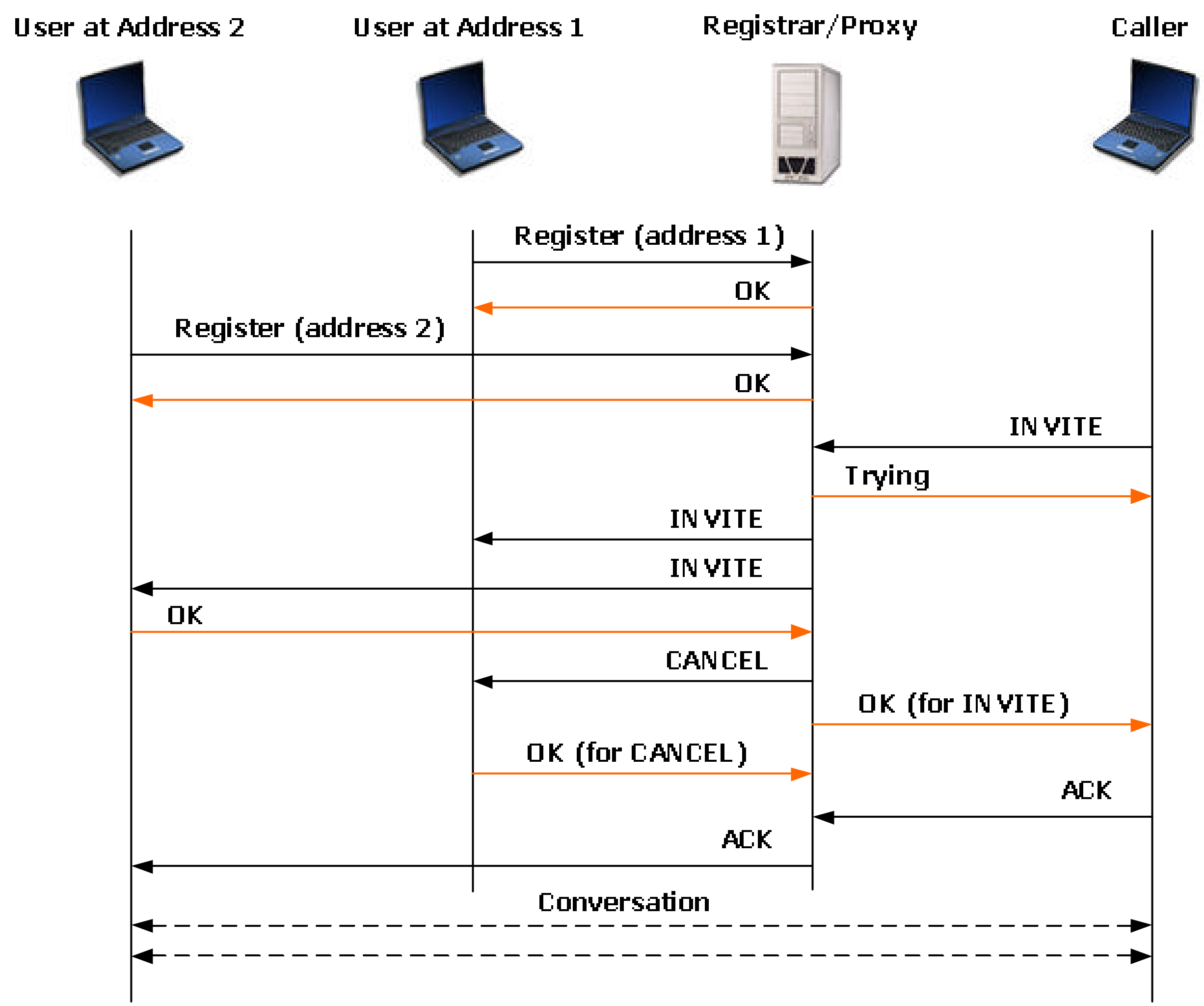
# SIP Requests [1/2]

- Method SP Request-URI SP SIP-version CRLF
- Request-URI
  - The SIP address of the destination
- Methods
  - INVITE, ACK, OPTIONS, BYE, CANCLE, 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
    - To support a particular type of media
- CANCEL
  - Terminate a pending request
  - Pending Request: 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

# “One Number” Service



# SIP INFO Method

- Specified in RFC 2976
  - For transferring information during an ongoing session
- The transfer of DTMF digits
- The transfer of account balance information
  - Pre-paid service
- The transfer of mid-call signaling information

# 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
  - sip:collins@home.net
  - sip:3344556789@telco.net

# Message Headers

- Provide further information about the message
- E.g.,
  - To:header in an INVITE
    - The called party
  - From:header
    - The calling party
- 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.

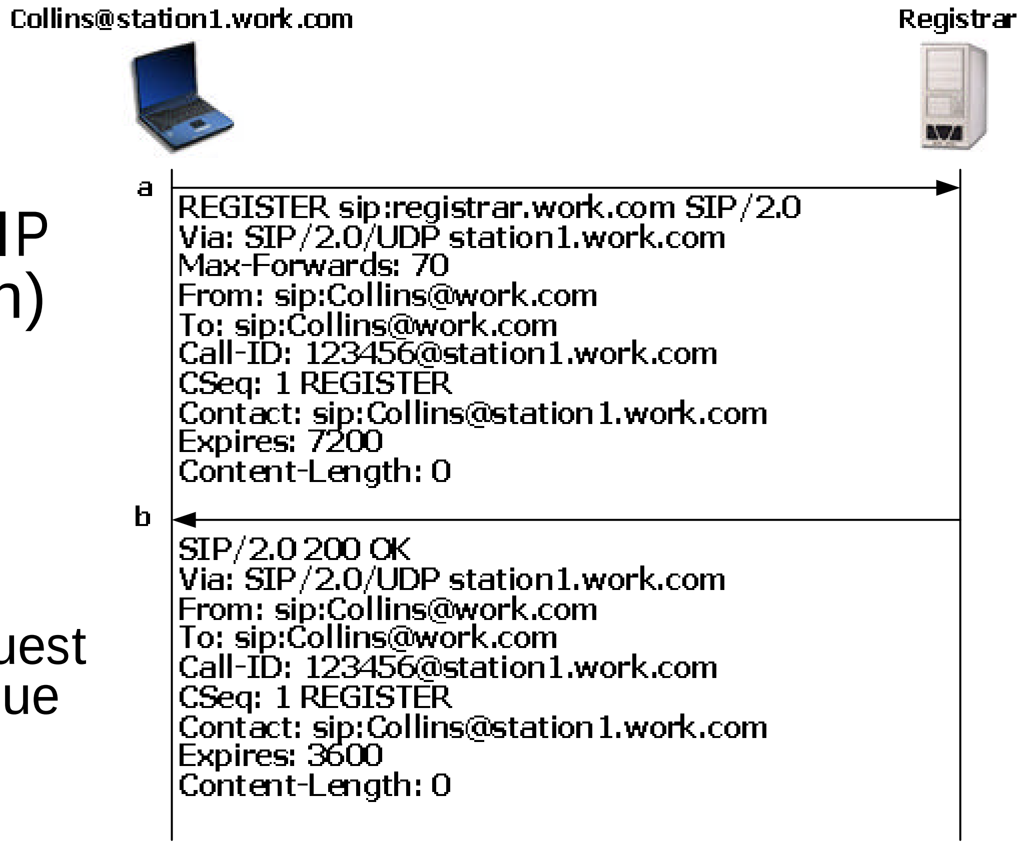
- Request Headers
  - Apply only to SIP requests
  - Addition information about the request or the client
  - E.g.,
    - Subject:
    - Priority: urgency of the request (emergency, urgent, normal, or non-urgent)
- Response Headers
  - Further information about the response that cannot be included in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

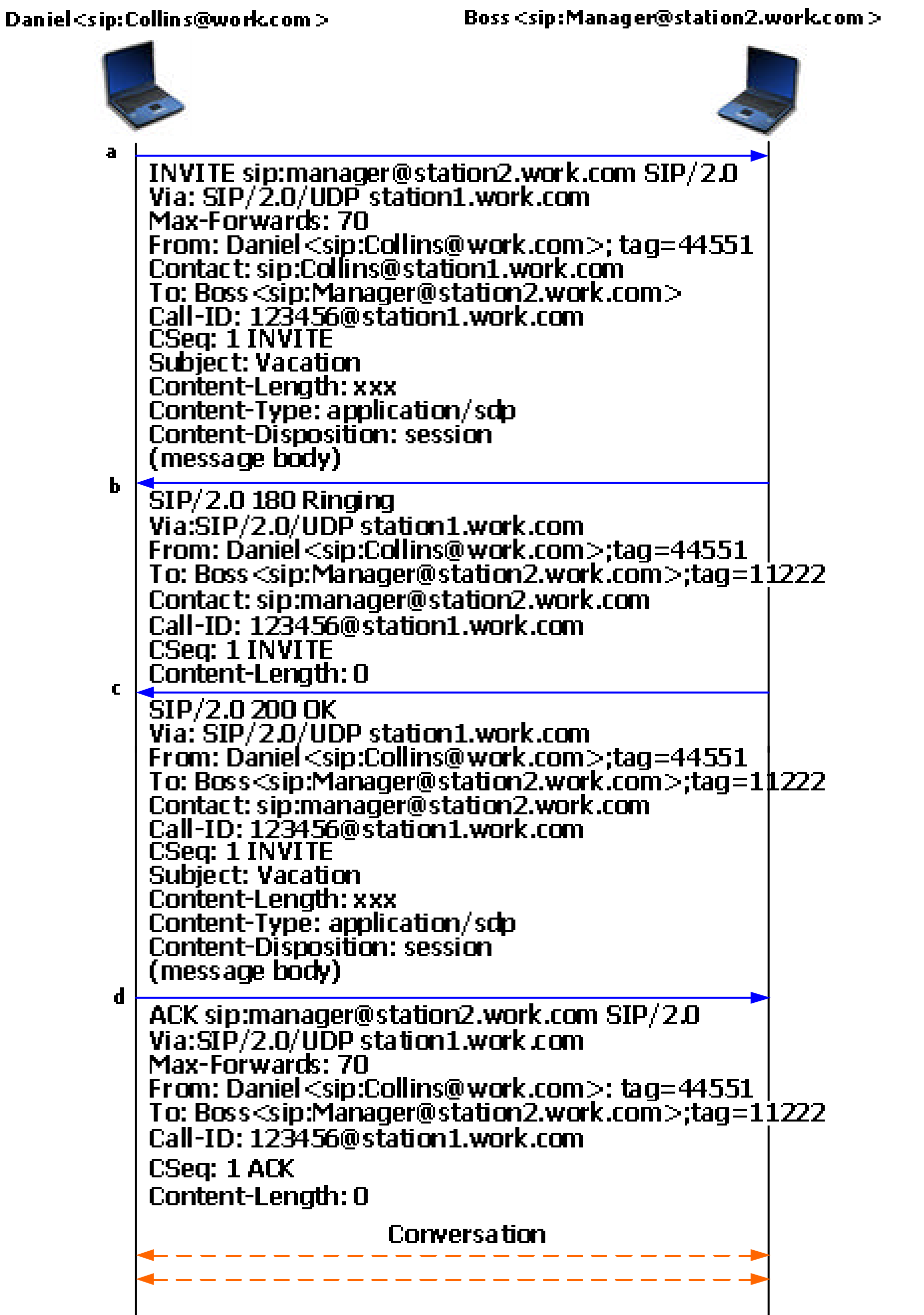
# Examples of SIP Message Sequences

- 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
  - 0, unreg



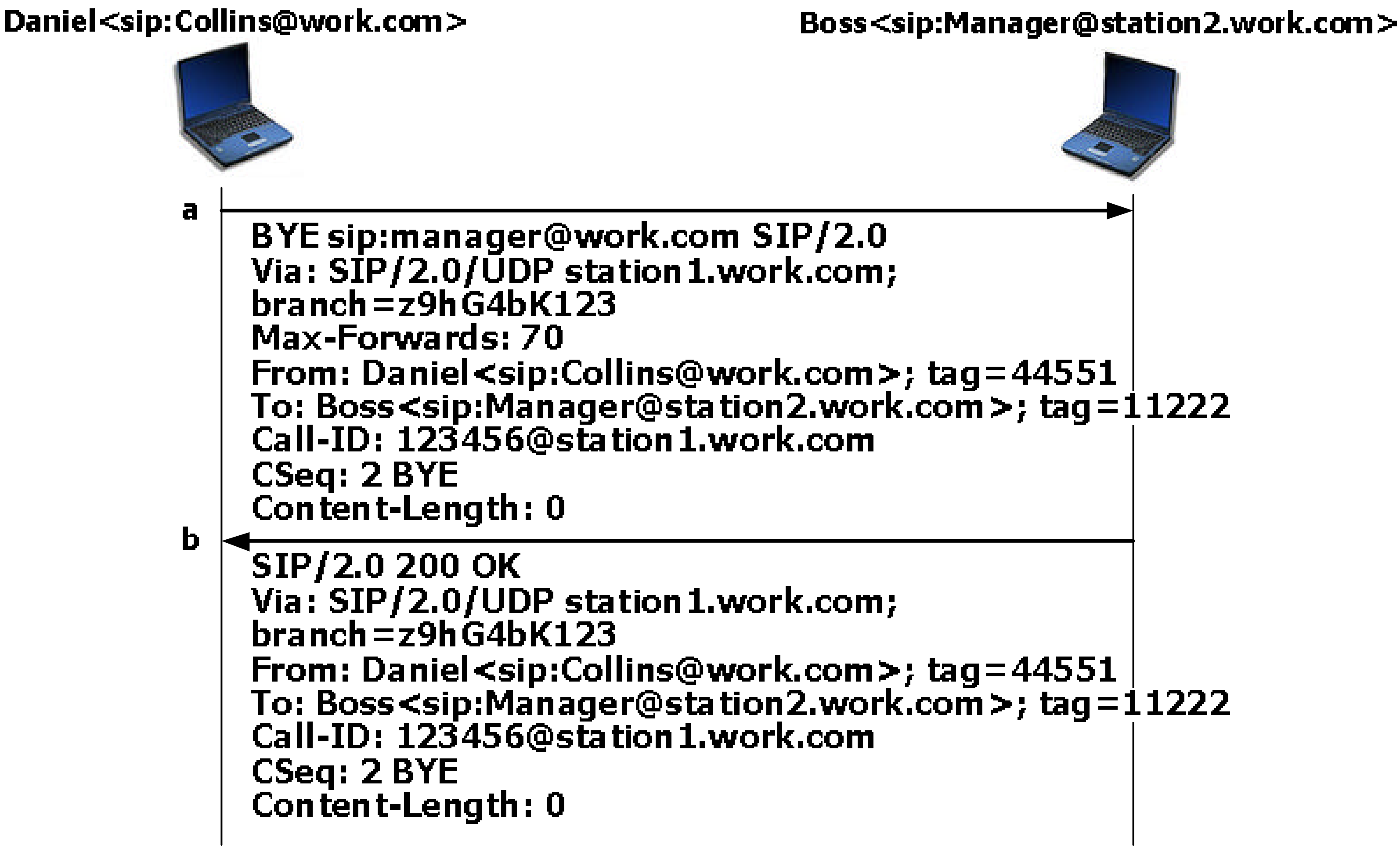
# Invitation

- 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



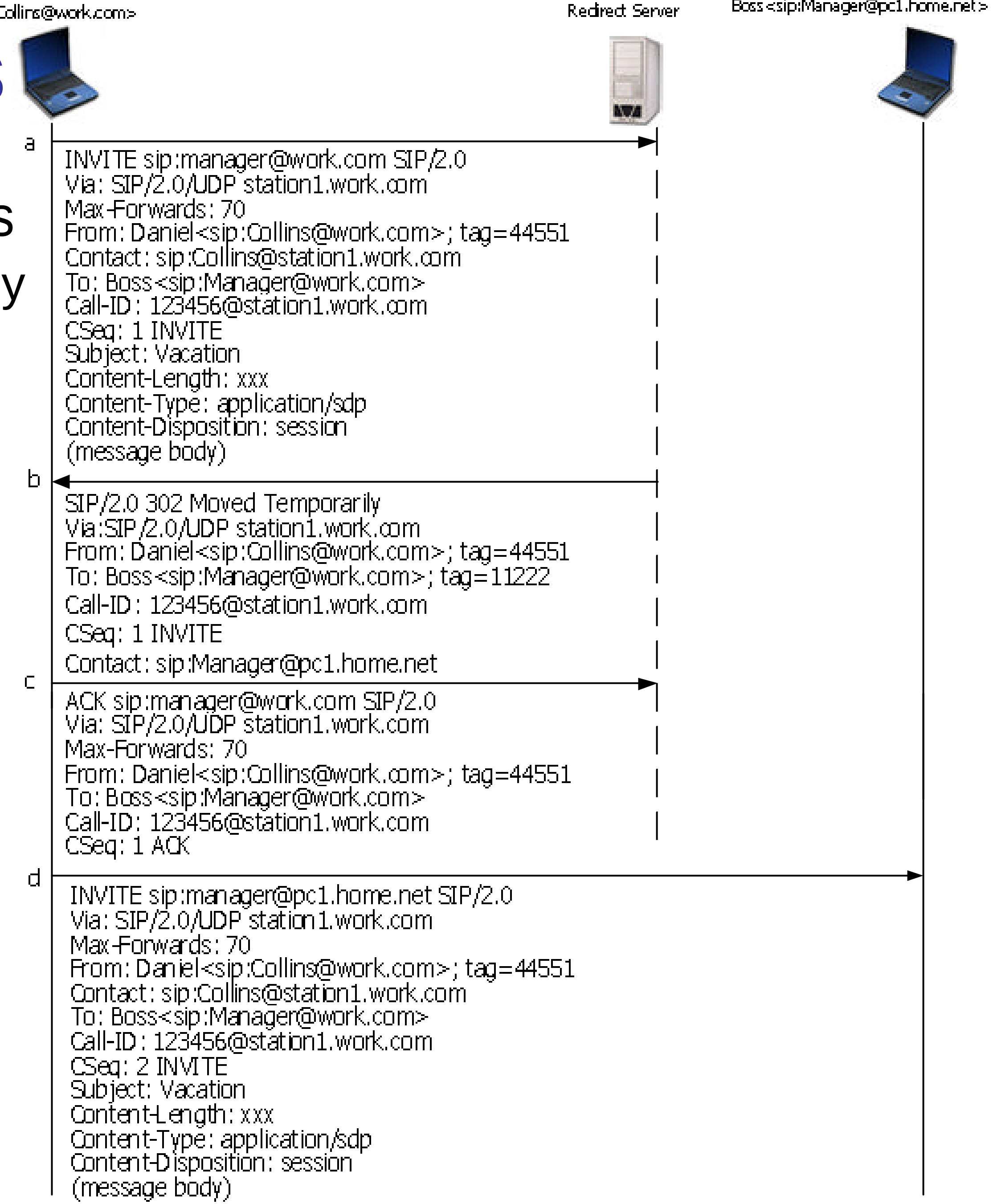
# Termination of a Call

- CSeq has changed.



# Redirect Servers

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



# Proxy Servers [1/2]

- 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 most common scenario will have at least two proxies: one at the caller and one at the callee end.
  - It is likely that only the last proxy in the chain changes the Request-URI.
  - The other proxies in the chain would simply use the domain part of the received Request-URI as input to a location function (e.g., DNS) to determine the next hop.

# Proxy Servers [2/2]

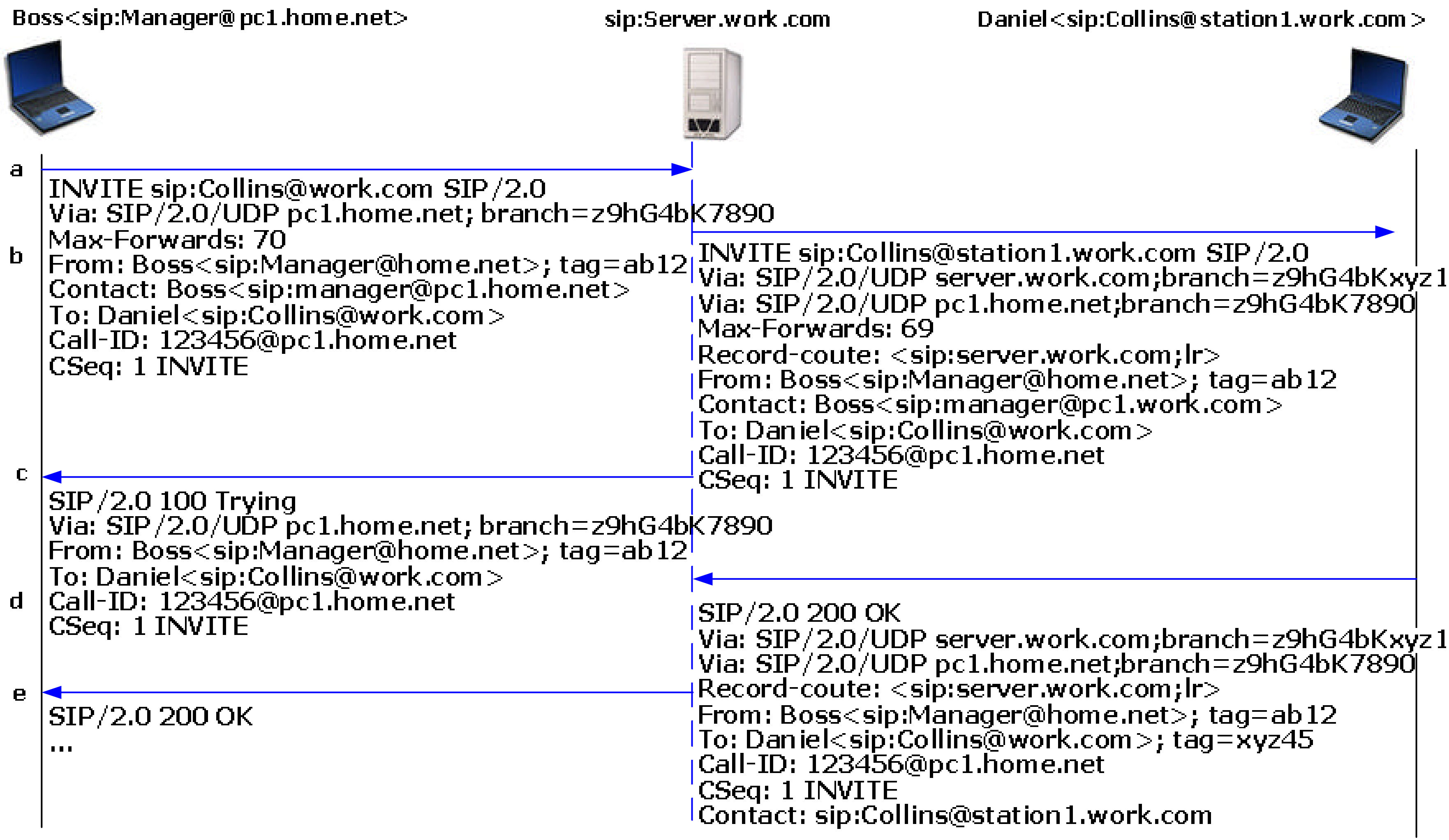
- Via:
  - The path taken by a request
  - Loop detected, 482 (status code)
  - For a response
    - The 1st Via: header is checked and removed.
    - The second Via: header is checked.
      - If it exists, perform forwarding.
      - If not, the response is destined to the proxy itself.
    - The response finds its way back to the originator of the request.
  - Branch: used to distinguish between multiple responses to the same request
    - Forking Proxy: Issue a single request to multiple destinations

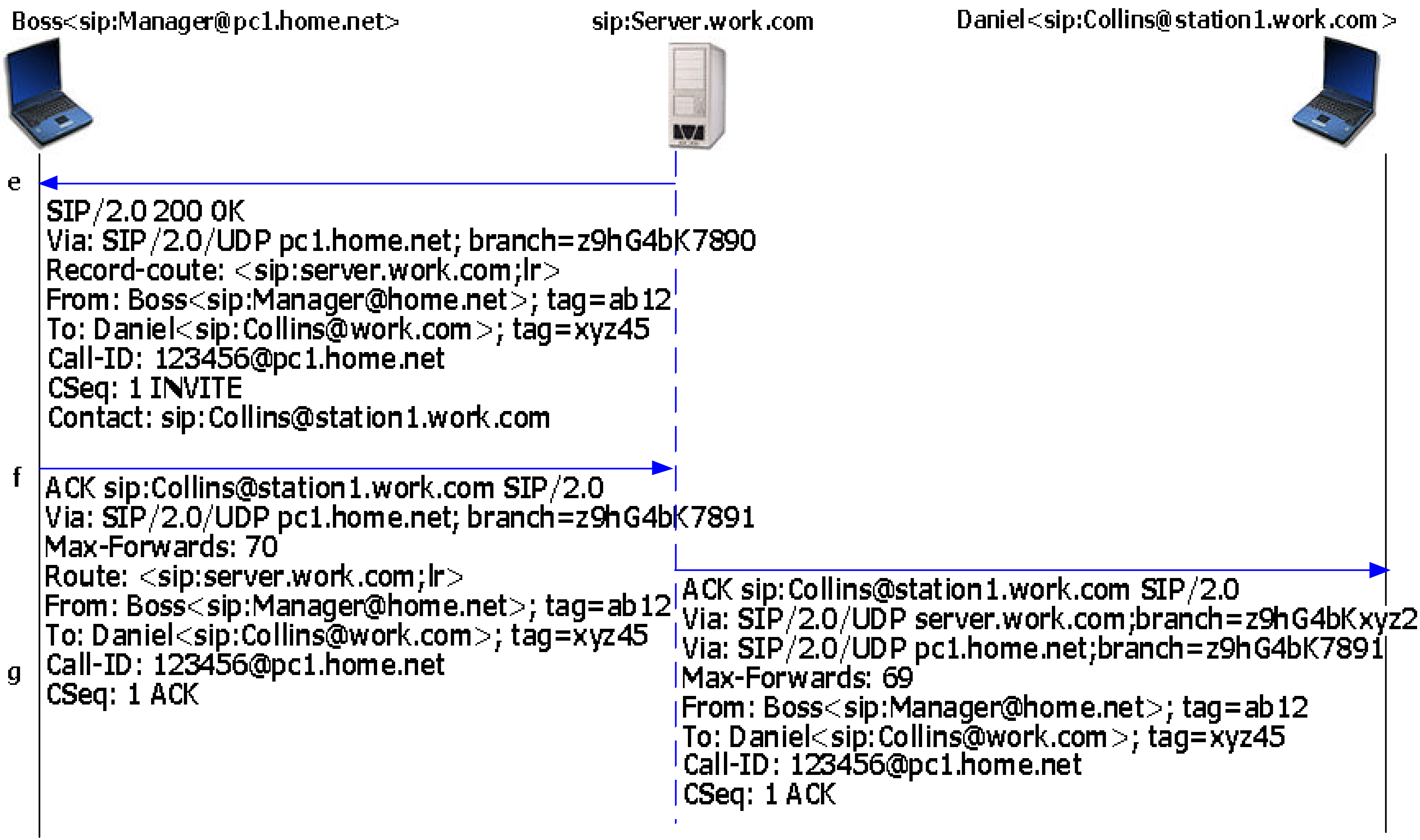
# Proxy State [1/2]

- Can be either stateless or stateful
- If stateless, the proxy takes an incoming request, performs whatever translation and forwards the corresponding outgoing request and forgets anything.
  - Retransmission takes the same path (no change on retransmission).
- If stateful, the proxy remembers incoming requests and corresponding outgoing request.
  - The proxy is able to act more intelligently on subsequent requests and responses related to the same session.

# Proxy State [2/2]

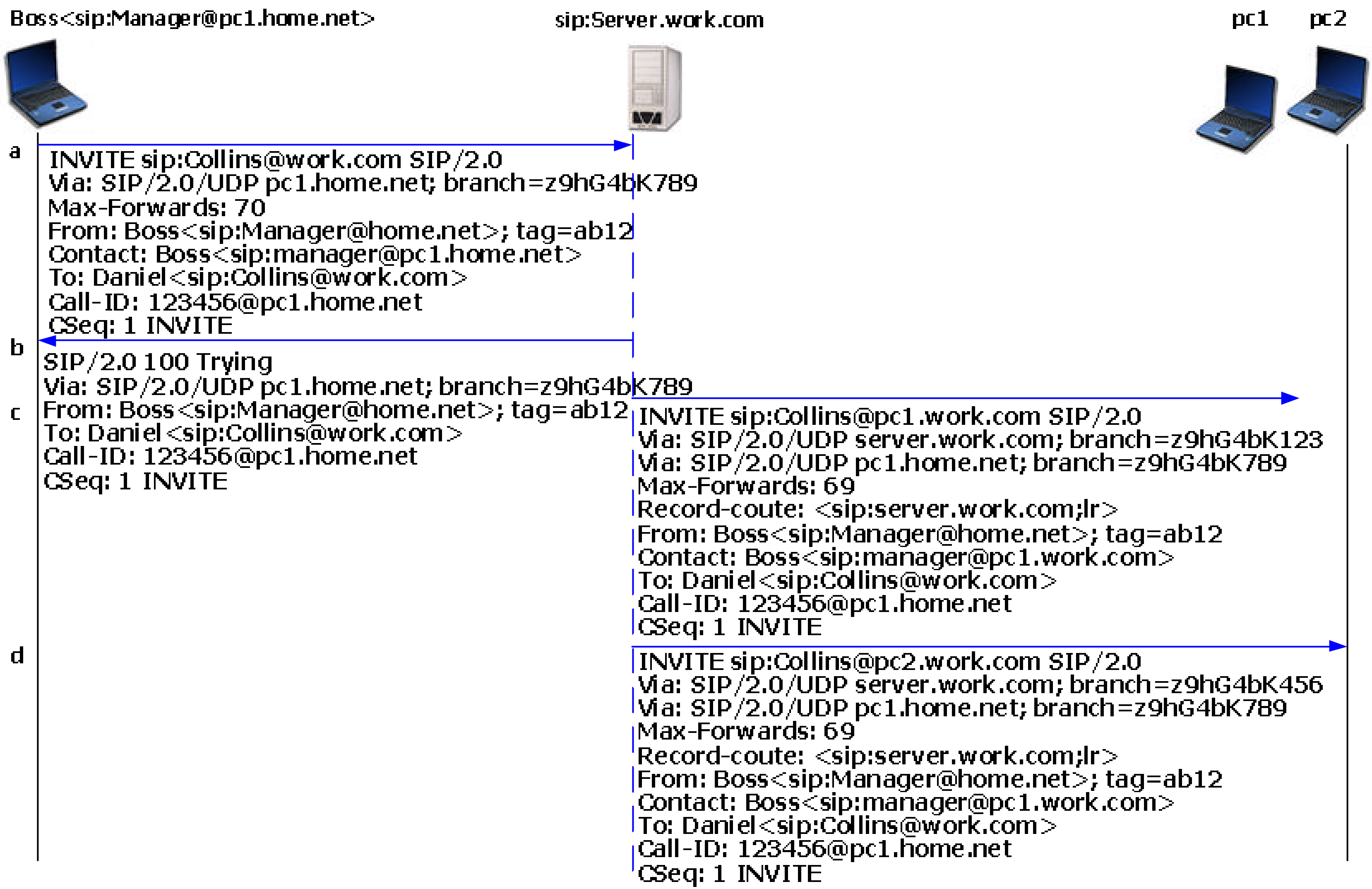
- Record-Route: and Route: Headers
  - The subsequent requests may not pass through the same path as the initial request/response.
    - E.g., use Contact:
  - A Proxy might require that it remains in the signaling path for all subsequent requests to provide some advanced service.
    - In particular for a stateful proxy
  - Insert its address into the Record-Route: header
  - The response includes the Record-Route: header
  - The information contained in the Record-Route: header is used in the subsequent requests related to the same call.
  - The Route: header is used to record the path that the request is enforced to pass.
  - lr (loose routing) vs. sr (strict routing)

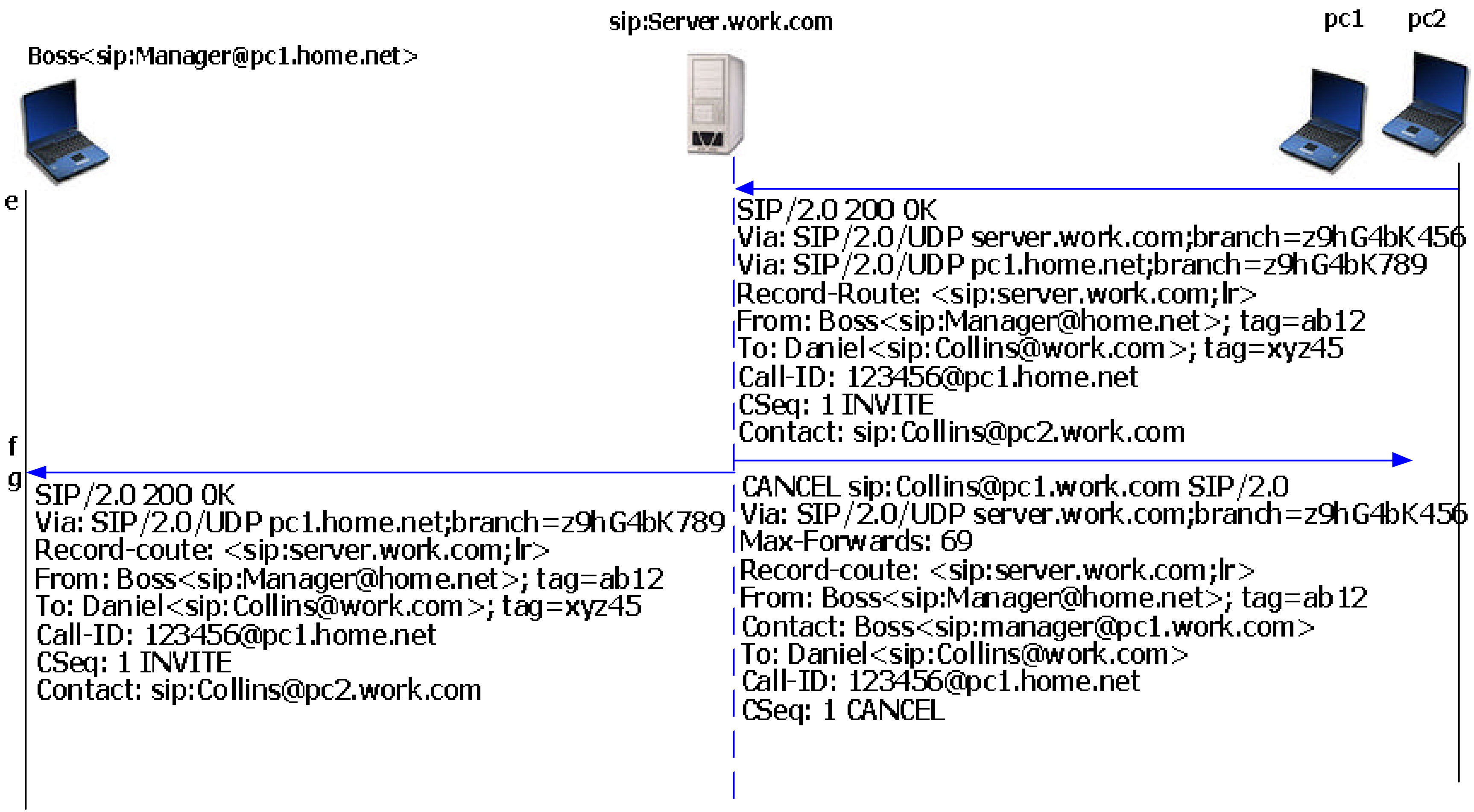

Boss<sip:Manager@pc1.home.net>
sip:Server.work.com
Daniel<sip:Collins@station1.work.com>
a
INVITE sip:Collins@work.com SIP/2.0
Via: SIP/2.0/UDP pc1.home.net; branch=z9hG4bK7890
Max-Forwards: 70
From: Boss<sip:Manager@home.net>; tag=ab12
Contact: Boss<sip:manager@pc1.home.net>
To: Daniel<sip:Collins@work.com>
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

# Forking Proxy

- A proxy can “fork” requests
- A user is registered at several locations
  - ;branch=xxx
- 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


sip:Server.work.com
Boss<sip:Manager@pc1.home.net>
pc1
pc2
e
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-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 INVITE
Contact: sip:Collins@pc2.work.com
f
g
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=z9hG4bK456
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