Session Initiation Protocol (SIP)

Introduction

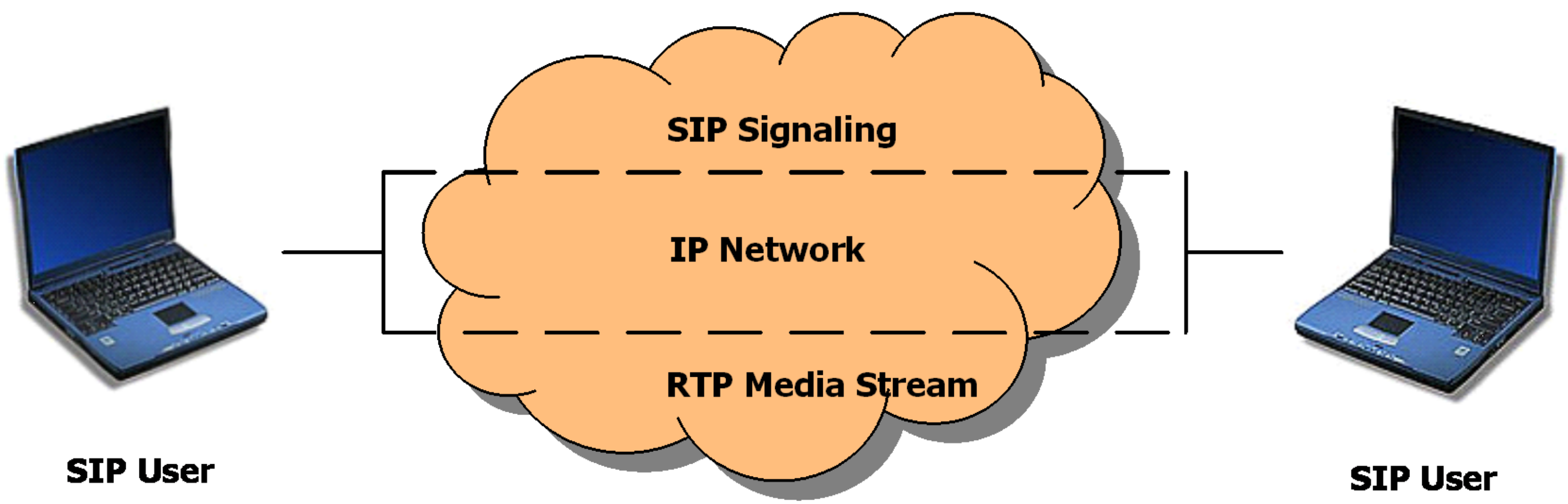
- A powerful alternative to H.323
- More flexible, simpler
- Easier to implement
  - Advanced features
- Better suited to the support of intelligent user devices
- A part of IETF multimedia data and control architecture
  - SDP, RTSP (Real-Time Streaming Protocol), SAP (Session Announcement Protocol)

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.

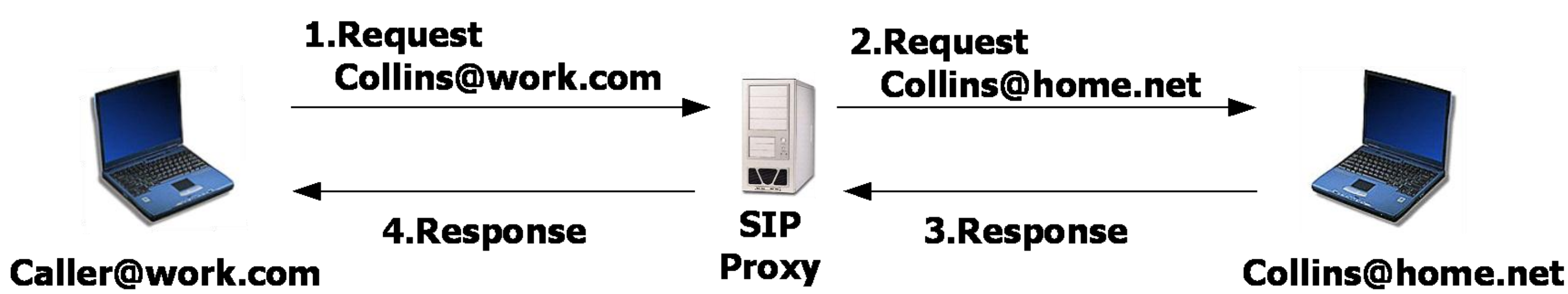


SIP Network Entities [1/4]

- 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

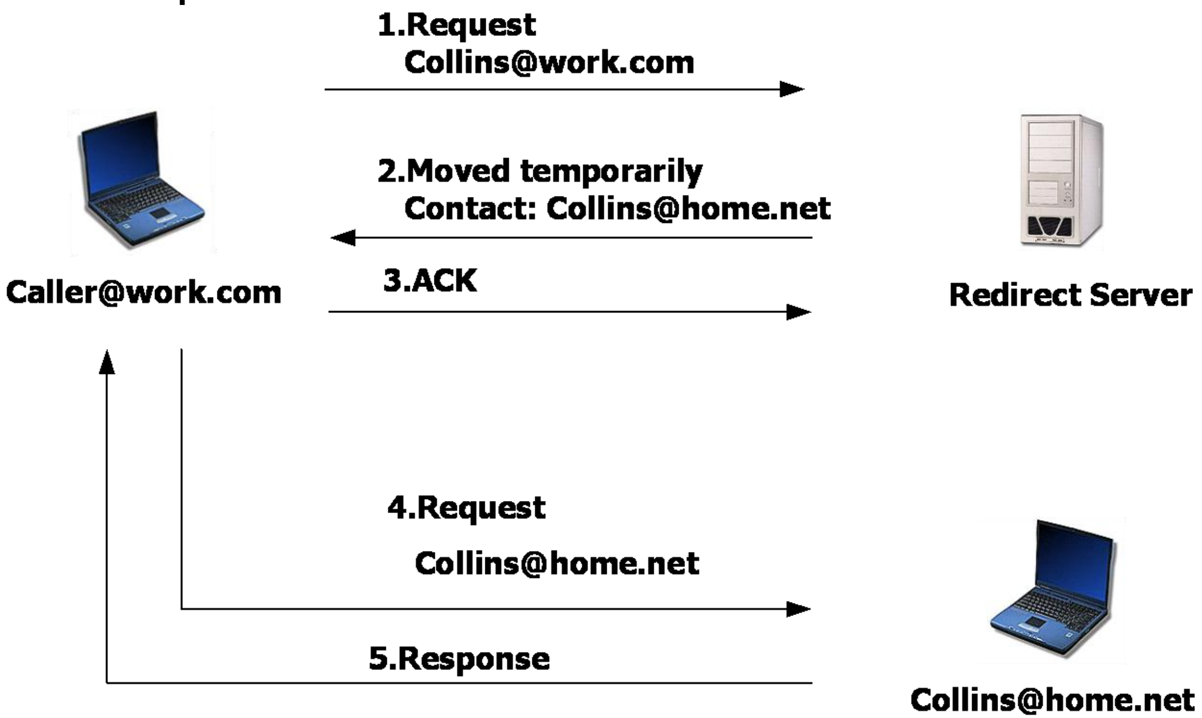
SIP Network Entities [2/4]

- 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

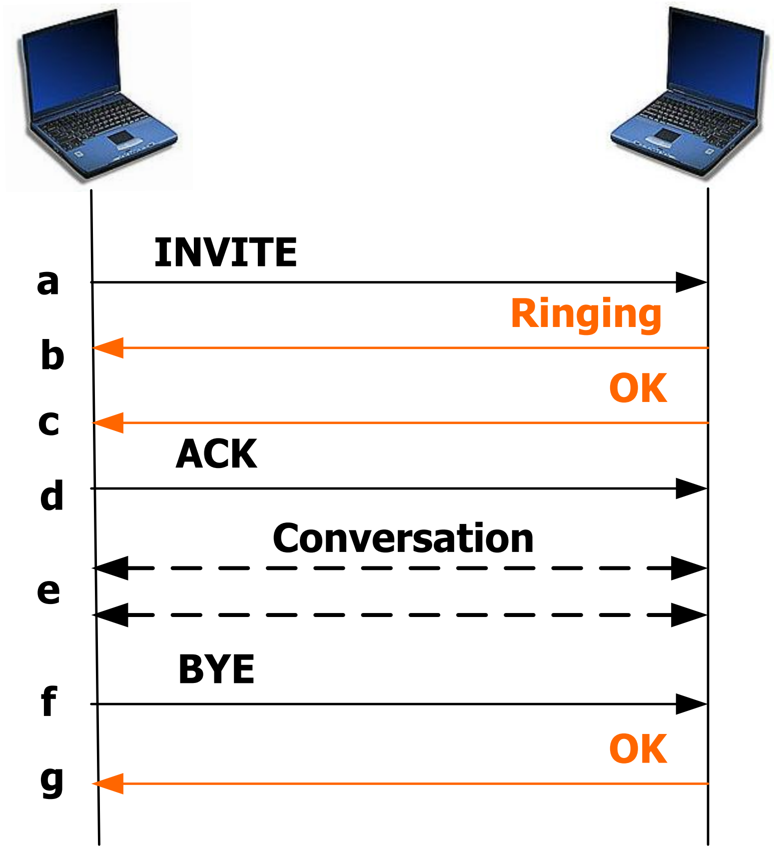


SIP Network Entities [4/4]

- 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

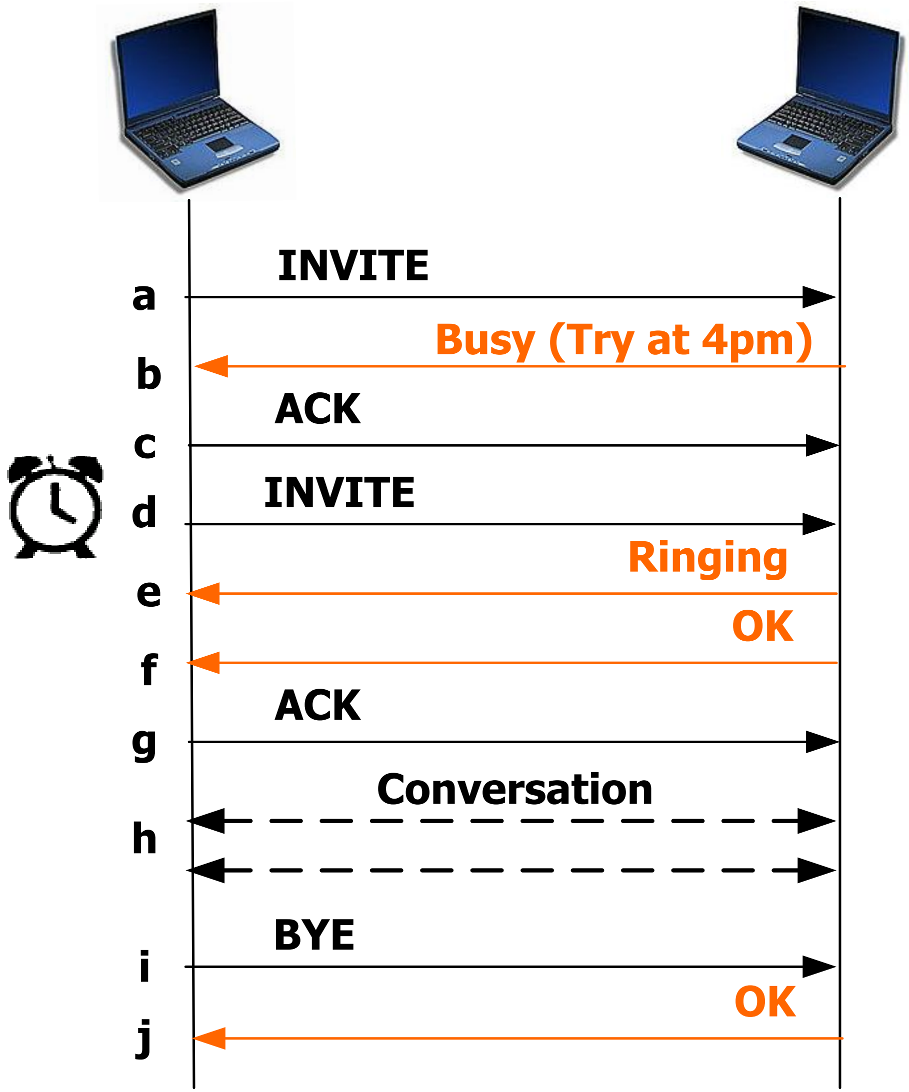
- A SIP call establishment is simple.
- A number of interim responses may be made to the INVITE prior to the called party accepting the call.



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

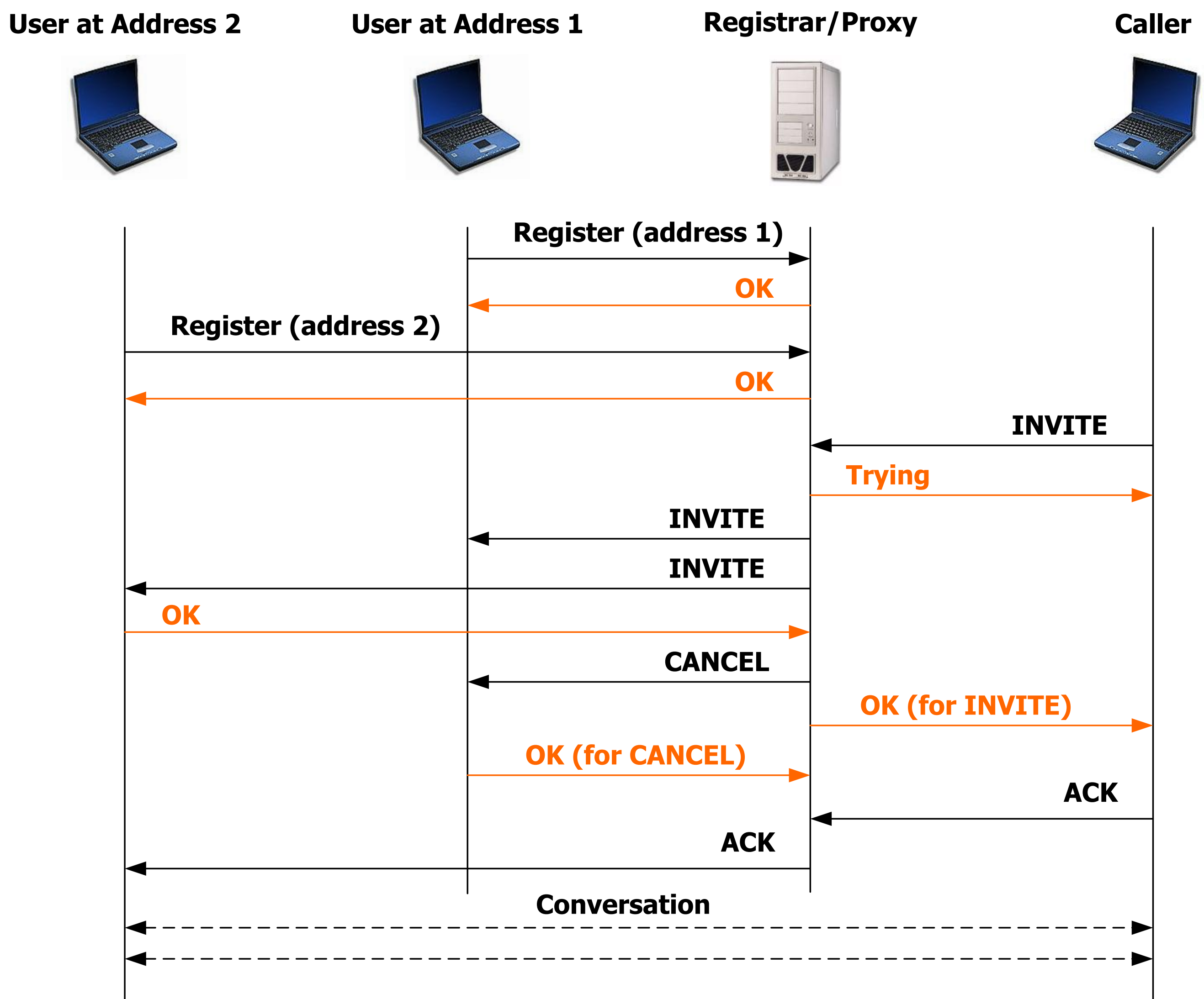
SIP Requests [1/2]

- 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
    - whether a called UA can support a particular type of media?
    - how a called UA would respond if sent an INVITE?
- 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 200 is defined: the request has been understood and has been performed)
  - 3XX Redirection (302: the called party is not available at the address used in the request, and the request should be reissued to a new address included with the response)
  - 4XX Request Failure (401: the client is not authorized to make the request)
  - 5XX Server Failure (505: the server does not support the SIP version specified in the request)
  - 6XX Global Failure (600: busy)
  - All responses, except for 1XX, are considered final
    - Should be ACKed if the original message happened to be an INVITE

SIP Addressing

- SIP URIs (Uniform Resource Indicators)
  - 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
  - M (mandatory), M* (the header field should be present in the request, but a receiver should be prepared to process the request even if the header is absent), O (optional), T (the header should be included in the request if a stream-based transport is used), C (the presence of the header depends on the context of the message), N/A (the header should not be sent in the request)
- 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: identify unsupported features in the server
    - 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, icon, alert, render …