Fundamentals of SIP

Spring VoiceCon 2007, Orlando
David A. Bryan
SIPeerior Technologies, Inc.
www.SIPeerior.com

Outline

- Background on VoIP
- What is SIP?
- SIP Components
- SIP Messages
- Locating Users and Establishing Sessions
- Routing in SIP
- Media with SIP
- SIMPLE
- NATs, Firewalls, and SIP
- P2PSIP
- References

Some (brief!) Background on VoIP

- Packet Switched, not Circuit switched
  - There is no dedicated path that traffic follows
  - Segmented into IP packets, sent to target, particular path not specified
- Original telephone network was circuit switched (think of the operator w/wires)
  - SS7, defined by the ITU (International Telecommunications Union)

VoIP Protocols

- H.323
  - ITU defined VoIP protocol
    - ITU is very formal, government oriented
  - More “telephony” in nature than Internet
- SIP
  - **IETF** (**I**nternet **E**ngineering **T**ask **F**orce) defined protocol for general multimedia session establishment
    - IETF is much less formal, individual oriented
  - More “Internet” in nature than telephony

H.323 vs. SIP

Feature	***H.323***	***SIP***
Encoding	Binary	Text
Control	Centralized server	Endpoints (using proxy)
Intelligence	Most in central server; core	Most in the endpoints; edge
Modularity	Monolithic (all in one server)	Can be multiple servers
Defined by	ITU	IETF
Media	RTP	RTP

SIP

- **S**ession **I**nitiation **P**rotocol
- Defined (mostly) in IETF RFCs 3261-3264
  - Many other drafts extend SIP and add new features
- Text based, very similar to HTTP
- Establishes a media session between endpoints
- Allows mobility – locate users using a SIP URI (sip:dbryan@sipeerior.com)
  - URI is not tied to a particular host
- Usually runs on port 5060, using UDP or TCP
- Allows for IPv4 and IPv6 (Next generation mobile phones use SIP over TCP/IPv6)

SIP

- Offers Event management (Subscribe/Notify) for presence etc.
- Extended by SIMPLE to support IM (Instant Messaging)
- Designed to allow easy interoperation with other Internet features such as WWW and email

Why is SIP Important?

- The promise of interoperable equipment
  - A Nortel phone talking to a Cisco proxy to reach another user with an Avaya phone
- Designed for the future
  - Any media (voice, video, text, ???)
  - Internet/endpoint centric design
  - Easily extensible
- Has become the dominant VoIP protocol

Major Components of SIP

- SIP systems specify components *logically*
- There are several components specified
  - *UA* (User-Agent, Endpoint) – the phone itself
  - *Registrar* – keeps track of where the user is within a system
  - *Redirect Server* – used to inform devices when they need to contact different locations
  - *Proxy Server* – used to relay messages back and forth within the system
- In practice, several *logical* functions may actually reside in the same *physical* server/program

User Agents

- A User Agent (also called a UA, Phone, or Endpoint) represents a user of a SIP system
  - A particular user may have more than one UA
  - Can be hard device (fixed or mobile) or a “soft client” – an application running on a PC

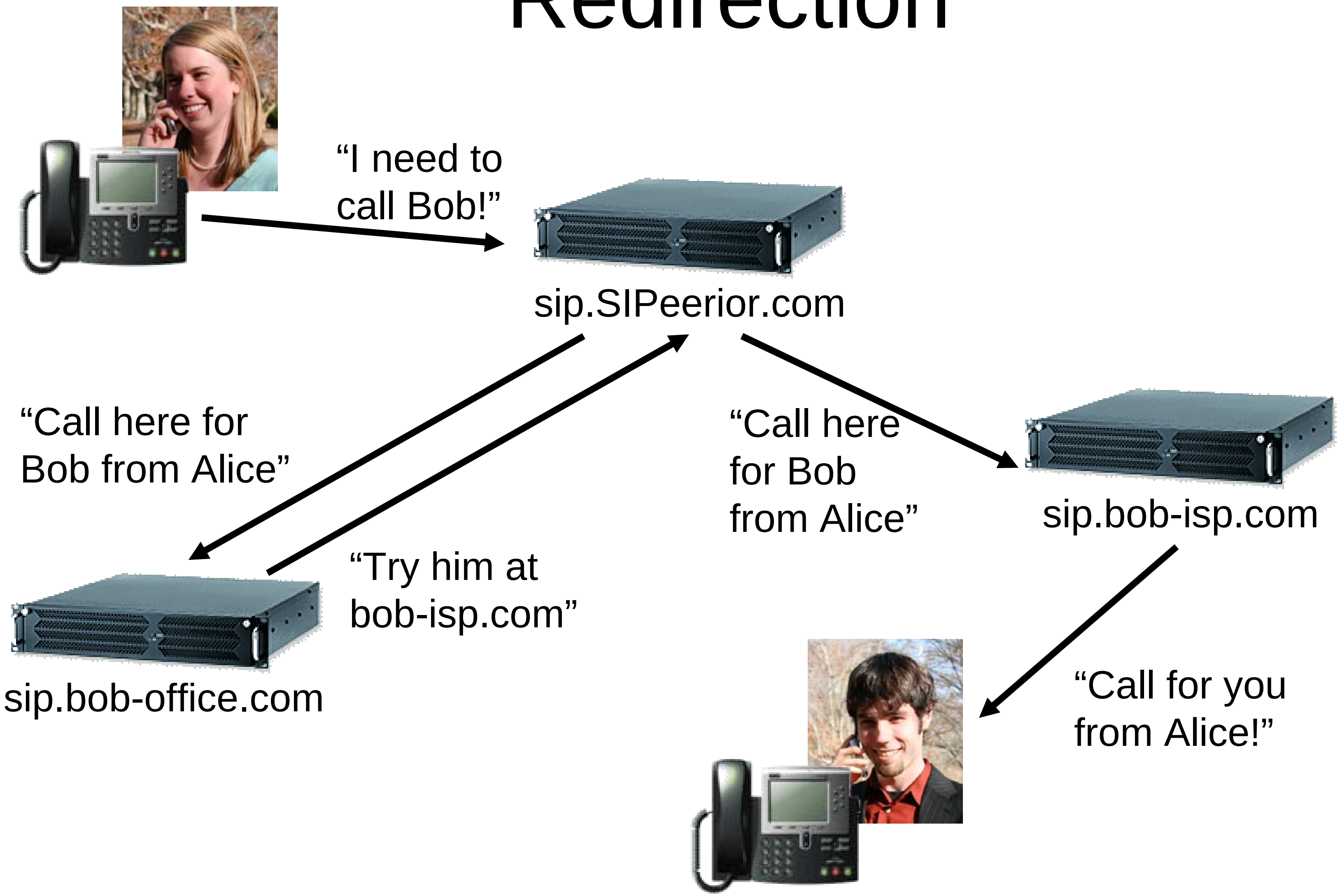
Hard Phone

Mobile Hard Phone

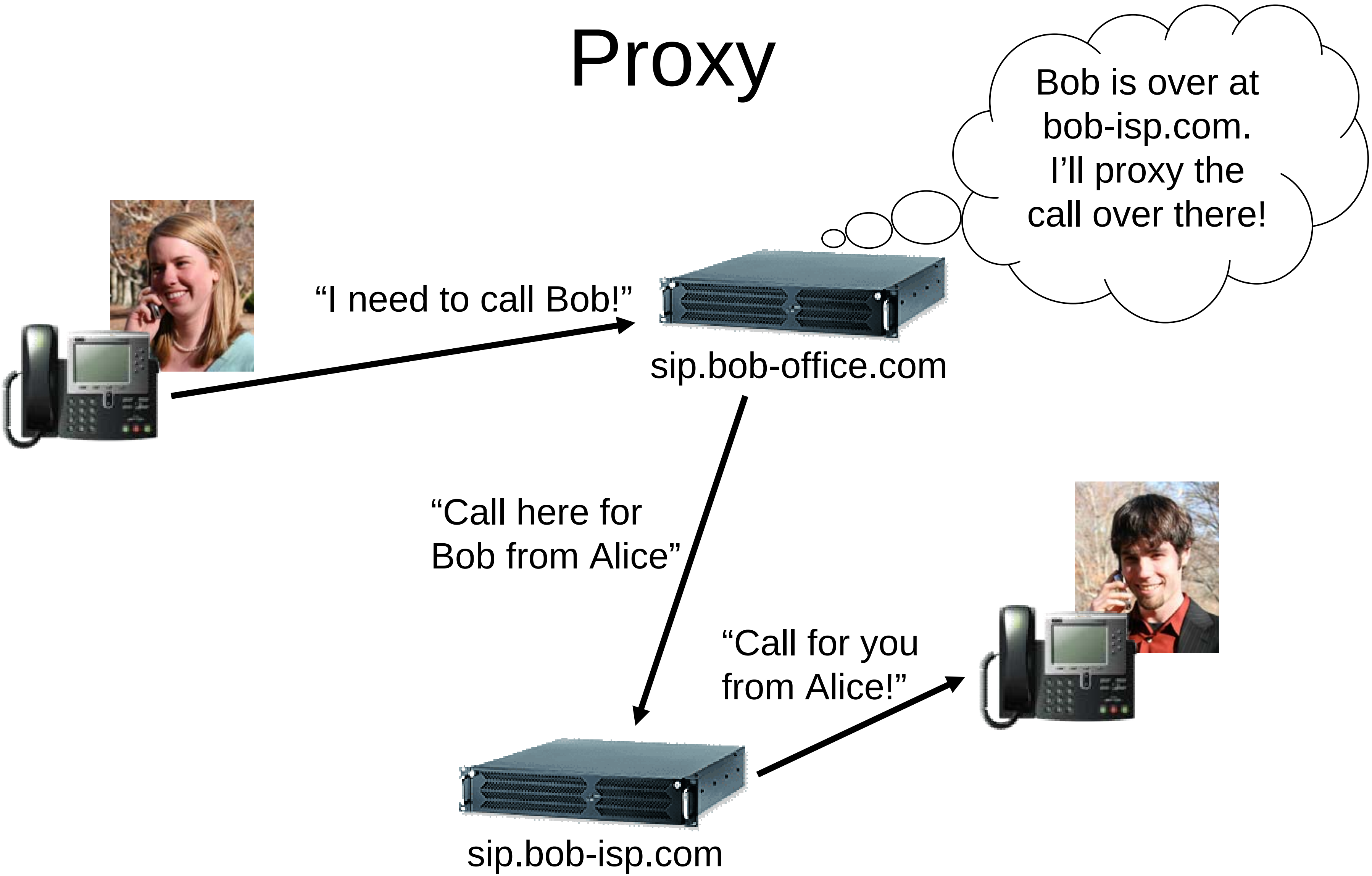
Soft Client

Registration

Redirection



Proxy


Bob is over at bob-isp.com. I'll proxy the call over there!
"I need to call Bob!"
sip.bob-office.com
"Call here for Bob from Alice"
"Call for you from Alice!"
sip.bob-isp.com

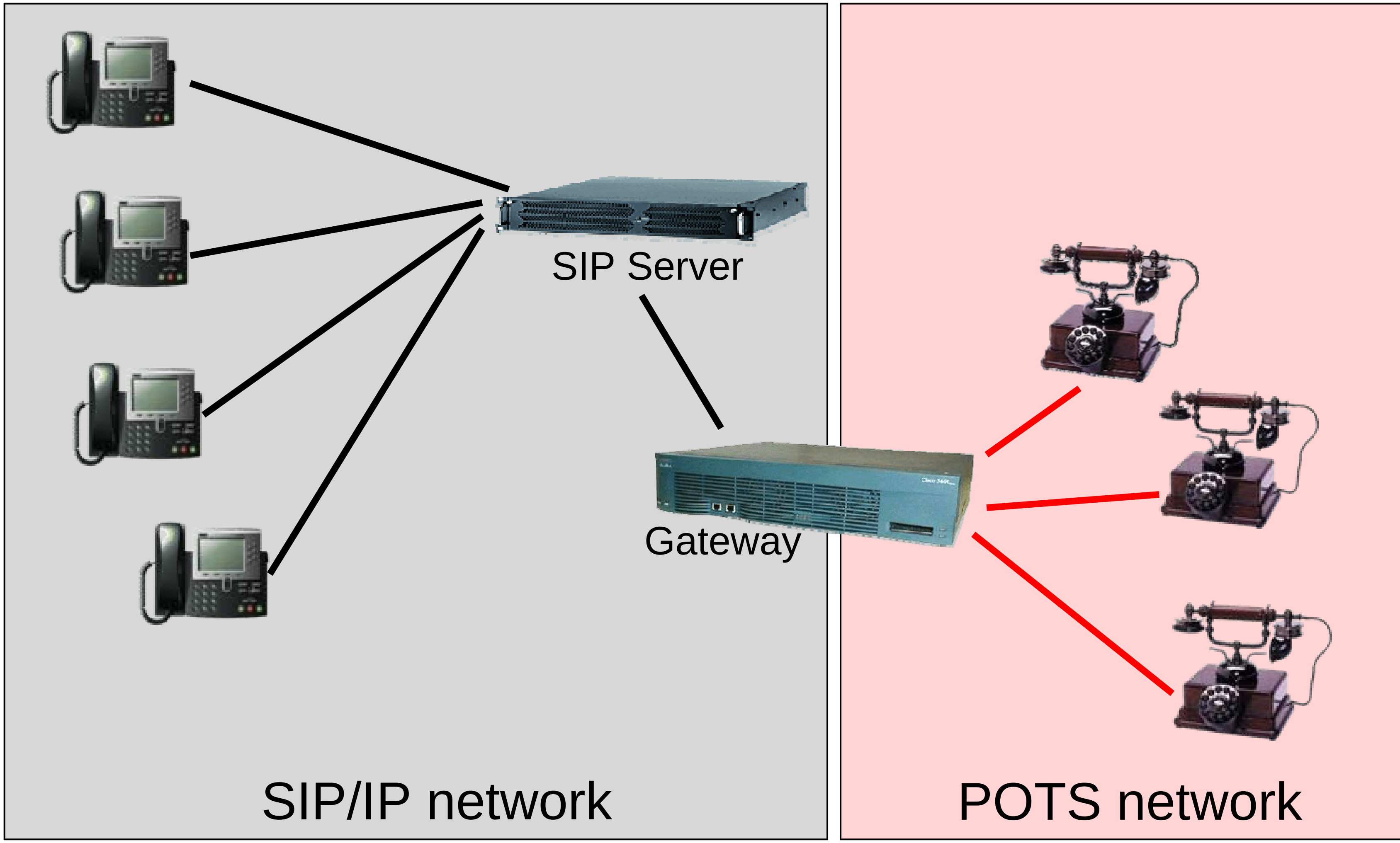
Mapping to Physical Entities

- SIP Proxy
  - Often what is sold as a proxy is really a Proxy/Registrar/Redirect server
- Many things are really built on UAs
  - Session Border Controllers are usually 2 UAs connected
    - B2BUA or “Back-to-Back User Agent”
  - Voice mail servers
  - Interactive Voice Response systems
- Softswitch often incorporates *all* of this

Gateways

- How about connecting to the outside world?
  - If you want to talk to another SIP user, your proxy may know how to contact that proxy directly
  - To call out (or to have calls come in) you use a *gateway*. It connects to one or more regular phone lines to let you receive and place calls to old fashioned “Plain Old Telephone Service”, or POTS

Gateway example



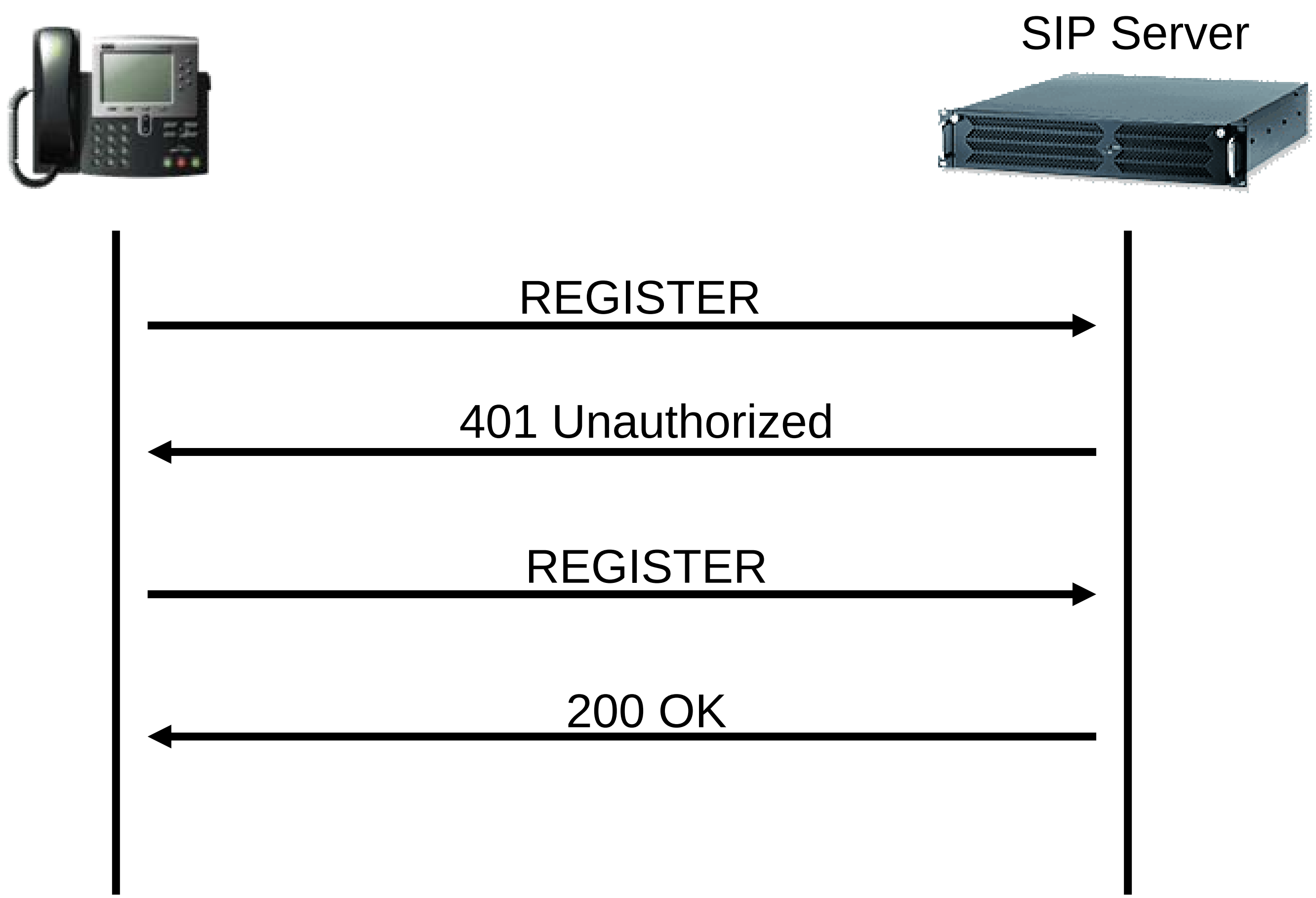
SIP Messages

- SIP messages fall into two types – *requests* and *responses*
  - Requests include a method telling what it is they are doing.
  - REGISTER, to register where to contact a user
  - INVITE, to start a new session (call)
  - BYE, to end a session that is established
  - CANCEL, to end a session that is still being set up (no final response yet)
  - ACK, part of the triple handshake
  - Many others

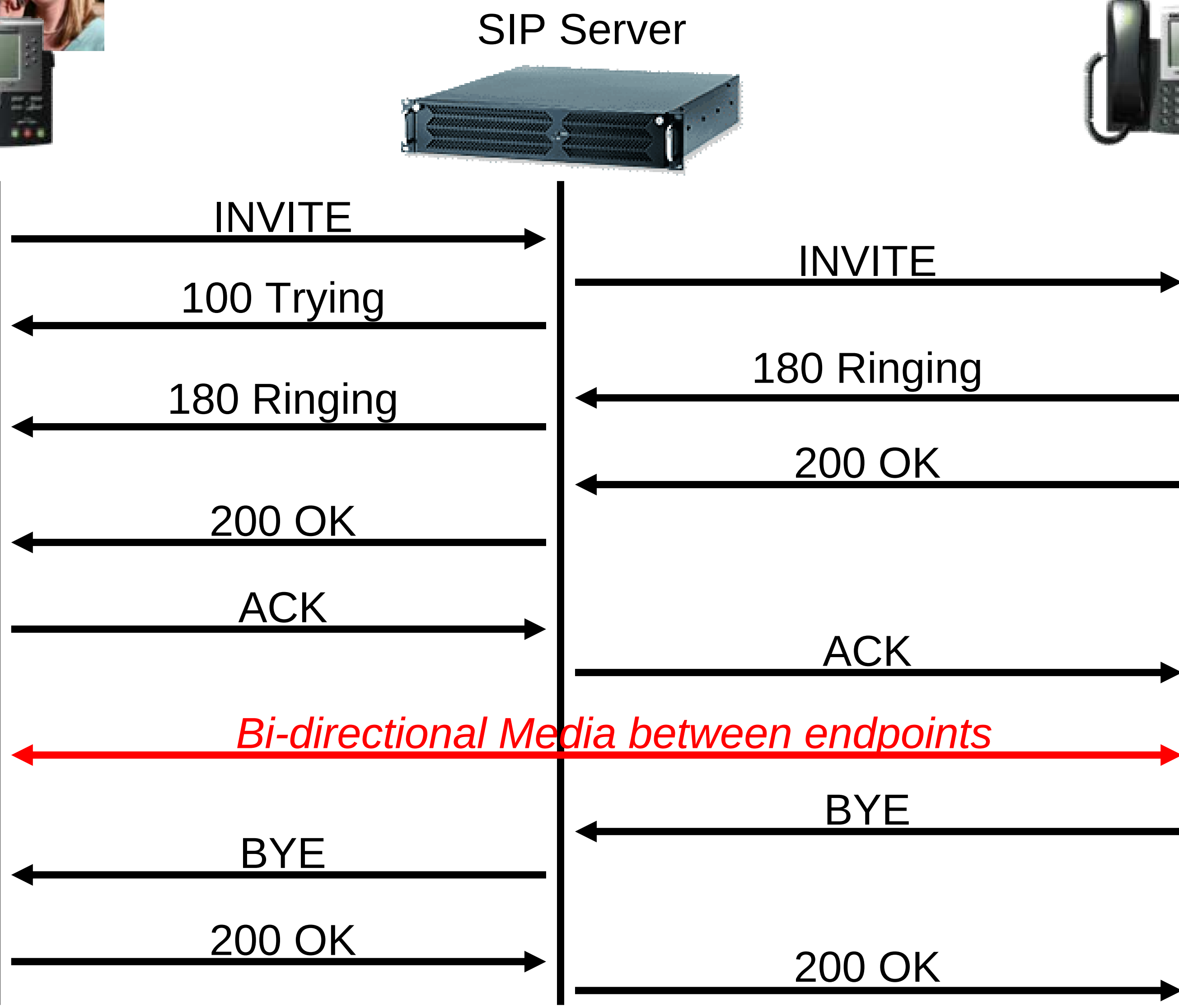
SIP Messages

- Responses are numeric, much like HTTP
  - 1xx Provisional – 100 Trying, 180 Ringing
  - 2xx Successful – 200 OK
  - 3xx Redirection – 301 Moved Permanently, 302 Moved Temporarily
  - 4xx Failure – 404 Not Found, 410 Gone, 403 Forbidden
  - 5xx Server Failure – 503 Service Unavailable
  - 6xx Global Failure – 600 Busy Everywhere

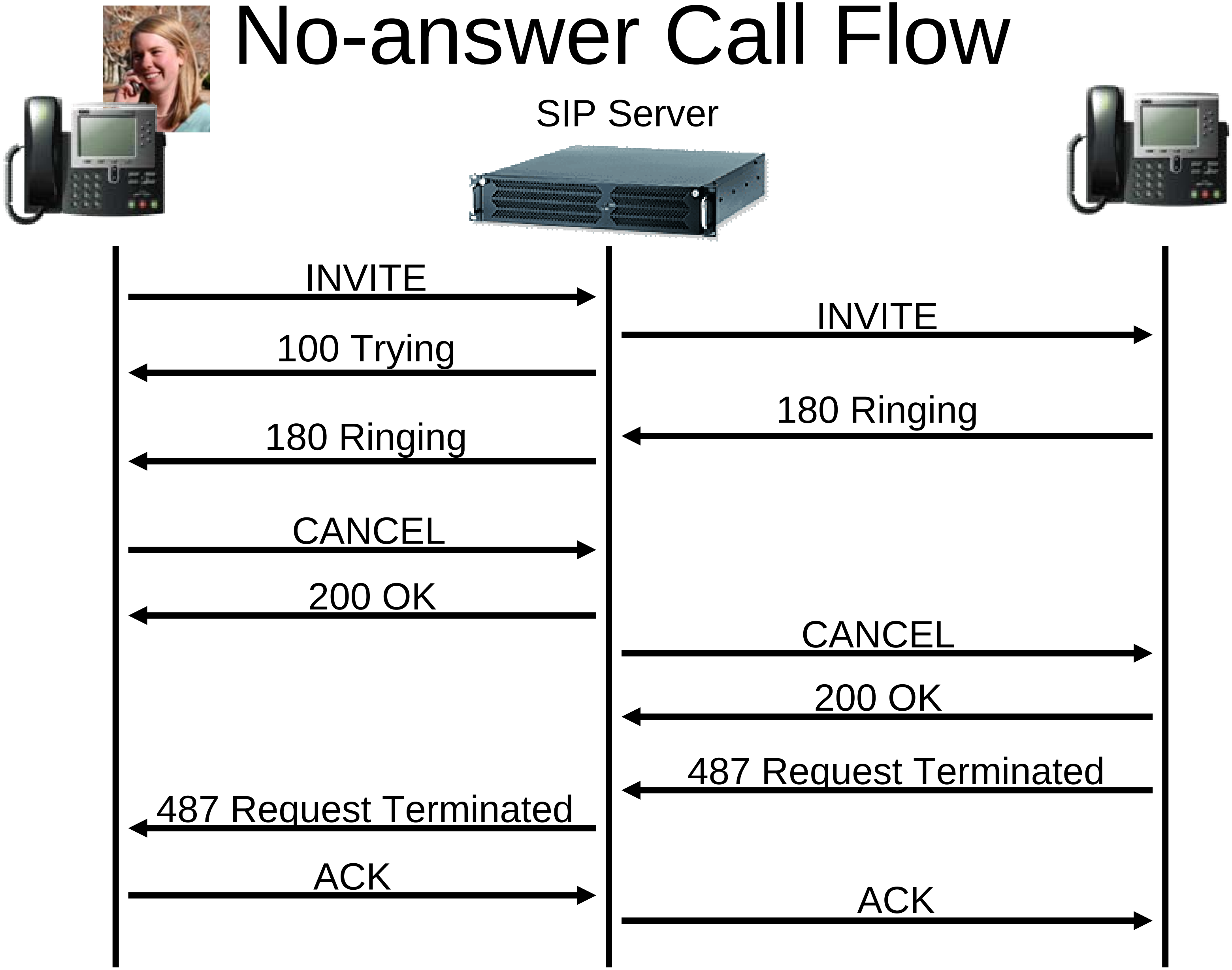
Registration Call Flow



Basic Call Flow



No-answer Call Flow



AORs and Contacts

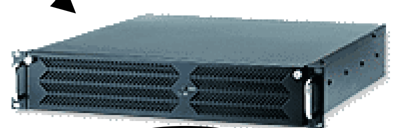
- In SIP, a registrar maps between an AoR and a contact
  - AoR: **A**ddress **o**f **R**ecord
    - Permanent, something on your business card
    - Not tied to a specific host
    - sip:dbryan@SIPeerior.com
  - Contact
    - Ephemeral, tied to a host
    - sip:dbryan@ua32.sipeerior.com
  - Being able to change AoR→Contact mapping allows mobility

Registration, revisited

192.168.0.1

"I'm Alice (alice@phonecompany.com), and I'll be using a phone at 192.168.0.1 today!"

Registrar



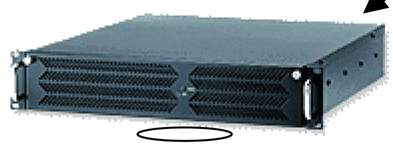
AoR → Contact Map:
alice@phonecompany.com → alice@192.168.0.1

Registration, revisited

“Lunch time! Mobile Time!
(alice@phonecompany.com)
will be using 10.0.1.1 now!”

10.0.1.1

Registrar



AoR → Contact Map:
alice@phonecompany.com → alice@10.0.1.1

INVITE message

INVITE sip:bob@bigcompany.com SIP/2.0
Via: SIP/2.0/TCP aliceua.phonecompany.com:5060
From: Alice <sip:alice@phonecompany.com>;tag=ss95cbav
To: Bob <sip:bob@bigcompany.com>
Call-ID: 3848276298220188511@aliceua.phonecompany.com
CSeq: 1 INVITE
Max-Forwards: 10
Contact: <sip:alice@aliceua.phonecompany.com;transport=tcp>
…snip!...

(Alice is going to call Bob…)

INVITE message

INVITE sip:bob@bigcompany.com SIP/2.0

SIP request, method is INVITE, target is an AoR (for Bob)

Via: SIP/2.0/TCP aliceua.phonecompany.com:5060

Via header tracks where it has been (more on this later)

From: Alice <sip:alice@phonecompany.com>; tag=ss95cbav

Message is from Alice, and this is the SIP URL for Alice. The tag is unique to this call and selected by Alice's UA

To: Bob <sip:bob@bigcompany.com>

Alice is trying to contact Bob at his SIP URL

Call-ID: 3848276298220188511@aliceua.phonecompany.com

Call-ID is a unique ID to track this particular call (selected by Alice's UA)

CSeq: 1 INVITE

Used to track which responses go with which messages. Responses to this invite will also have a CSeq of 1 INVITE

Max-Forwards: 10

Used in routing (more on this later)

Contact: <sip:alice@aliceua.phonecompany.com;transport=tcp>

How to contact Alice directly. Notice this is a contact (has a host)

…snip…

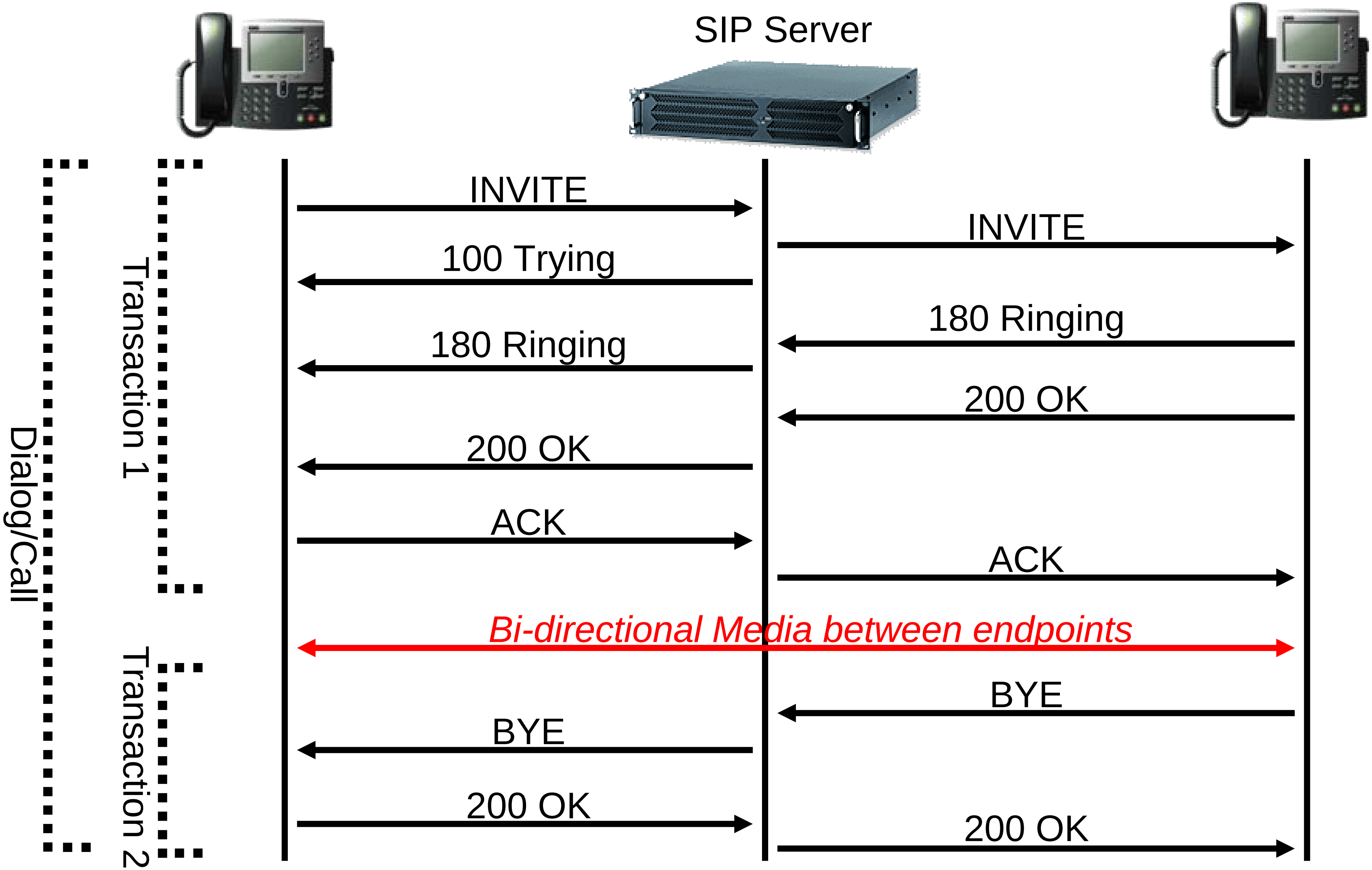
Calls and Dialogs

- A *call* is an informal term for communication between two devices
- A *dialog* is a more specific term, which defines a relationship between two UAs for a duration
  - In many ways, this is what most people would think of as a call
  - Defined by Call-ID, To tag, and From tag
  - Sender selects Call-ID, and From tag
  - Receiver selects To tag

Transaction

- A *transaction* defines a particular operation that is carried out between the endpoints
  - Example: (INVITE) Set up the call, (BYE) end the call
- Includes all responses (including the ACK)
- May be several transactions within a dialog
- Each transaction uses a new CSeq
  - Chosen by the initiator of the transaction
  - Each side must increase the CSeq for each new transaction they initiate – no specified starting point
  - ***But*** since some transactions are initiated by each side, within a call it might look out of order

Dialogs and Transactions



Routing

- Each entity forwards message along path to reach destination
  - Places next destination into Request-URI (top line)
  - Adds themselves as another Via
  - Max-Forwards is decremented to prevent loops and limit the length of path
    - if it reaches 0, the message isn’t forwarded again

INVITE message

INVITE sip:bob@bigcompany.com SIP/2.0
Via: SIP/2.0/TCP aliceua.phonecompany.com:5060
From: Alice <sip:alice@phonecompany.com>;tag=ss95cbav
To: Bob <sip:bob@bigcompany.com>
Call-ID: 3848276298220188511@aliceua.phonecompany.com
CSeq: 1 INVITE
Max-Forwards: 10
Contact: <sip:alice@aliceua.phonecompany.com;transport=tcp>
…snip!...

Let's see what happens as this message passes through the big company proxy on the way to Bob…

INVITE message

INVITE sip:bob@bobua.bigcompany.com SIP/2.0
Via: SIP/2.0/TCP proxy.bigcompany.com:5060
Via: SIP/2.0/TCP aliceua.phonecompany.com:5060
From: Alice <sip:alice@phonecompany.com>;tag=ss95cbav
To: Bob <sip:bob@bigcompany.com>
Call-ID: 38482762982201885110@aliceua.phonecompany.com
CSeq: 1 INVITE
Record-Route: <sip:proxy.bigcompany.com;lr>
Max-Forwards: 9
Contact: <sip:alice@aliceua.phonecompany.com;transport=tcp>
…snip!...

Notice the new URI, added VIA, and decremented Max-Forwards. This message is forwarded to Bob's UA.

But one more here. What is record route?

Record Route

- Once a UA has the address of the other UA in the call, they can communicate directly if they wish
- Record route is a way for a proxy to request “keep me in the loop”
  - Ensures that future transactions include this proxy, rather than going directly to the other UA

Routing Responses

- When Bob’s UA responds, it “unrolls” the Via headers to decide where to route the responses
- The response traces the path back, sending the response to the Via address of the entity that sent the message (responding!)
- That entity removes itself from the Via and passes the message along, unrolling until it reaches the sender

Forking

- While somewhat complicated, many features require “forking”
  - Send a message to more than one entity Need to track (and manage both sides)
- Parallel forking (send to more than one at a time)
    - Ring both desk and mobile at once, cancel the call to whichever fails to pickup
- Can also use redirection (sequential)
    - Ring desk phone, then try mobile if no answer

SIP, SDP, and RTP

- SIP is only a part of the picture
- SIP is used to establish the multimedia session
- SIP messages don’t describe the type of media themselves
  - Embed another protocol called SDP
- Yet another protocol, RTP, is used to actually stream the media between the devices

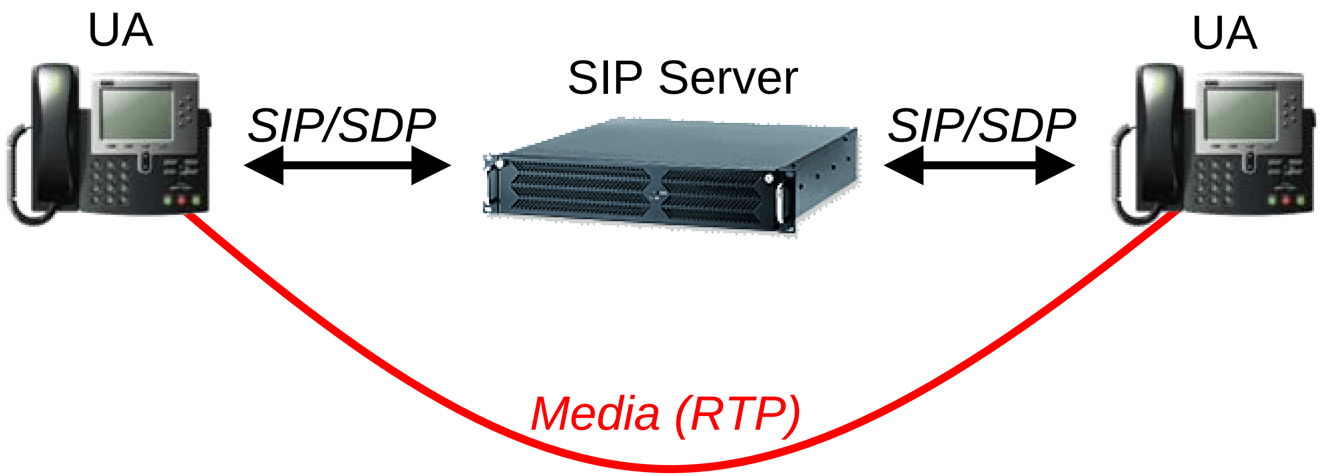
Codecs (encoding)

- The media that flows using RTP is encoded using a codec
- Codec is about how the audio has been converted to packets
  - Example: g.711, g.729, GIPS
  - In general, higher quality = more network bandwidth
  - Codec is described in SDP

SDP/RTP

- Both of these are older IETF protocols used by SIP
  - IETF tries to promote reuse whenever possible
- **S**ession **D**escription **P**rotocol
  - Used by SIP to describe the media format, encoding, destination, etc.
  - Included in SIP message as a payload
- **R**eal-time **T**ransport **P**rotocol
  - Used to transport the encoded media across the wire

The Protocols in a SIP Call



INVITE message

```
INVITE sip:bob@bigcompany.com SIP/2.0
Via: SIP/2.0/TCP aliceua.phonecompany.com:5060
From: Alice <sip:alice@phonecompany.com>; tag=ss95cbav
To: Bob <sip:bob@bigcompany.com>
Call-ID: 3848276298220188511@alicaua.phonecompany.com
CSeq: 1 INVITE
Contact: <sip:alice@aliceua.phonecompany.com;transport=tcp>
Content-Type: application/sdp
Content-Length: 134

v=0
o=SIPeerior-UA 10010 605 IN IP4 aliceua.phonecompany.com
s=SIP Call
c=IN IP4 192.0.1.1
t=0 0
m=audio 49172 RTP/AVP 0 8 16
```

INVITE message

<snip!>

Content-Type: application/sdp

Content-Length: 134

The content of the message is SDP – to describe the media of the call. We have 134 bytes of SDP

v=0

v is the SDP version number -- 0

o=SIPeerior-UA 10010 605 IN IP4 aliceua.phonecompany.com

o is a description of the UA and a session ID

s=SIP Call

s is an string to encode a name for this session

c=IN IP4 192.0.1.1

c tells the other side what address to send the media to

t=0 0

t indicates start and end times. 0 and 0 don't specify times in advance

m=audio 49172 RTP/AVP 0 8 16

m describes the media. We are listening for audio of type RTP/AVP on port 49172

Offer/Answer

- Different UAs may support different codecs
- Calling UA lists what it supports and is willing to use in the INVITE message SDP
- Called UA responds with the subset of that it is willing to use in the response SDP (200 OK)
- If nothing in common, can't communicate and call is rejected
- Mechanisms exist to change this mid call (for example, switching from audio to video)

Instant Message and Presence

- SIP has a set of extensions called SIMPLE
  - **SIMPLE: SIP** for **I**nstant **M**essaging (IM) and **P**resence **L**everaging **E**xtensions
  - Allows sending text messages between devices
  - Can cleanly transition to a voice chat, since the session is all SIP
  - Presence (ability to see “status” of other user) is also supported (and persistent on server)
    - Idle, Offline, Available, Busy, Offline till Monday…

Instant Message and Presence

- MESSAGE method used to send text messages
- SUBSCRIBE to express an interest, “subscribe” to someone’s status
- PUBLISH to change your status on the presence server
- NOTIFY is sent from server to those who have subscribed

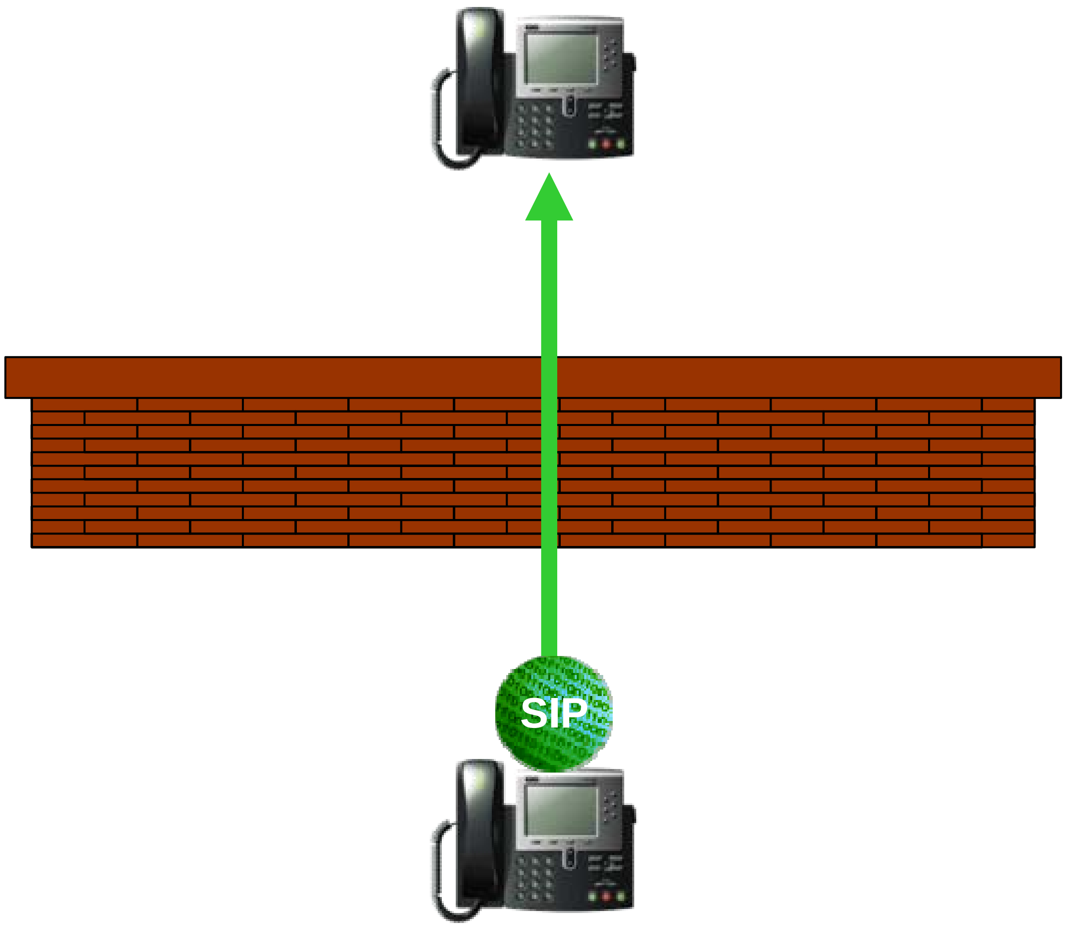
Extending SIP

- SIMPLE isn’t the only extension
- SIP was designed from the beginning to be highly extensible
- Can add new methods, new responses, headers
- Several groups at IETF determine what is really an accepted “standard” extension
  - SIP, SIPPING, SIMPLE working groups
- Many vendors add non-interoperable “extensions” to SIP
  - These may not work across different vendor platforms

NAT/FW Traversal

- SIP has traditionally had some trouble getting through firewalls (FWs) and **N**etwork **A**ddress **T**ranslators (**NAT**s)
- Two major causes
  - NATs use "private" IP addresses, not routable from the outside
    - SIP embeds these, outside entity has no way to respond
  - SIP and RTP (the media) are on different "ports"
    - Each requires a specific forwarding through the firewall to work and must be opened separately

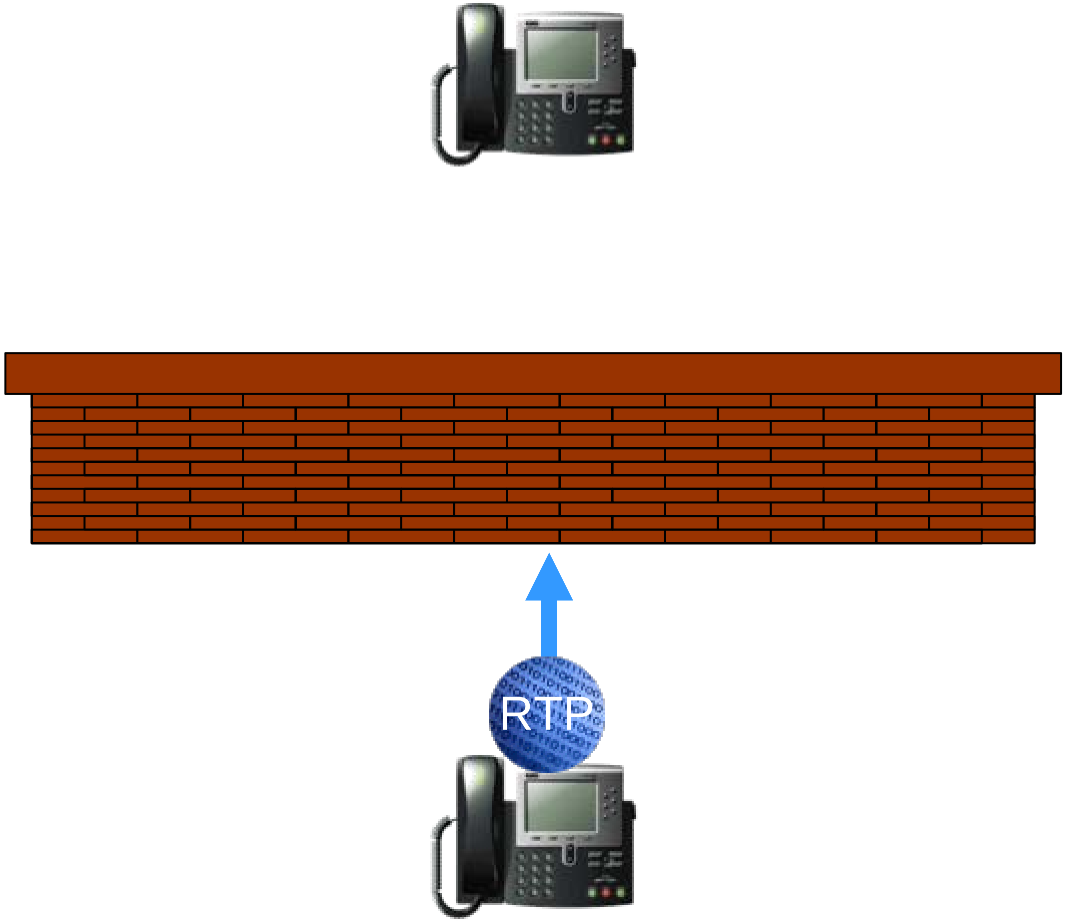
Firewalls and Media



A UA Contacts another UA (or a proxy), sending SIP messages.

The process of sending the message (on most NATs/FWs) creates a mapping that allows responses to reach the sender

Firewalls and Media



Media (RTP) is on a different port (different opening in the FW) than SIP

It is possible that at the time the media is sent, it is rejected by the firewall

NAT/FW Traversal

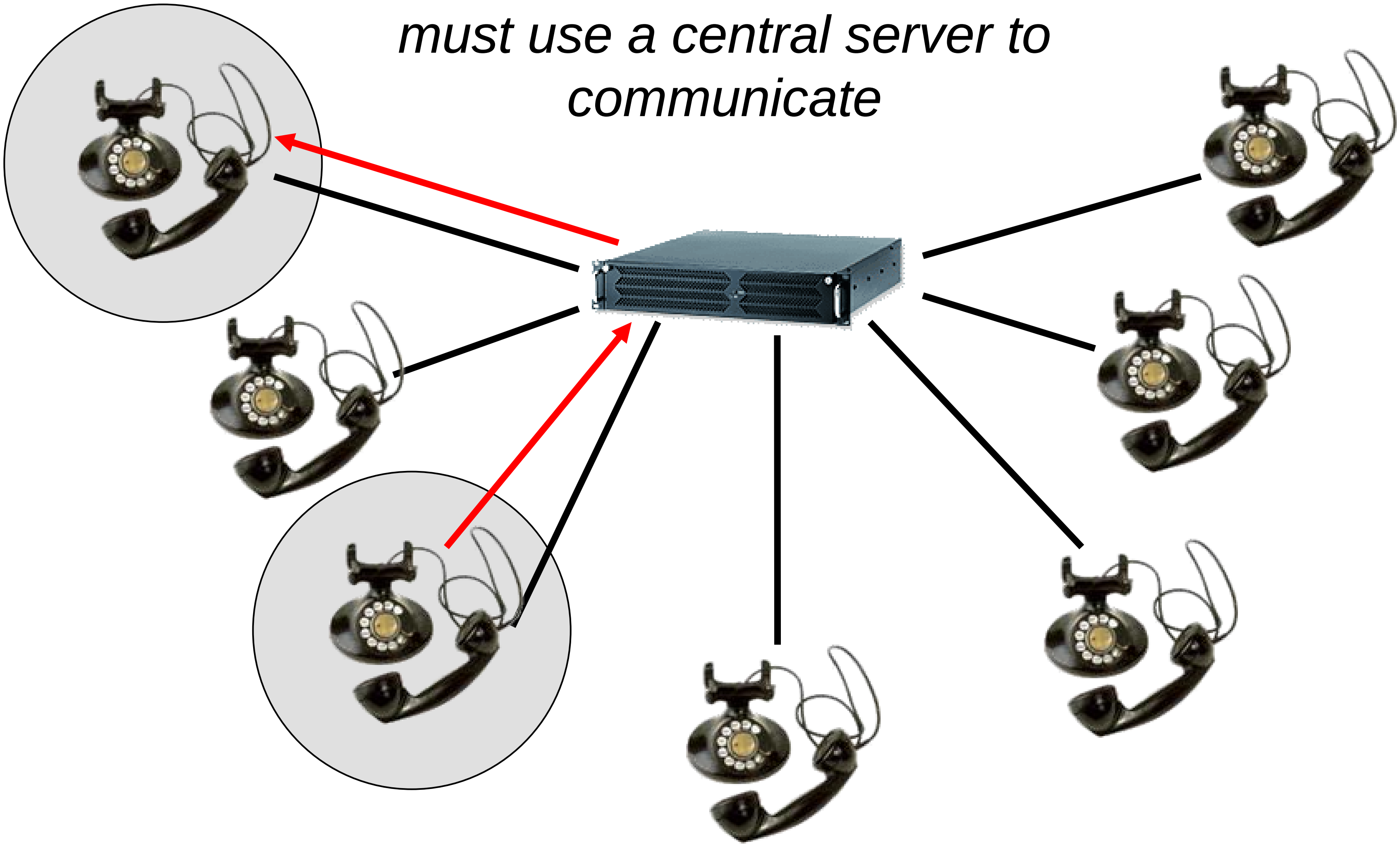
- Several Solutions
  - IETF efforts
    - STUN, TURN and ICE
    - Query outside servers to find out “public” address
    - Can use media relays to help get media between two hosts behind firewalls
  - **S**ession **B**order **C**ontrollers (**SBC**s)
    - Sit near the firewall/NAT, rewrite the internals of the packets, work with firewall to open ports

Peer-to-Peer (P2P) SIP

- Growing but very new area
- Basic idea – reduce or completely eliminate the central servers (proxy/softswitch)
- Essentially all functionality is moved to the UA
- IETF is looking at this, a few companies have (for now nonstandard) implementations
- Very cost effective for small enterprises
- Highly scalable, so also good for internet-wide deployments

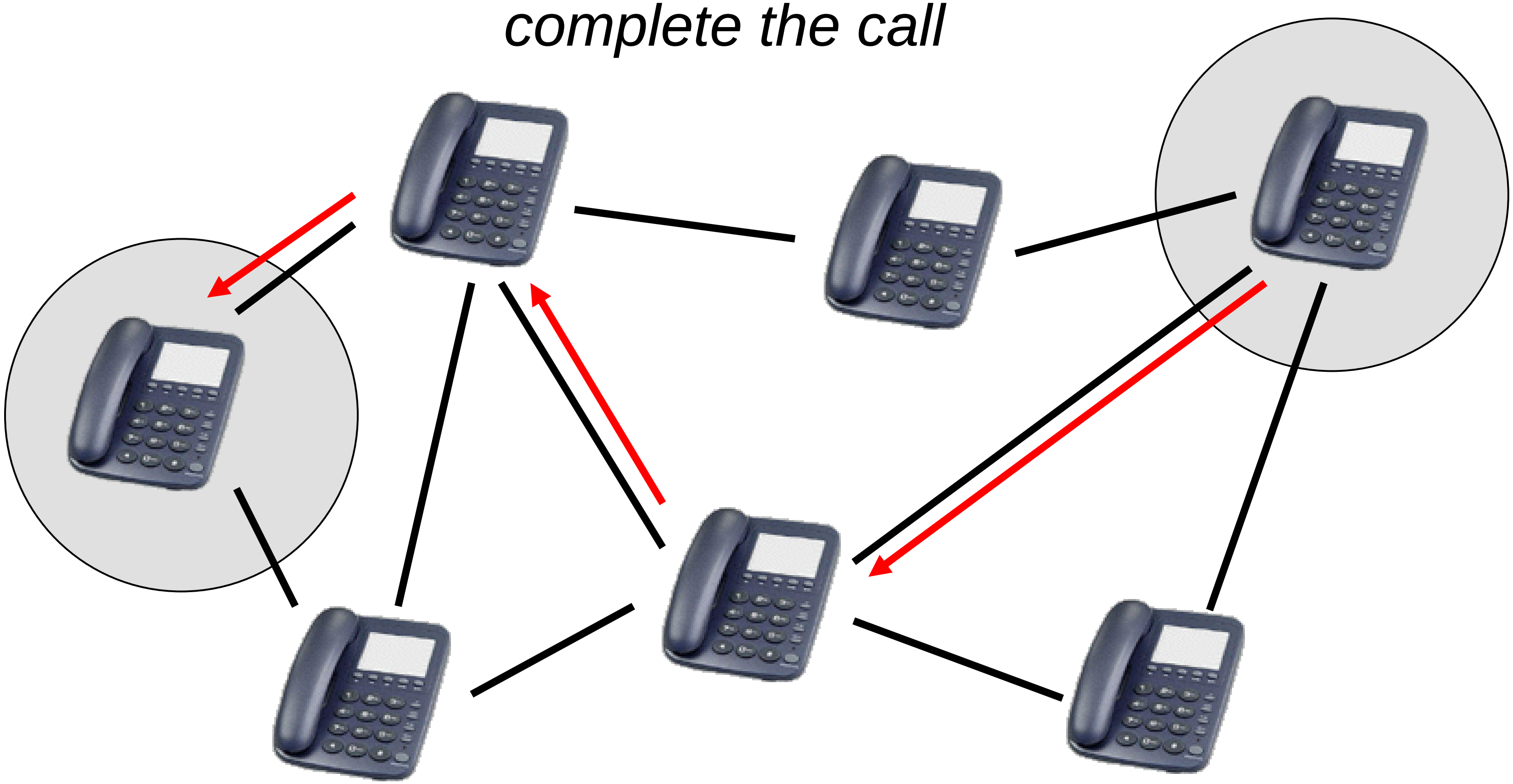
Client/Server Session

In a Client/Server session, two nodes must use a central server to communicate



P2P Session

In a Peer-to-Peer session, when two nodes communicate, a few other nodes, rather than a central server, help complete the call



Open Source SIP projects

- Several good open source SIP projects
  – ReSIProcate / Repro
    - Highly compliant stack and proxy
  – VOCAL
    - Full SIP “softswitch” – includes proxy, softclient, voice mail, etc.
  – IPTel Sip Express Router
    - High performance Proxy. Less full featured than VOCAL, but faster and more robust

Some Good SIP Webpages

- Tech-Invite (www.tech-invite.com) is a great source for general SIP information with many beautiful color illustrations
- SIP Tutorial at iptel.org, (www.iptel.org/sip/siptutorial.pdf)
- SIP versus H.323, also at iptel.org, (www.iptel.org/info/trends/sip.html)
- ReSIProcate open source stack and proxy project (www.sipfoundry.org)
- IETF information can be found at www.ietf.org and www.softarmor.com
- P2PSIP.org (www.p2psip.org) is the biggest community site for P2PSIP

Books on SIP

- **SIP Demystified**, *Gonzalo Camarillo*, McGraw-Hill Telecom Series, 2002
- **Internet Communications Using SIP**, *Henry Sinnreich and Alan B. Johnston*, Wiley Networking Council Series, 2001

Acknowledge/Thanks

- Thanks to those who have worked on SIP over the years, presented SIP tutorials, or written SIP books!