SIP: Call Setup and Beyond

Henning Schulzrinne
Dept. of Computer Science
Internet Real-Time Laboratory (IRT)
Columbia University
New York, New York
schulzrinne@cs.columbia.edu

Ericsson Signaling Days 2000 (Kalrlstad, Sweden)

November 14, 2000

With Jonathan Rosenberg, Adam Roach and other participants in the SIP WG

Overview

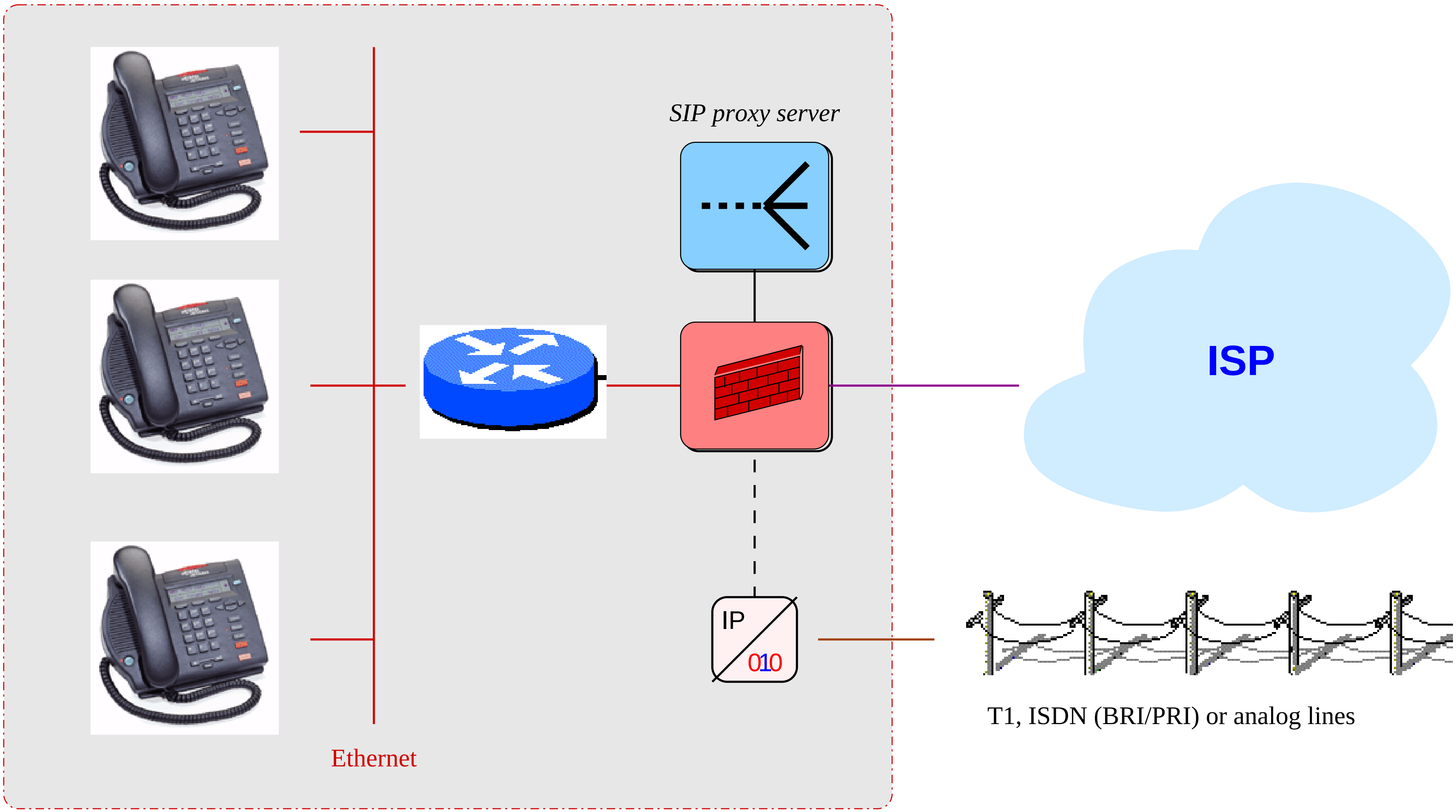
- VoIP service models
- the IETF VoIP architecture
- the Session Initiation Protocol (SIP)
- programming Internet telephony services
- challenges on the horizon:
  - emergency services
  - instant messaging & presence
  - generic event notification
  - integration with 2G mobile (GSM, CDMA)
  - next-generation wireless (3GPP, 3GPP2, MWIF, . . .)

Internet Telephony Service Models

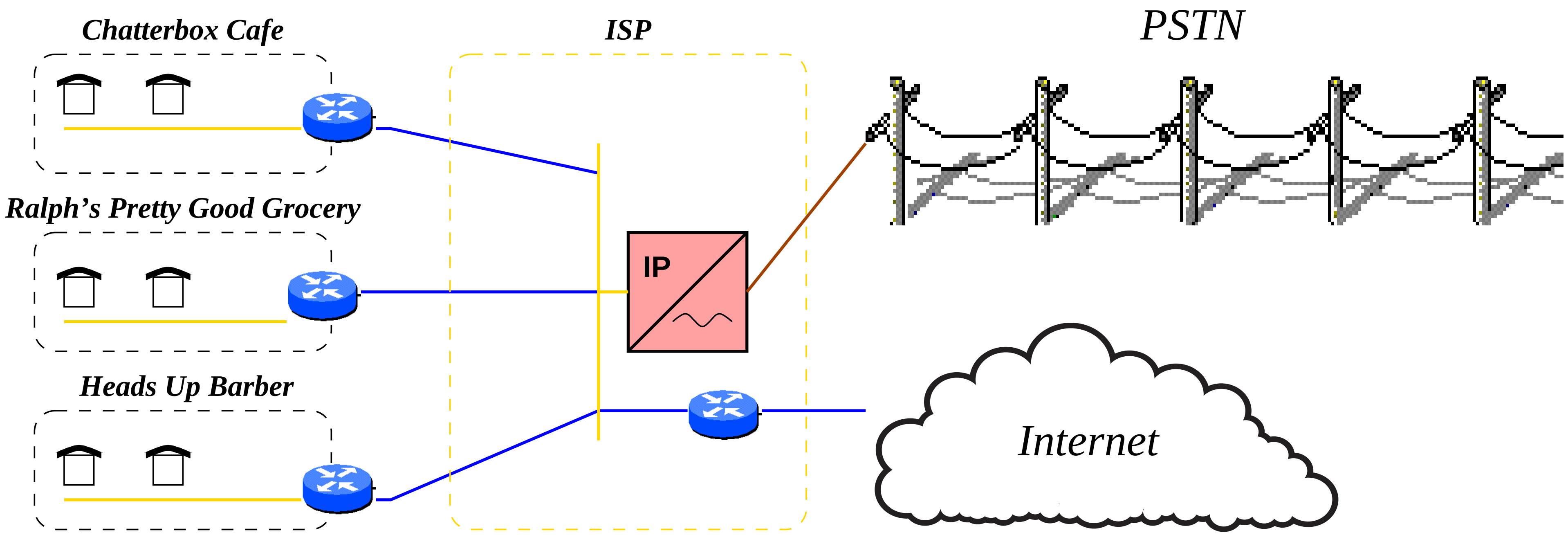
- Internet “PBX”
- Internet Centrex
- Internet Carrier

➠ same basic equipment, but size of gateway varies

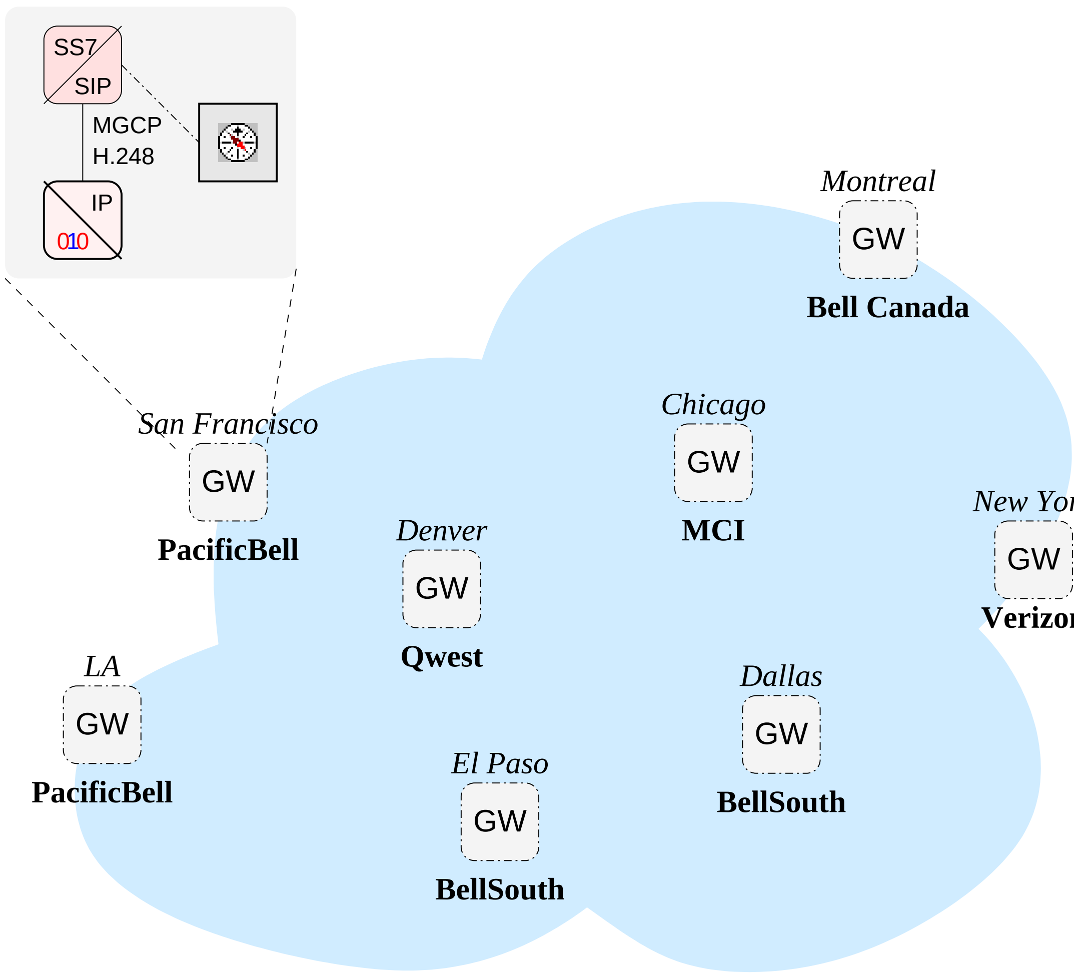
Internet PBX



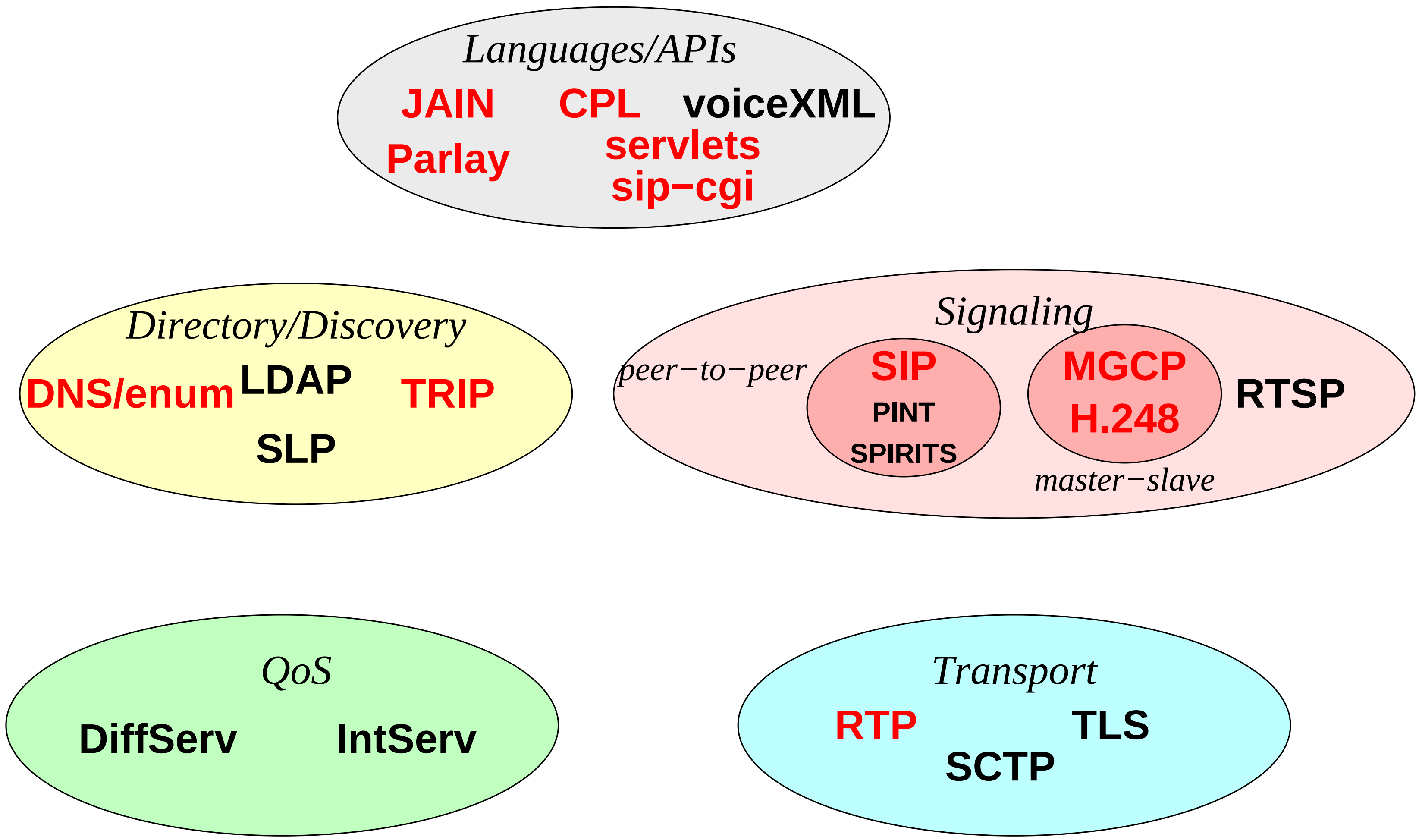
IP Centrex


Chatterbox Cafe
ISP
PSTN
Ralph's Pretty Good Grocery
IP
Heads Up Barber
Internet

IP Carrier


SS7
SIP
MGCP
H.248
IP
010
Montreal
GW
Bell Canada
Chicago
GW
MCI
San Francisco
GW
PacificBell
Denver
GW
Qwest
New York
GW
Verizon
LA
GW
PacificBell
Dallas
GW
BellSouth
El Paso
GW
BellSouth

IETF VoIP Protocol Architecture



IETF VoIP Protocols & APIs

Most protocols are re-used $\longrightarrow$ core

SIP	session setup, services
MGCP	gateway control
SDP	describe multimedia sessions
RTP	multimedia transport
TRIP	find nearest gateway

CPL	XML-based language
sip-cgi	SIP-based scripts

IETF Protocol Reuse

protocol	designed for	VoIP use
RTSP	streaming media	voicemail
DNS	name lookup	E.164 mapping
SCTP	reliable transport	ISUP transport
PGP	email	call authentication
MIME	email	signaling info
SDP	multicast sessions	SIP, MGCP

Protocol “Holes”

- “tight” session control for conferences
  - admission control
  - multicast key distribution
  - advanced capability negotiation
- scalable authentication for individuals
- cross-provider QoS: primarily a business problem

IETF VoIP Architecture Characteristics

- universal identifier *user@domain*: SIP URL = email = NAI
- separation of transport of services
- media-neutral, including beyond audio and video
- emphasis on user-programmable services
- web integration: content, mutual referral
- integration with IM and presence

SIP Overview

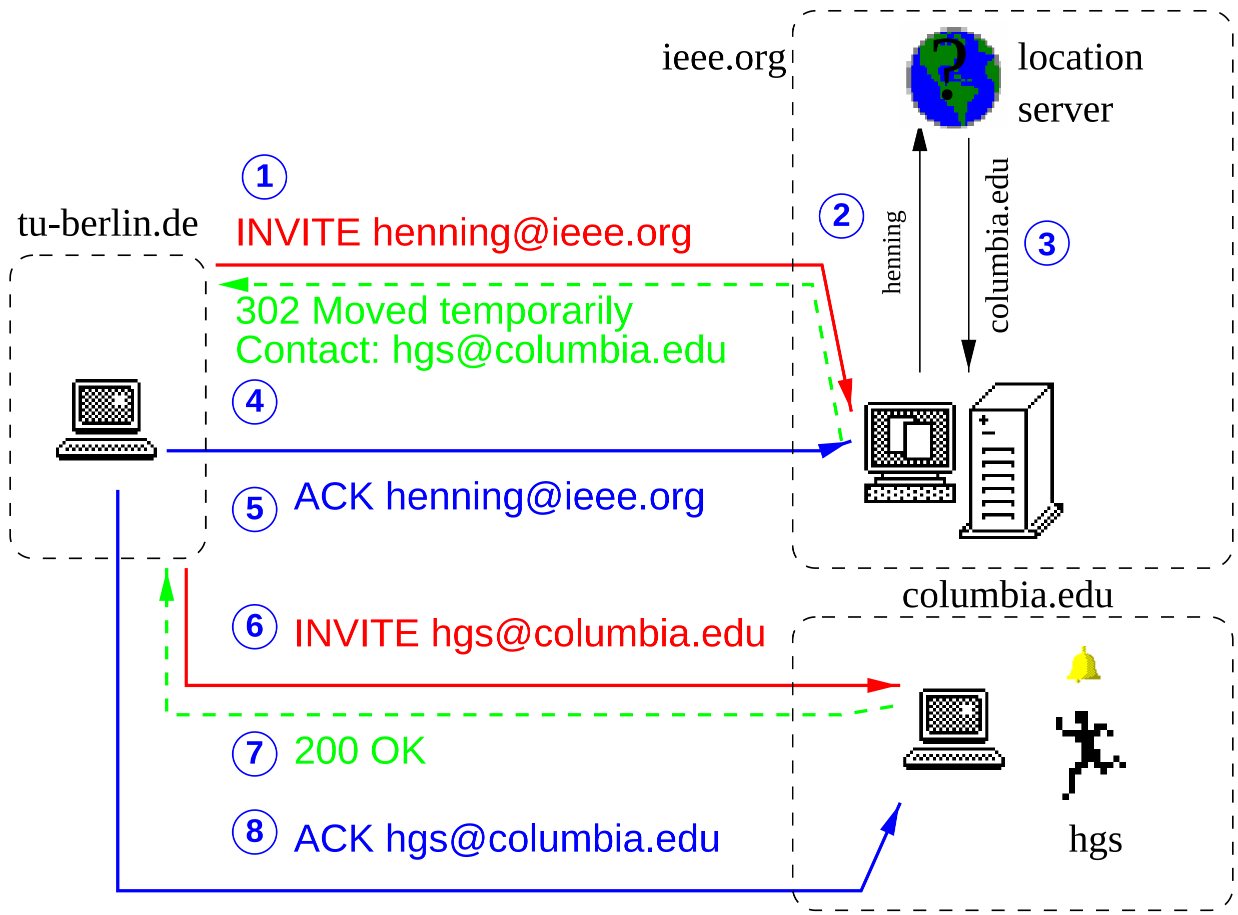
- protocol for establishing, modifying, tearing down (multimedia) sessions
- IETF Proposed Standard since March 1999
- multimedia = audio, video, shared applications, text, . . .
- also used for “click-to-dial” (PINT wg) and possibly Internet call waiting (SPIRITS wg)
- to be used for PacketCable Distributed Call Signaling
- to be used for Third-Generation Wireless (3GPP, 3GPP2)

SIP Components

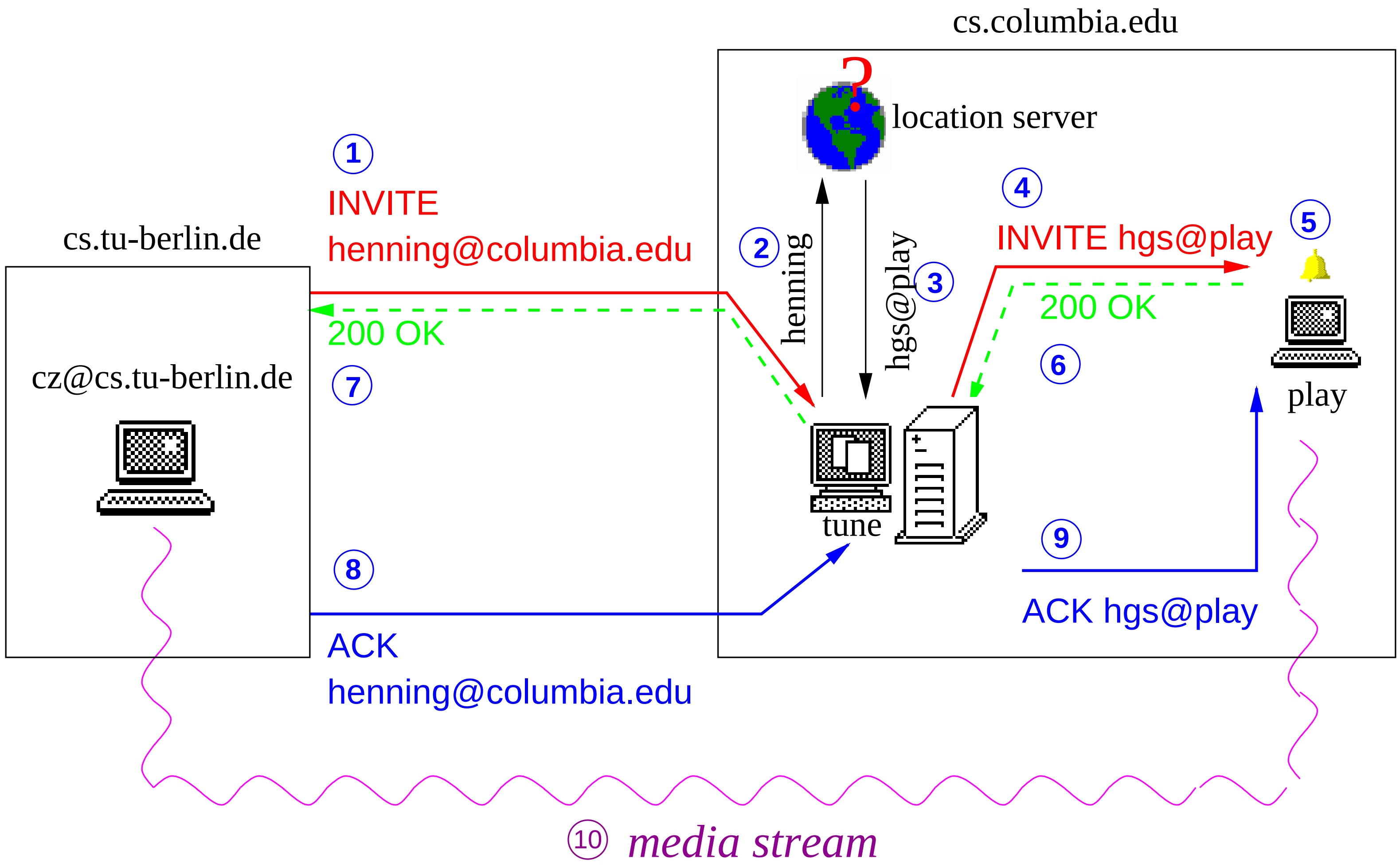
entity	does	examples
proxy server	forward calls	firewall controller, “call router”
redirect server		“application server”
user agent	end system	SIP phone, gateway, “softswitch”
registrar	location mgt.	mobility support

Roles are changeable, on a request-by-request basis

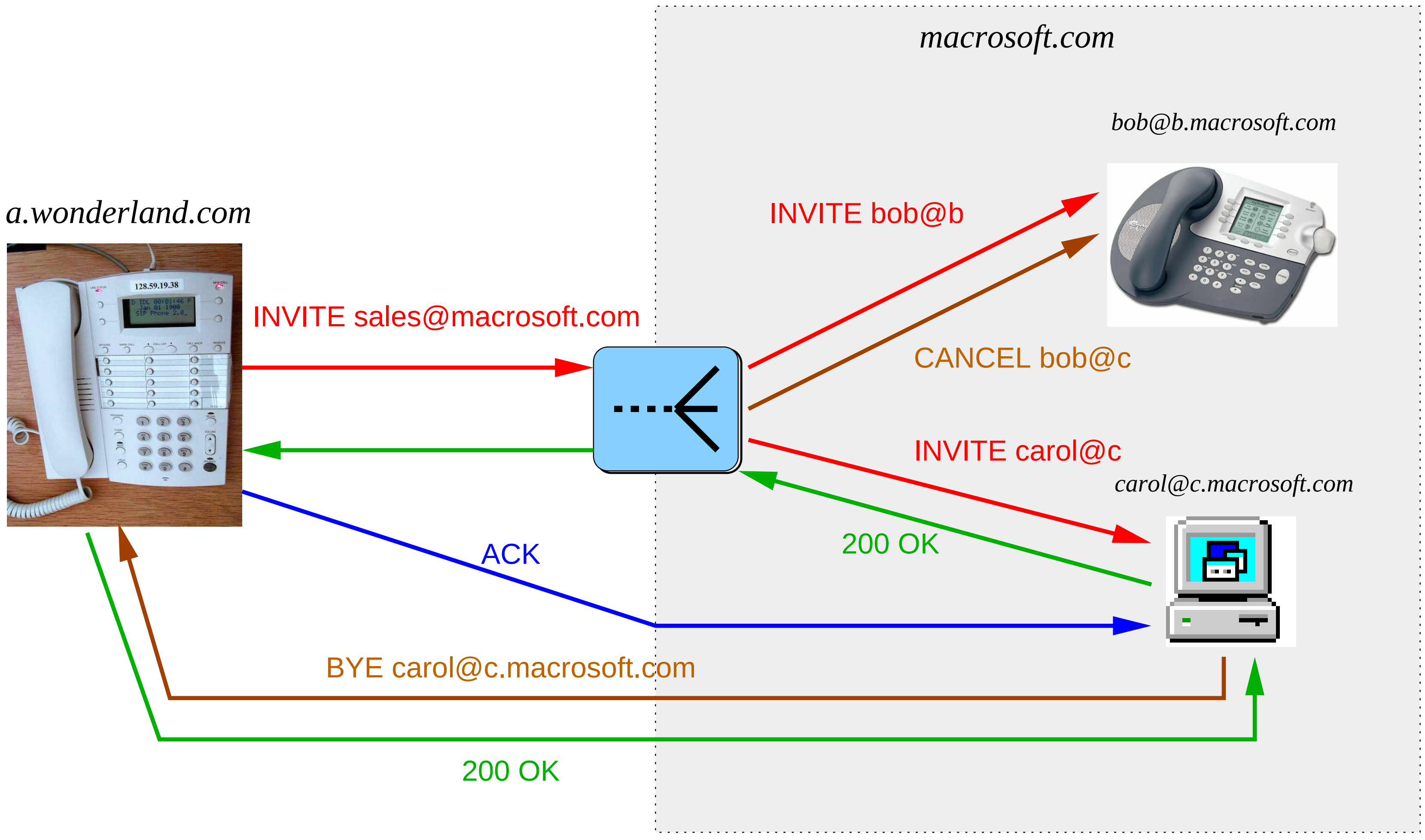
SIP Example: Redirection



SIP Example: Proxying



SIP Forking Proxies



SIP syntax

request

method URL **SIP/2.0**

response

SIP/2.0 *status reason*

message header

Via: **SIP/2.0/** *protocol host:port*
From: *user <sip:from_user@source>*
To: *user <sip:to_user@destination>*
Call−ID: *localid@host*
CSeq: *seq# method*
Content−Length: *length of body*
Content−Type: *media type of body*
Header: *parameter ;par1=value ;par2="value"*
;par3="value folded into next line"

blank line

message body

V=0
o= *origin_user timestamp timestamp* **IN IP4** *host*
c=IN IP4 *media destination address*
t=0 0
m= *media type port* RTP/AVP *payload types*

message

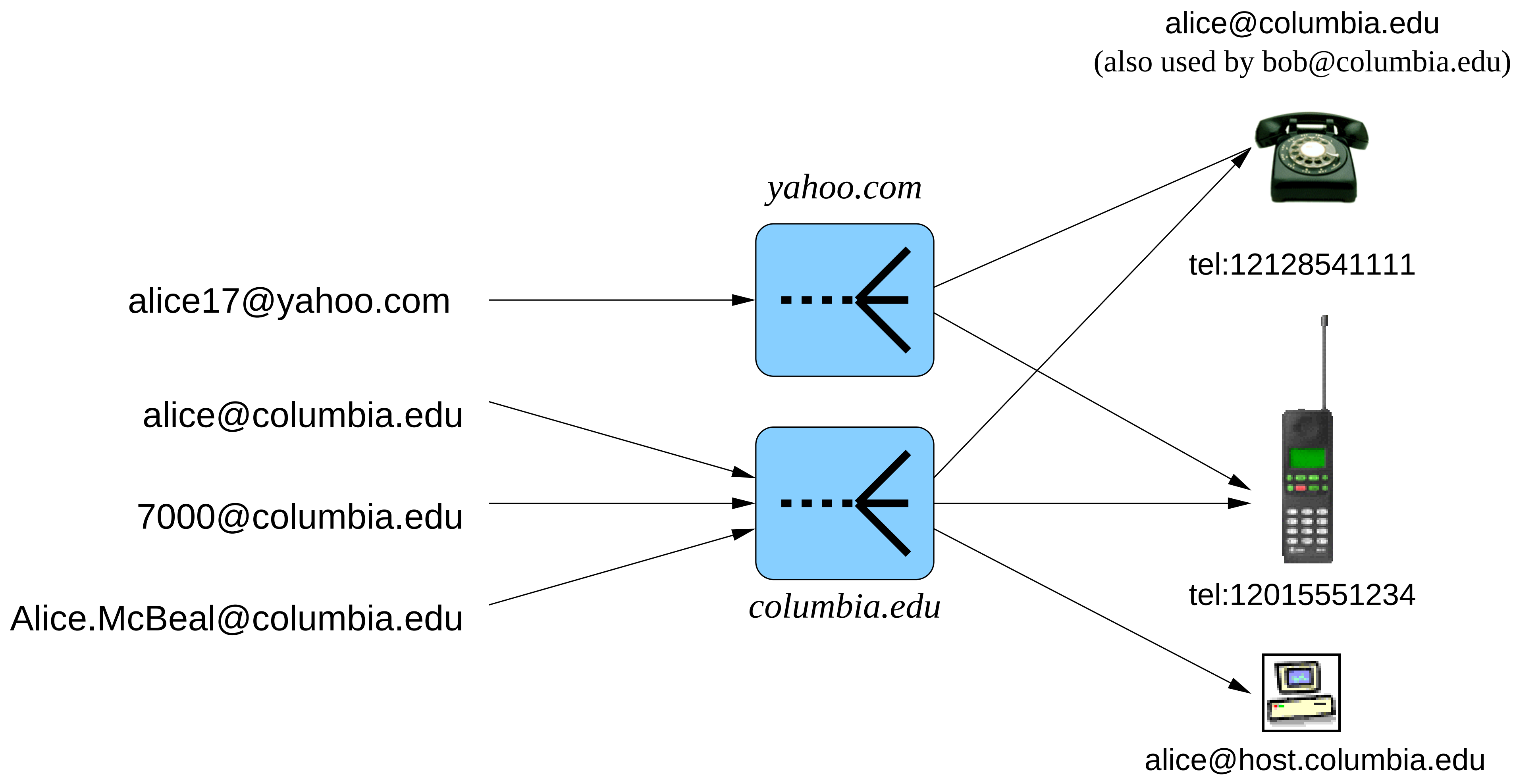
SIP Advanced Features

- forking
- extensibility: new headers, methods, bodies
- security: web-like, PPP/CHAP or PGP
- multicast-capable
- support for personal, session, terminal, service mobility
- caller preferences: direct calls based on properties

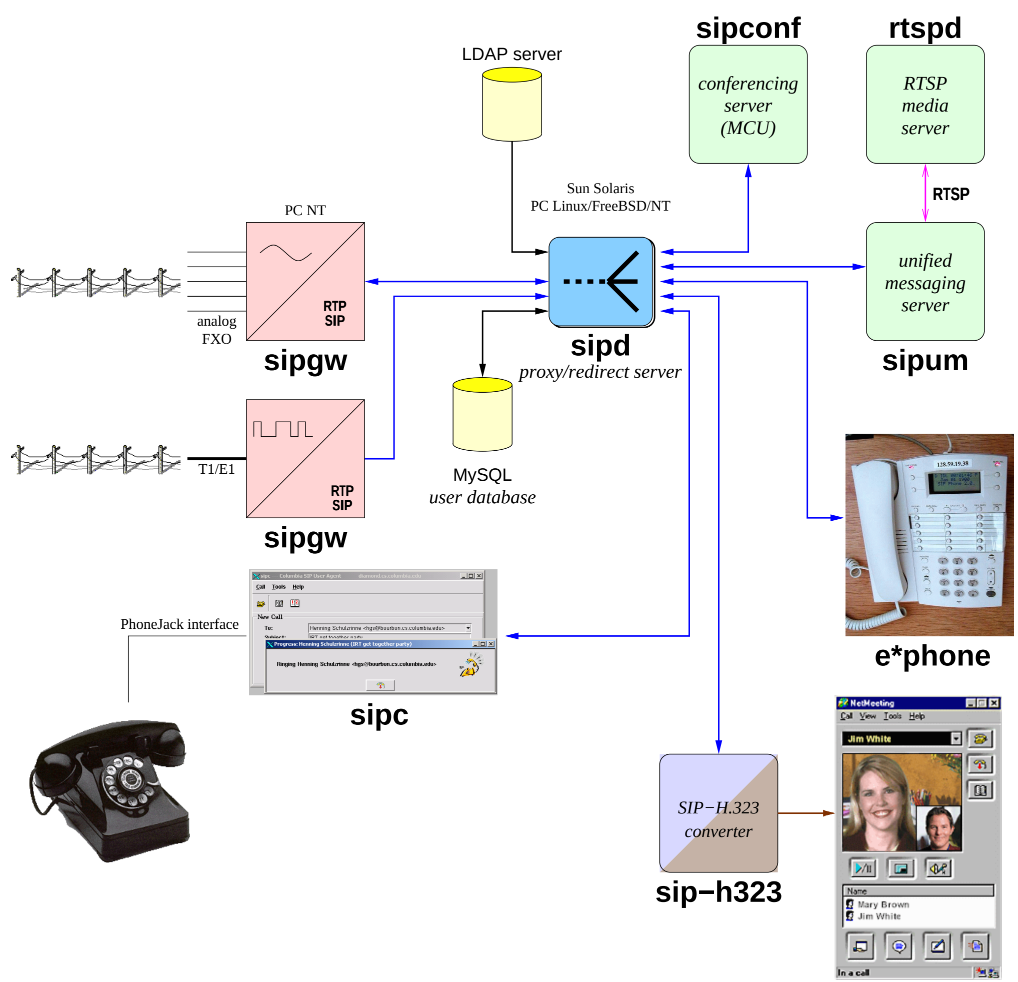
SIP Mobility

terminal	cross-provider	REGISTER, re-INVITE
personal	different terminals, same address	REGISTER
service	different terminals, same services	upload
session	move sessions across terminals	REFER

SIP Personal Mobility



Example SIP System



SIP-Based Telephony Services

conferencing	“dial-in”, “dial-out”
forwarding	basic SIP
ACD	proxy, no protocol extensions
call transfer	REFER extension
DTMF transport	in RTP, not SIP
billing	in resource reservation, (mostly) not SIP

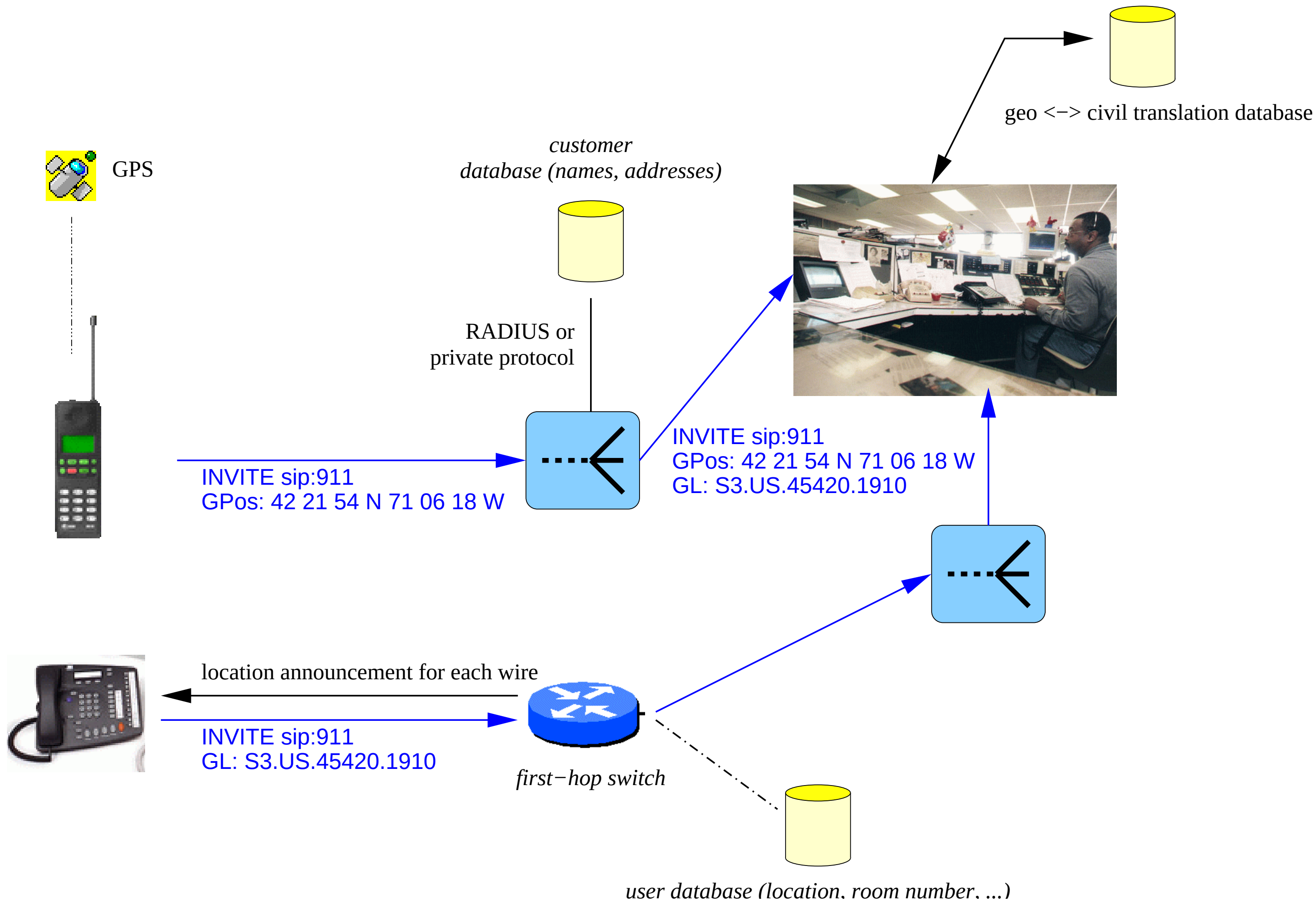
Current SIP efforts

- SIP to Draft Standard
- QoS and security preconditions
- inter-domain AAA and billing
- session timer for liveness detection
- early media (PSTN announcements)
- SIP for presence / instant messaging
- SIP-H.323 interworking
- reliable provisional responses
- DHCP configuration for finding SIP servers
- SIP for firewalls and NATs
- caller preferences
- services (transfer, multiparty calls, home)
- ISUP carriage

SIP Emergency Services

- need
  - emergency address
  - find nearest PSAP
  - PSAP determines caller location
- cannot just rely on gateway calling 911
- generally, allow devices to be location-aware (“what time is it where I’m about to call?” “call pizza parlor”)
- offers new opportunities: database access, video, measurements, accessibility, ...

SIP Emergency Services



SIP Bake-Off

- takes place every four months, 5th at Pulver.com August 2000
- 45 organizations from 11 countries
- about 50-60 implementations:
  - IP telephones and PC apps
  - proxy, redirect, registrar servers
  - conference bridges
  - unified messaging
  - protocol analyzers
- first IM/presence interop test
- emphasis on advanced services (multi-stage proxying, tel URLs, call transfer, IVR, . . .)

The Dangers of VoIP

- focus on single service: voice, fax, ...
- PSTN: service orientation $\longleftrightarrow$ Internet: neutral transport
- APIs as least common denominator across POTS, ISDN, SS7 $\longrightarrow$ 100-year old functionality
- carbon-copy replication of existing services
- terminology overload

Differences: Internet Telephony ↔ POTS

- separate control, transport (UDP) ➠ no triangle routing
- separate connectivity from resource availability
- separate services from bit transport
- datagram service ➠ less bootstrapping
- in-band signaling ➠ higher speed
- features “network” → end system: distinctive ringing, caller id, speed dialing, number translation, . . . ➠ scaling
- features: intra-PBX = inter-LATA and general
- protocols: user-network = network-network signaling

PSTN Legacies to Avoid

- E.164 numbers – might as well wear bar codes
- tones (e.g., failure indications)
- in-band signaling (DTMF)
- systems with user interface knowledge (12 keys, voice)
- voice-only orientation (e.g., MGCP/Megaco)
- integration of bit transport and services
- service-specific billing
- trigger model for service creation
- trusted networks without crypto authentication

$\longrightarrow$ confine PSTN knowledge to edge of network

Replication of Existing Services

- “user is familiar with PSTN services”
- but how many users actually know how to use call transfer or directed pick-up?
- user interface is often just legacy of key systems or other ancient technology
- avoid binding of identifiers to devices – call person or group of people, regardless of location
- instead, model desired behavior
- single-server features don’t need standardization
- find general mechanisms (e.g., REFER for three-party calls and various call transfers)

Terminology Overload

Invasion of the meaningless technical-sounding terms, attempting to familiar mimic PSTN boxes:

- CO switch $\longrightarrow$ soft switches = gateway + SIP UA + ?
- SCP $\longrightarrow$ application servers = proxy? web server? media server?
- PBX $\longrightarrow$ Internet PBX = proxy? + gateway?
- ...

Temptation: new name $\longrightarrow$ new protocols, APIs, ... – the old box boundaries don't necessarily make sense!

It's That Simple…

We really only have a few basic components:

- PSTN gateway, with some combination of FXO/FXS
- SIP proxy/redirect/registrar servers (or H.323 gatekeepers)
- SIP user agents (or H.323 terminals): PCs, phones
- media storage servers
- DNS, directory, web, email, news, … servers

Invisible Internet Telephony

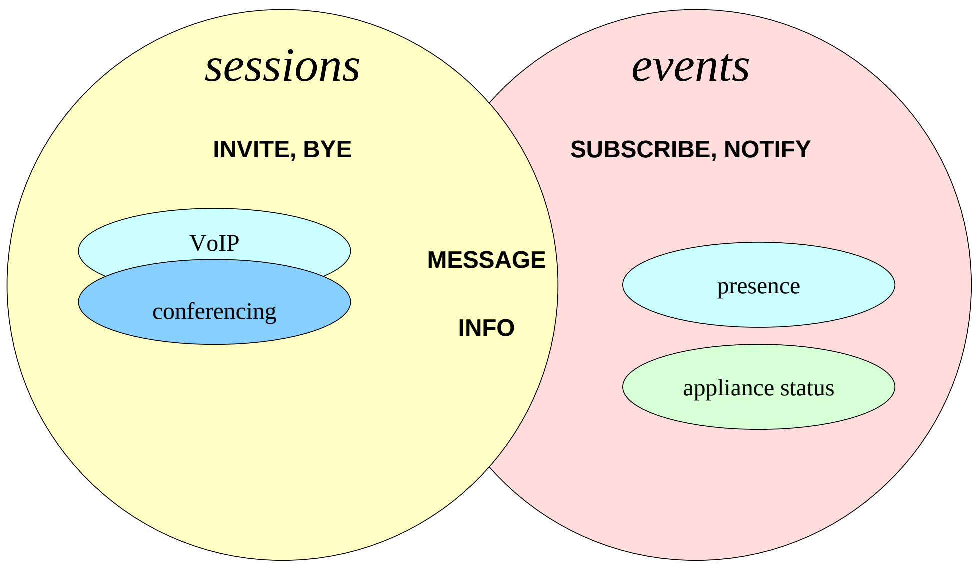
VoIP technology will appear in . . .

- Internet appliances
- home security cameras, web cams
- 3G mobile terminals
- fire alarms
- chat/IM tools
- interactive multiplayer games

The Largest Signaling Network is Not Running SS7

- AT&T: 280 million calls a day
- AOL: 110 million emails/day, total about 18 billion/day
- total $>$ 1 billion instant messages a day (AOL: 500 million)
- signaling effort of call $\approx$ IM

Signaling and Events



Signaling: "do this" (push) – Events: "this just happened"

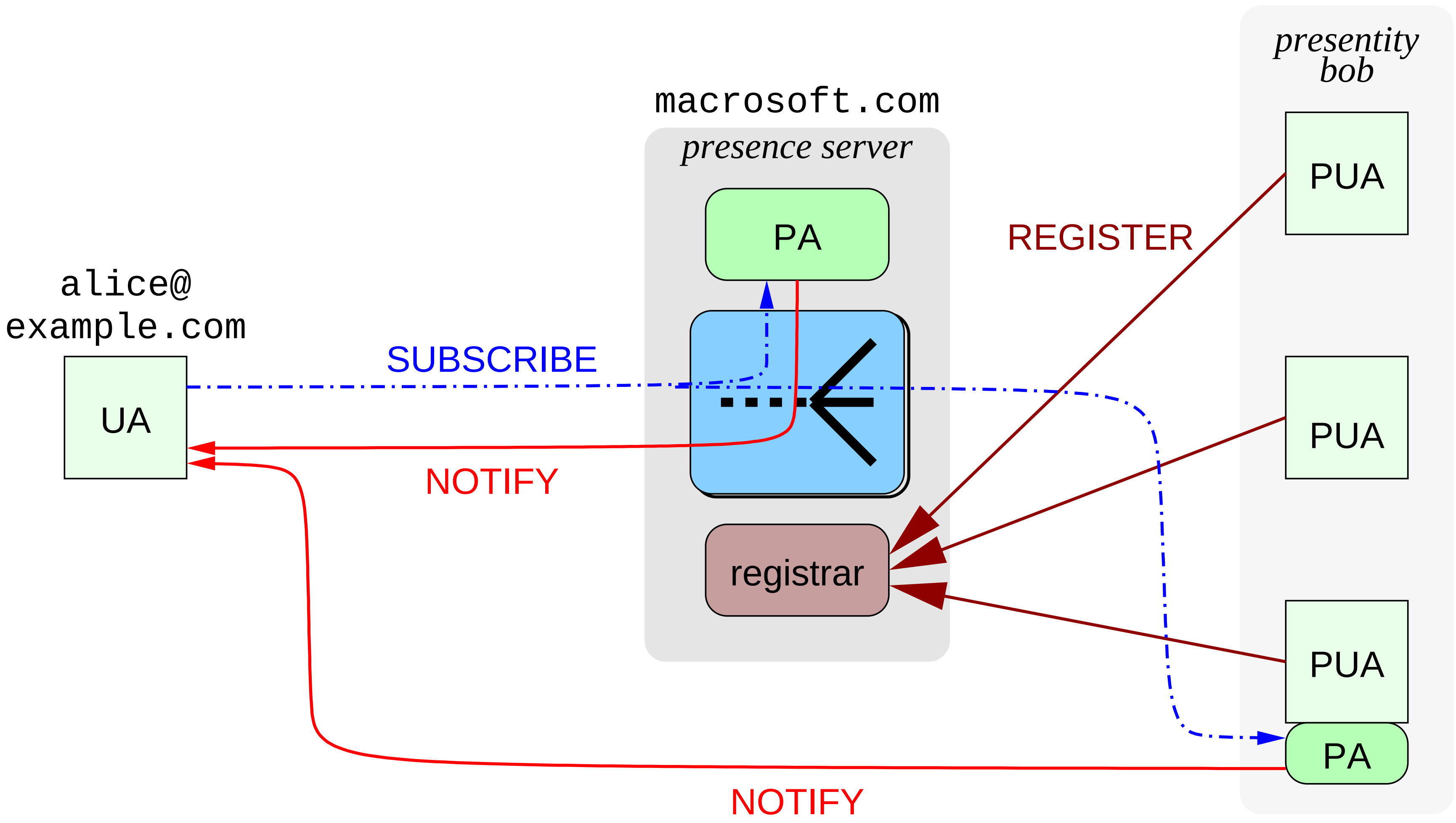
Commonalities between Signaling and Events

- presence is just a special case of events: “Alice just logged in” $\approx$ “temperature in boiler exceeds $300°$ F”
- need to *locate* mobile end points
- may need to find several different destinations (“forking”)
- same addressing for users
- presence often precursor to calls
- may replace call back and call waiting
- likely to be found in same devices
- events already in VoIP: message alert, call events

SIP as a Presence & Event Platform

- requires minimal extensions to SIP: SUBSCRIBE to ask to be alerted, NOTIFY when event occurs
- MESSAGE for sending text messages (“IM”)
- with forking, can easily register MESSAGE recorder
- true “chat” is voice (+ video)
- services such as reaching mobile phone while in meeting
- types of events:
  - inside existing call leg
  - within call, but outside call leg
  - unrelated to call leg

SIP Presence Architecture



SIP Presence Components

Presentity: logical entity being subscribe to, e.g., alice@wonderland.com, with several agents

Registrar: receives REGISTER requests

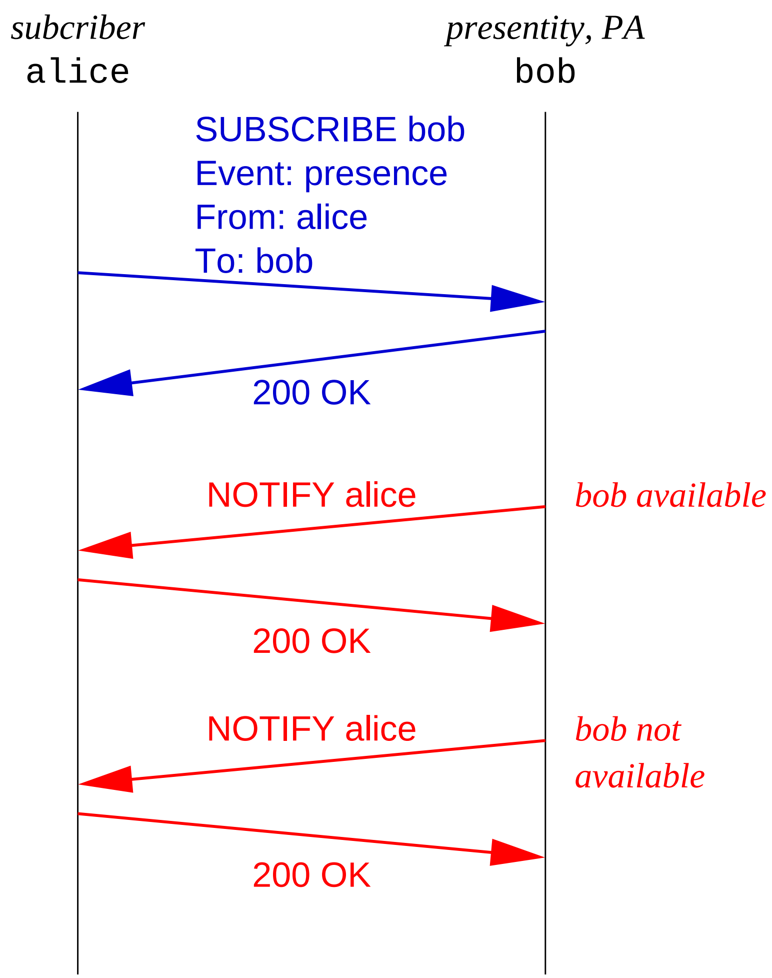
Presence user agent (PUA): generates REGISTER, but no SUBSCRIBE or NOTIFY ➠ any non-presence-aware SIP software

Presence agent: receive SUBSCRIBE, generate NOTIFY

Presence server: SIP proxy + PA

Presence client: SIP UA + PA

SIP Presence Protocol



SIP SUBSCRIBE Example

```
SUBSCRIBE sip:bob@macrosoft.com SIP/2.0
Event: presence
To: sip:bob@macrosoft.com
From: sip:user@example.com
Contact: sip:user@userpc.example.com
Call-ID: knsd08alas9dy@3.4.5.6
CSeq: 1 SUBSCRIBE
Expires: 3600
Content-Length: 0
```

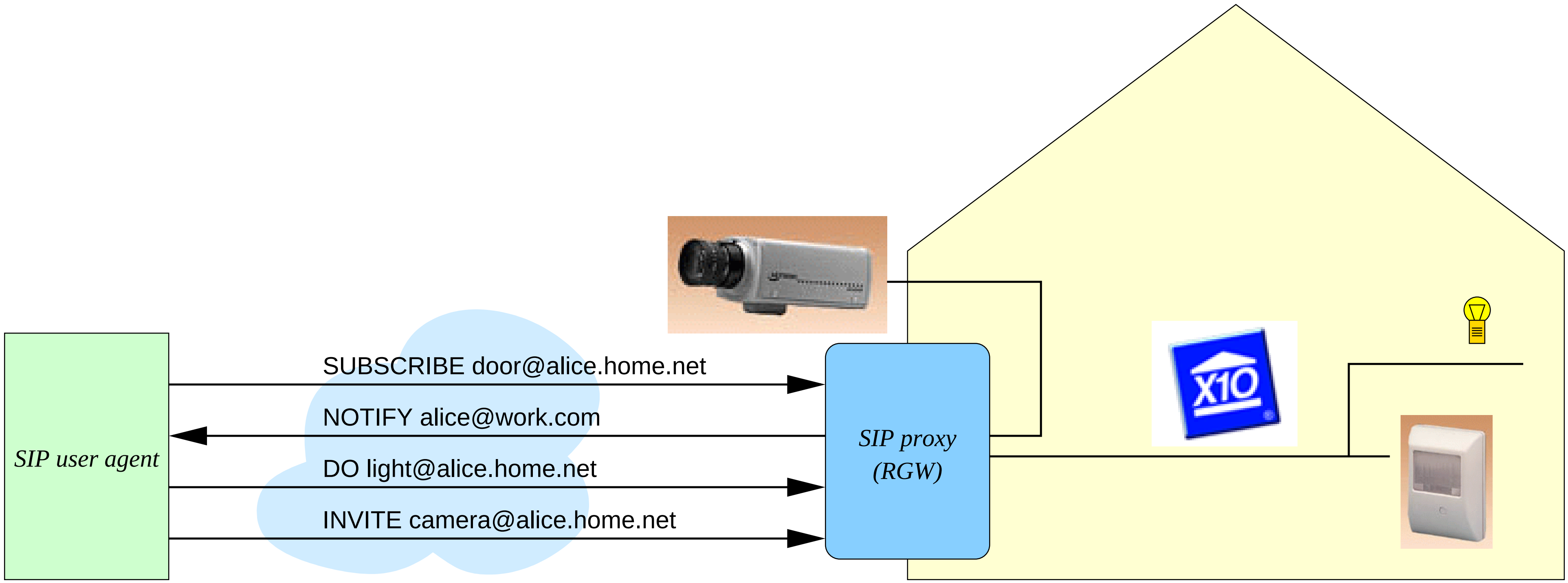
- Forked to all PUAs that have REGISTERed with method SUBSCRIBE.
- 200 (OK) response contains current state.

SIP NOTIFY Example

```
NOTIFY sip:user@userpc.example.com
To: sip:user@example.com
From: sip:alice@wonderland.com
Call-ID: knsd08alas9dy@3.4.5.6
CSeq: 1 NOTIFY
Content-Type: application/xpidf+xml

<?xml version="1.0"?>
<!DOCTYPE presence
 PUBLIC "-//IETF//DTD RFCxxxx XPIDF 1.0//EN" "xpidf.dtd">
<presence>
  <presentity uri="sip:alice@wonderland.com;method="SUBSCRIBE">
    <atom id="779js0a98">
      <address uri="sip:alice@wonderland.com;method=INVITE">
       <status status="closed"/>
      </address>
    </atom>
  </presentity>
</presence>
```

Events: SIP for Appliances



(Work with Telcordia)

Programmable Internet Telephony

	APIs	servlets	sip-cgi	CPL
Language-independent	no	Java only	yes	own
Secure	no	mostly	no, but can be	yes
End user service creation	no	yes	power users	yes
GUI tools w/portability	no	no	no	yes
Call creation	yes	no	no	no
Multimedia	some	yes	yes	yes

Example: integration with iCal $\longrightarrow$ automatically export personal calendar to call handling

Conclusion

- basic IETF-based architecture in place
- SIP as foundation for services – see http://www.cs.columbia.edu/sip
- extensions to mobility, emergency services, . . . in progress
- first (and last?) chance to recover from 120 years of legacy
- avoid replication of PSTN on packets
- most VoIP applications won't look like a telephone
- opportunities in emergency services, mobile, event notification