Communication Applications in SIP-enabled Networks – Trends and Futures

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

Sylantro (Campbell, California)

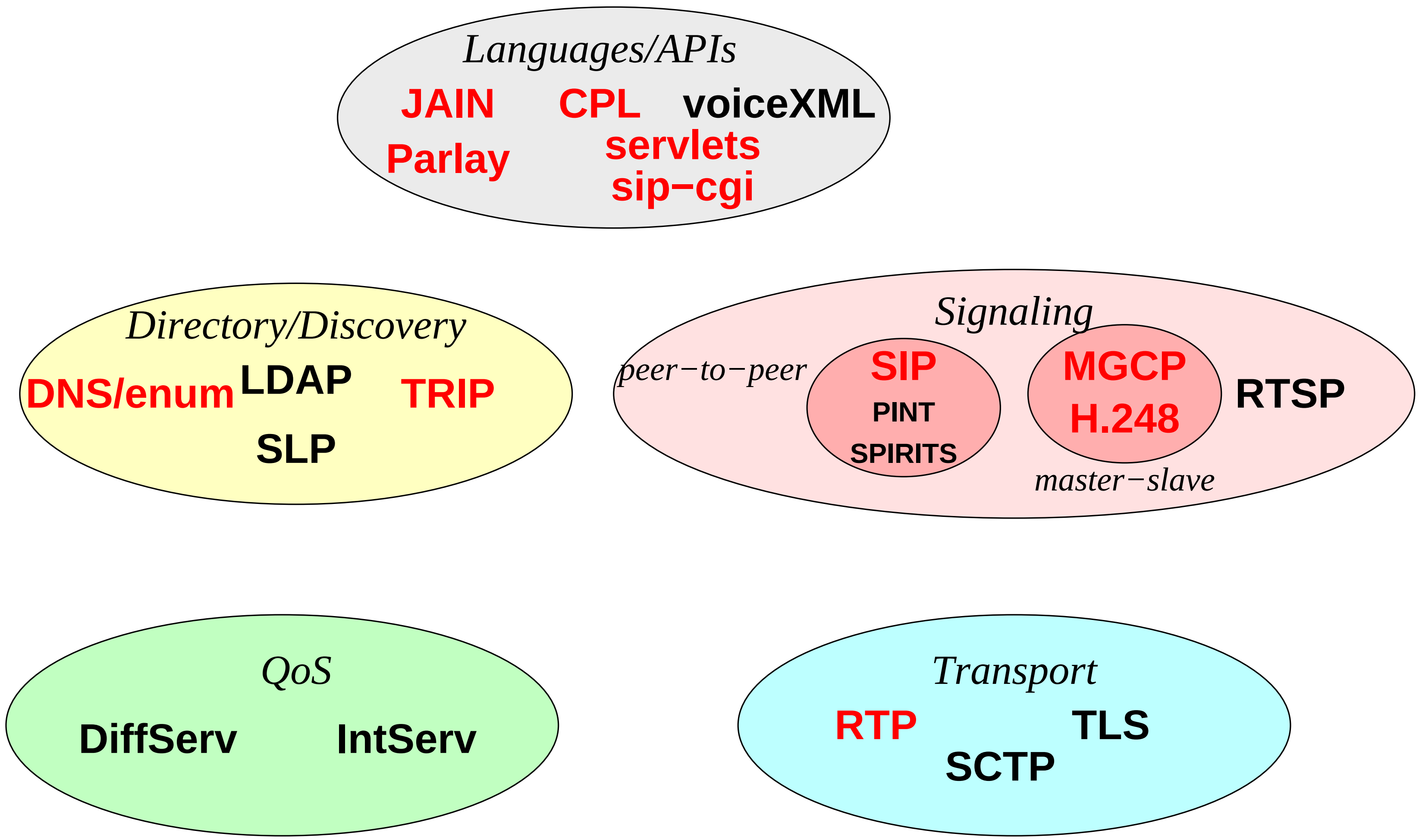
December 8, 2000

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

Overview

- 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, . . .)

IETF VoIP Protocol Architecture



Protocol “Holes”

- “tight” session control for conferences
  - admission control
  - multicast key distribution
  - advanced capability negotiation ➠ SDPng
- protocols for whiteboard, screen sharing, floor control
- 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
- SIP: specify general mechanisms, not individual services
- integration with IM and presence

SIP design choices and alternatives

Alternative	but. . .
XML instead of RFC822	space overhead? parsing?
RPC model	higher message count
More restrictive syntax	implementation creativity

What is SIP good for (and not)

Good for . . .

- messaging
- application-layer routing ➠ end-system abstraction
- low-overhead

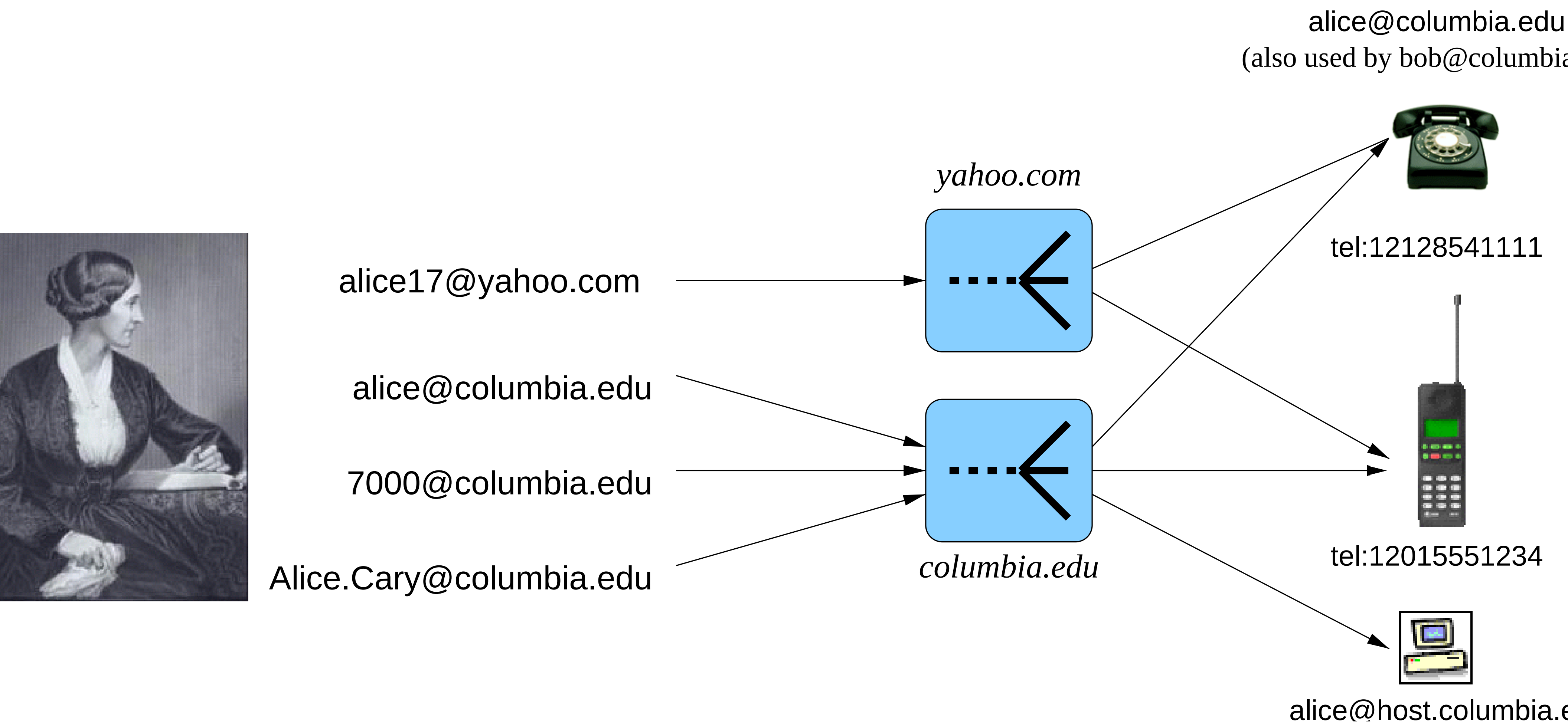
Not good for . . .

- general RPC mechanism (mostly)
- high-volume messaging (proxies)
- UDP ➠ large message bodies
- Megaco “stimulus” replacement

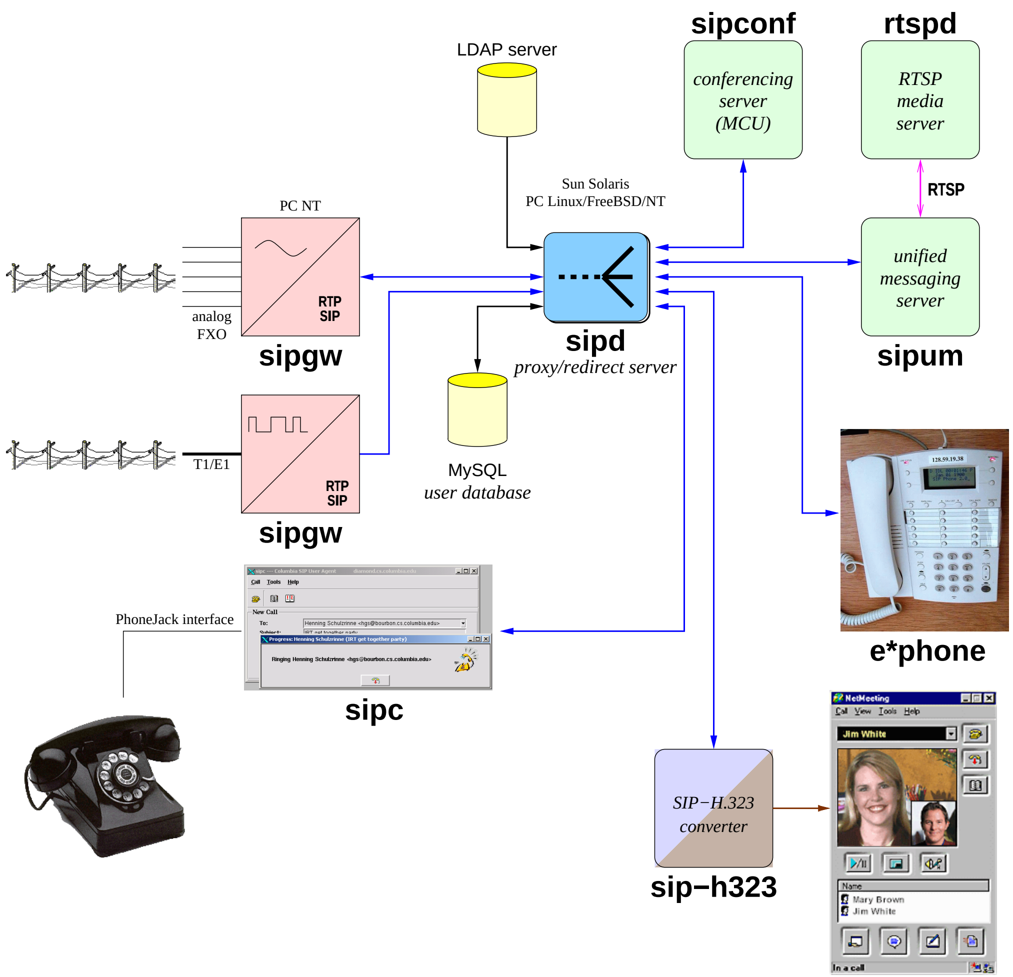
SIP mobility

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

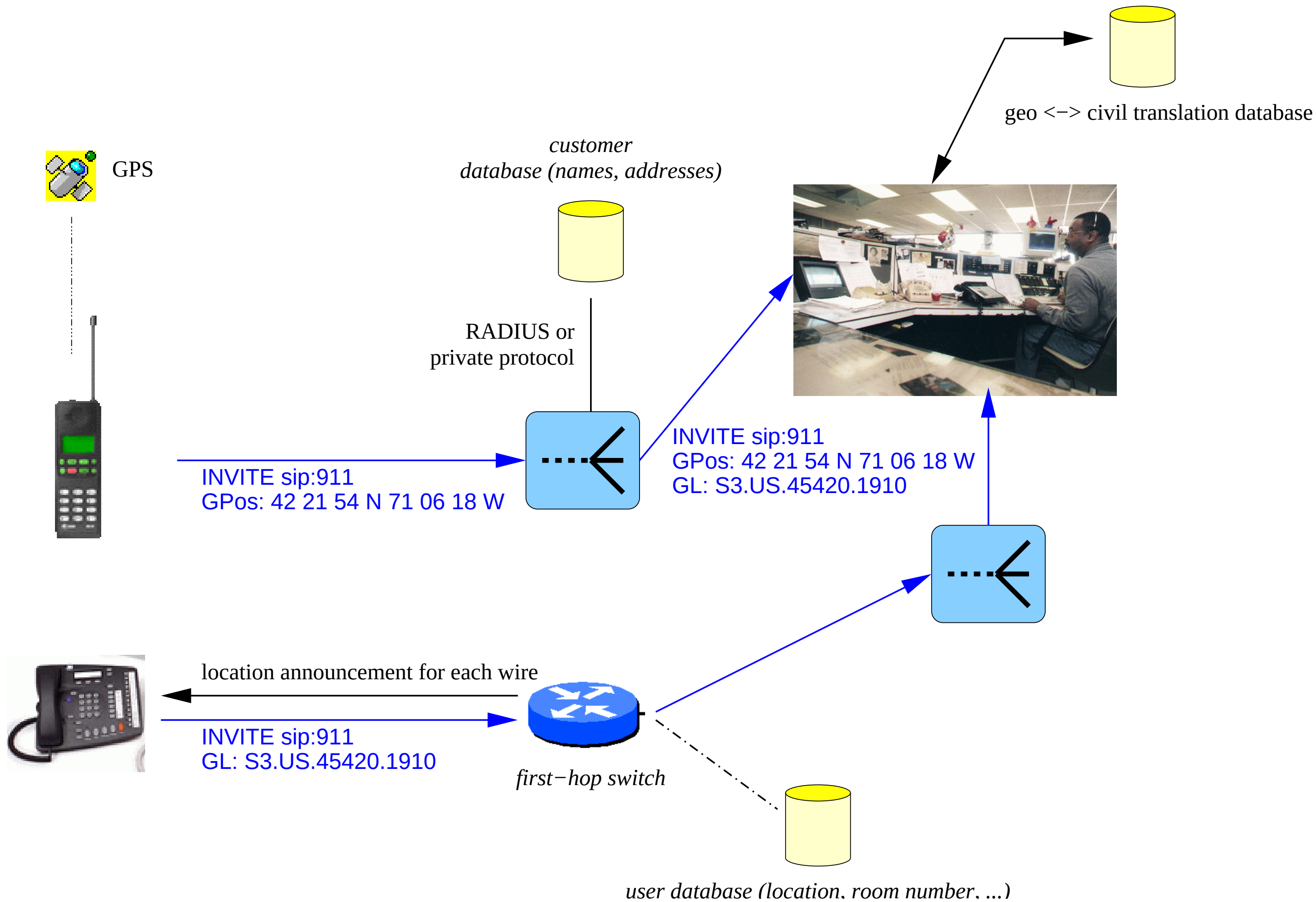
Current SIP efforts

- 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 $\surd$
- services (transfer, multiparty calls)
- ISUP carriage $\surd$

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



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!

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
- 3D worlds: proximity triggers call

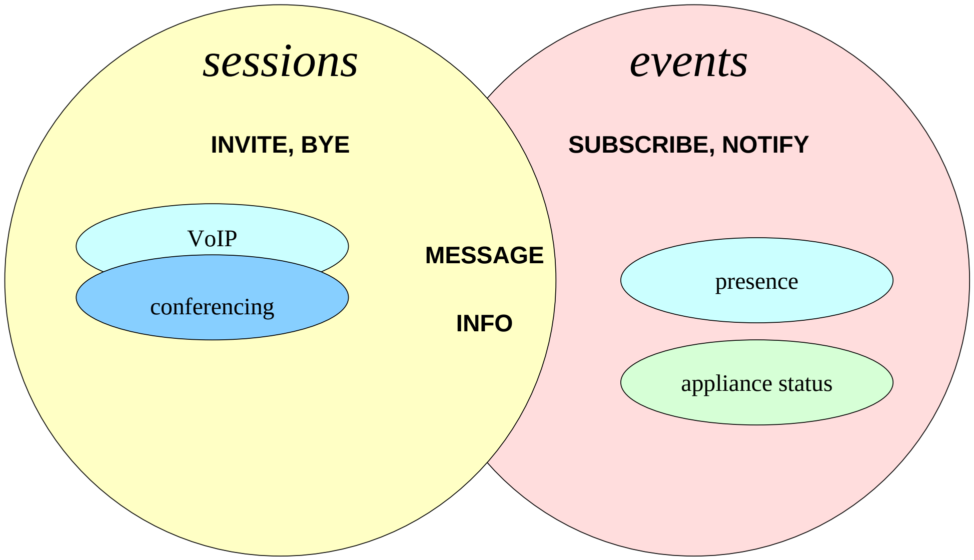
New services

- web/email-based calling
- transfer between email, web, SIP, IM, LDAP, ...
- multiple media: cross-platform screen sharing (e.g., AT&T VNC), whiteboard, games, ... (and video, too)

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”

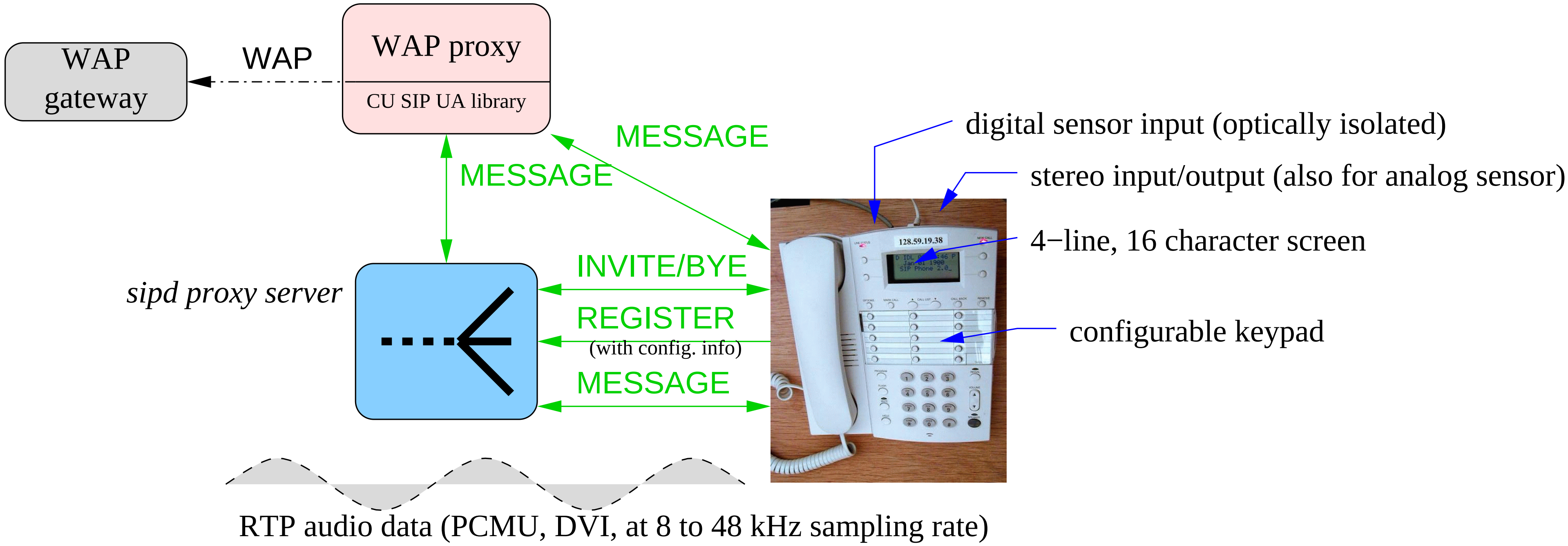
Commonalities between signaling and events

- 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

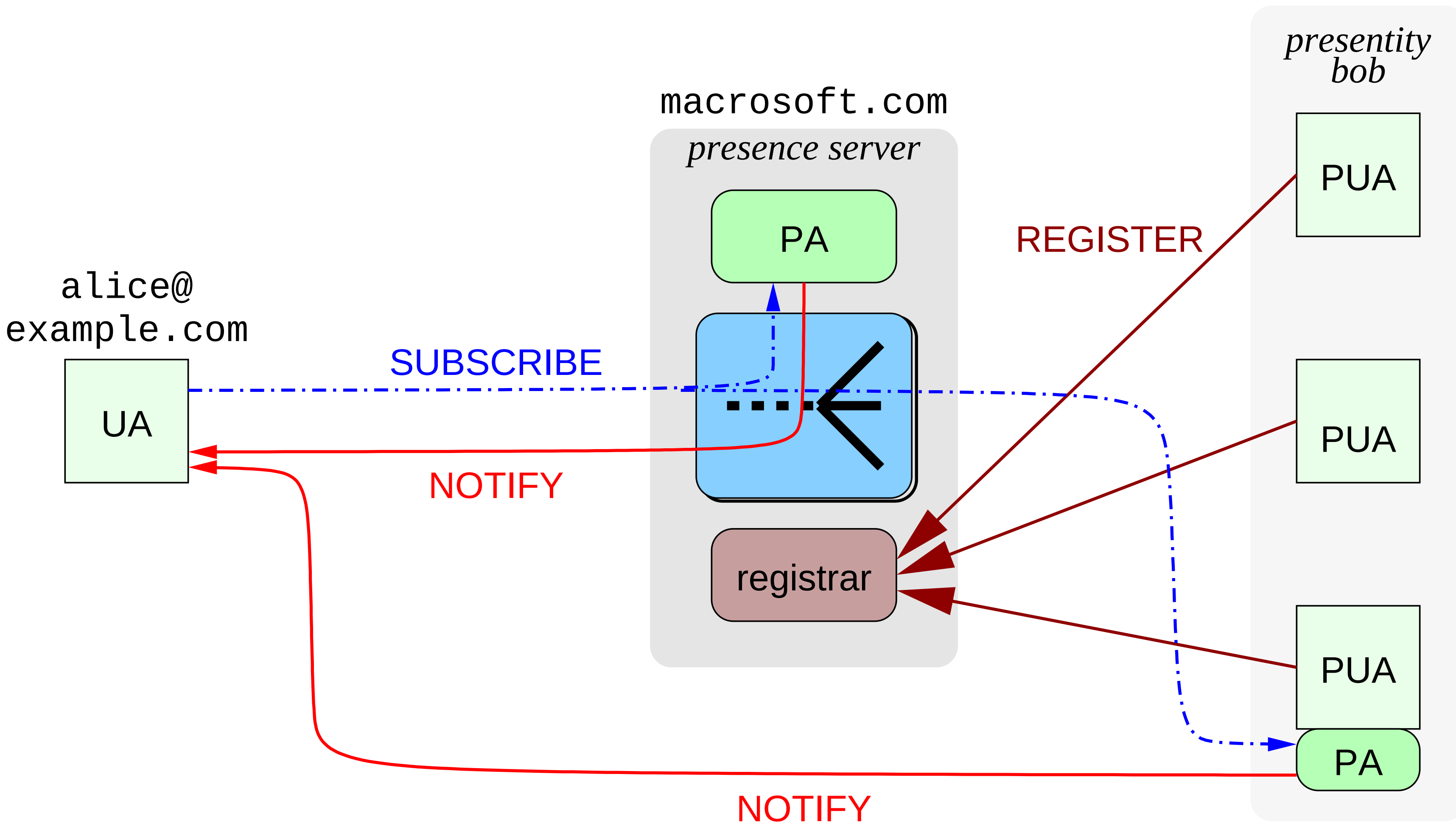
SIP as a presence & event platform

- 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 and WAP


WAP gateway
WAP
WAP proxy
CU SIP UA library
MESSAGE
MESSAGE
sipd proxy server
INVITE/BYE
REGISTER
(with config. info)
MESSAGE
digital sensor input (optically isolated)
stereo input/output (also for analog sensor)
4−line, 16 character screen
configurable keypad
RTP audio data (PCMU, DVI, at 8 to 48 kHz sampling rate)

SIP presence architecture



SIP presence components

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

Registrar: receives REGISTER requests

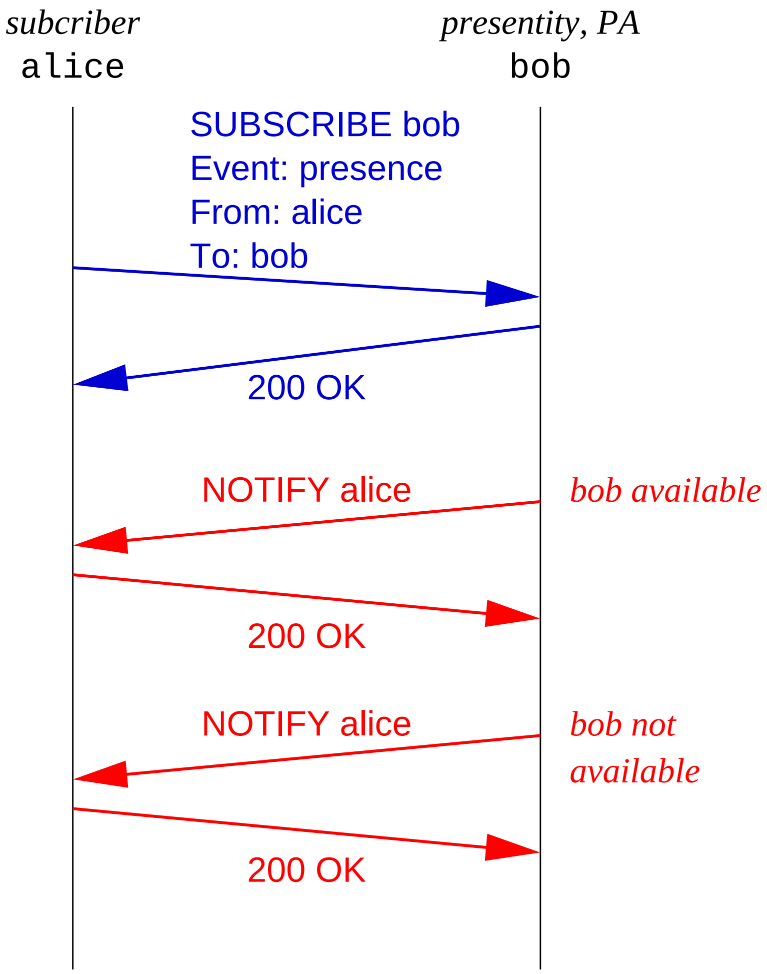
Presence user agent (PUA): generates REGISTER, but no SUBSCRIBE or NOTIFY ➠ any non-presence-aware SIP software

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
```

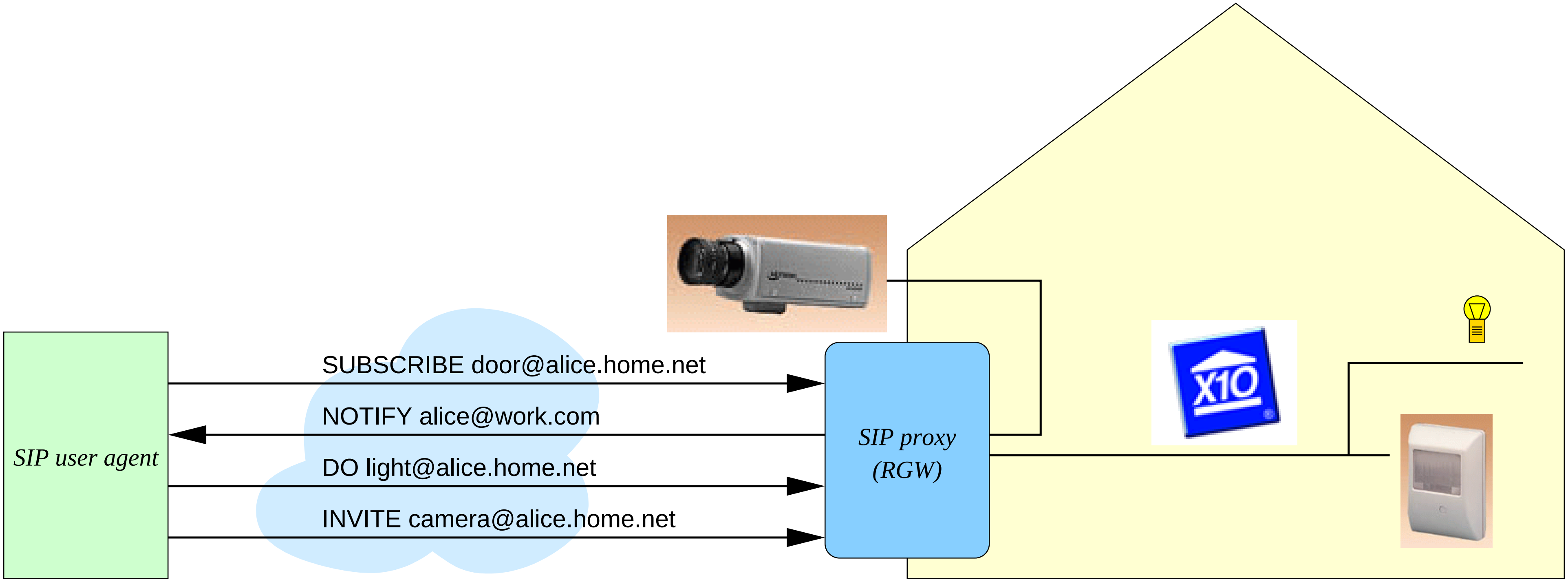
- Forked to all PUAs that have REGISTERed with method SUBSCRIBE.
- 200 (OK) response contains current state.

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

CPL textual representation

```
  <incoming>
    <address-switch field="origin" subfield="host">
      <address subdomain-of="example.com">
        <location url="sip:jones@example.com">
          <proxy>
            <busy> <sub ref="voicemail" /> </busy>
            <noanswer> <sub ref="voicemail" /> </noanswer>
            <failure> <sub ref="voicemail" /> </failure>
          </proxy>
        </location>
      </address>
      <otherwise>
        <sub ref="voicemail" />
      </otherwise>
    </address-switch>
  </incoming>
</cpl>
```

Challenges for programmable services

- integration of authentication information
- handling of SUBSCRIBE, NOTIFY
- integration of JavaScript and CPL?
- modifiable
- model for program generation: flow charts? menus?
- end-system programming: abstracted user interface?

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