# Creating Services for Internet Telephony using the Session Initiation Protocol

Henning Schulzrinne
Dept. of Computer Science
Columbia University
New York, New York
(sip:)schulzrinne@cs.columbia.edu

# Overview

- What is SIP (and not)?
- What is it good for?
- Internet telephony architectures
- SIP for VoIP services
- SIP for instant messaging and presence
- hurdles for **Internet** telephony

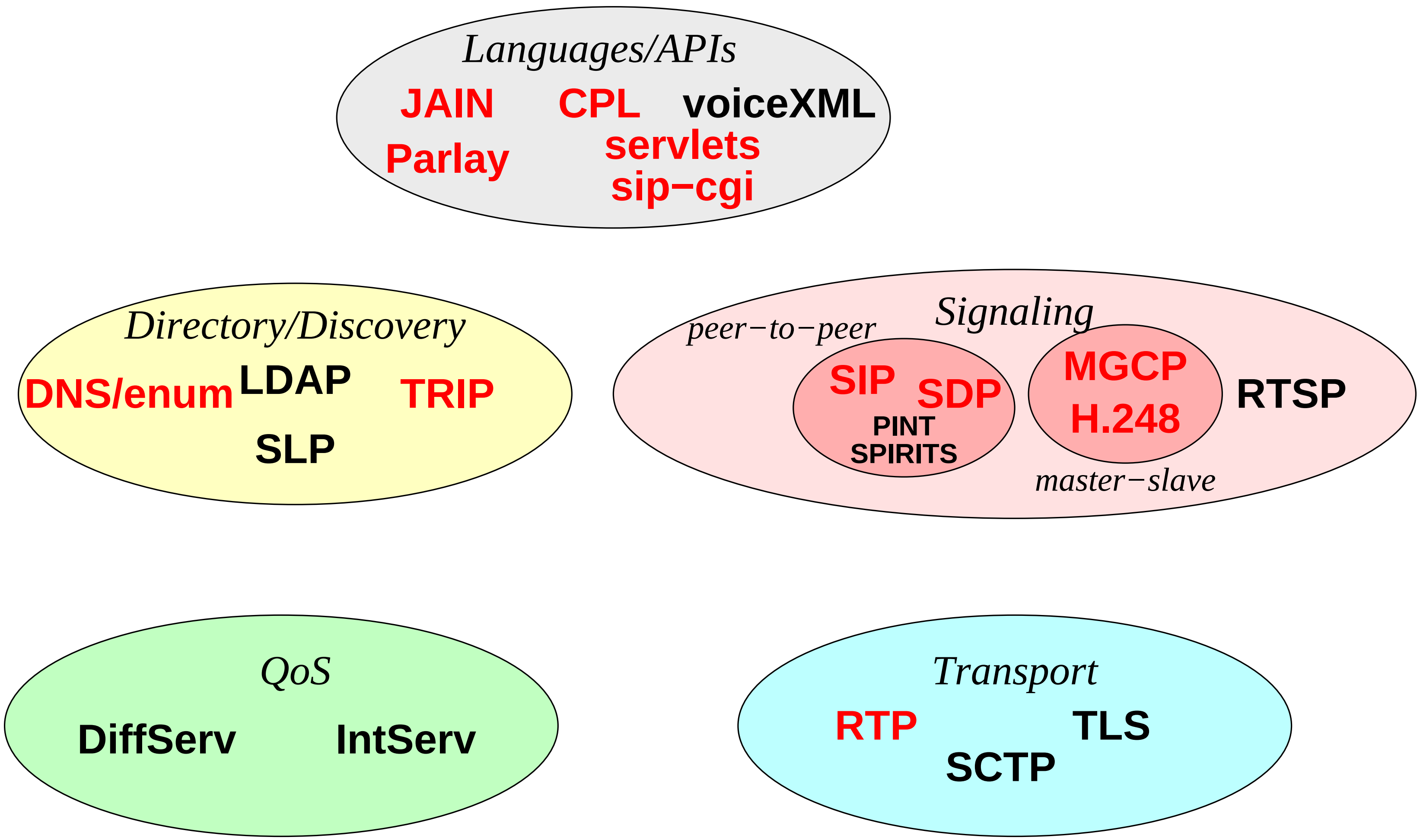
# What is SIP?

- sets up and tears down *sessions*
  - content-neutral: audio, video, shared applications, . . .
  - network-neutral: ATM, FR or IP, but mostly IP
- notifies users of *events*: “I’m online”, “person entered room”, “dishes are done”, . . .
- sends messages – instant (text) messages (“SMS”, “IM”),

# How does SIP work?

- protocol similar to HTTP
- uses either UDP or TCP
- uses URLs that identify *logical* destination, not IP address of end system

# SIP in the VoIP protocol ecosystem



# What is it not?

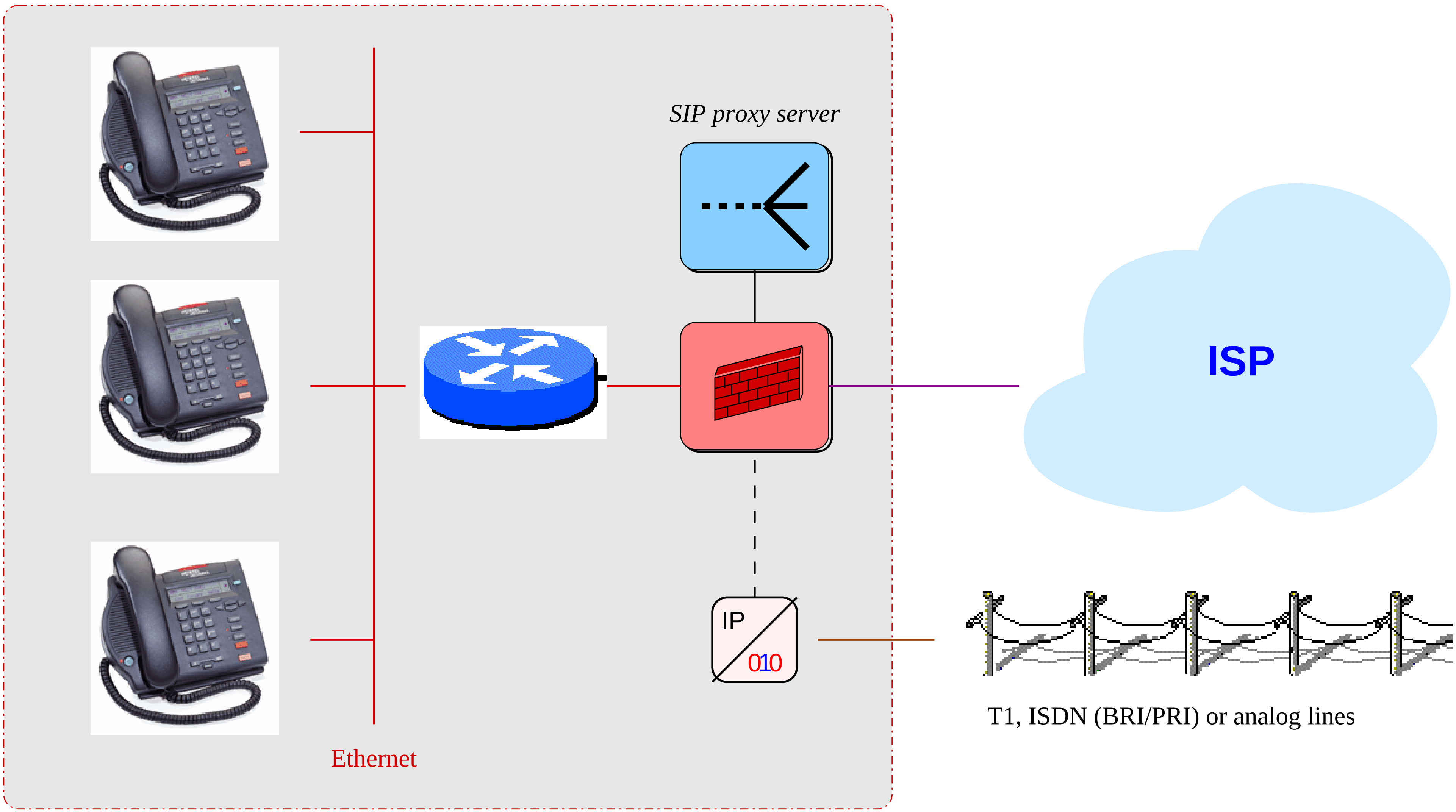
- replacement for the web (HTTP) or email
- media or data transport protocol ➠ RTP
- conference control protocol ➠ ?
- database access protocol ➠ LDAP, DNS

# Internet telephony service models

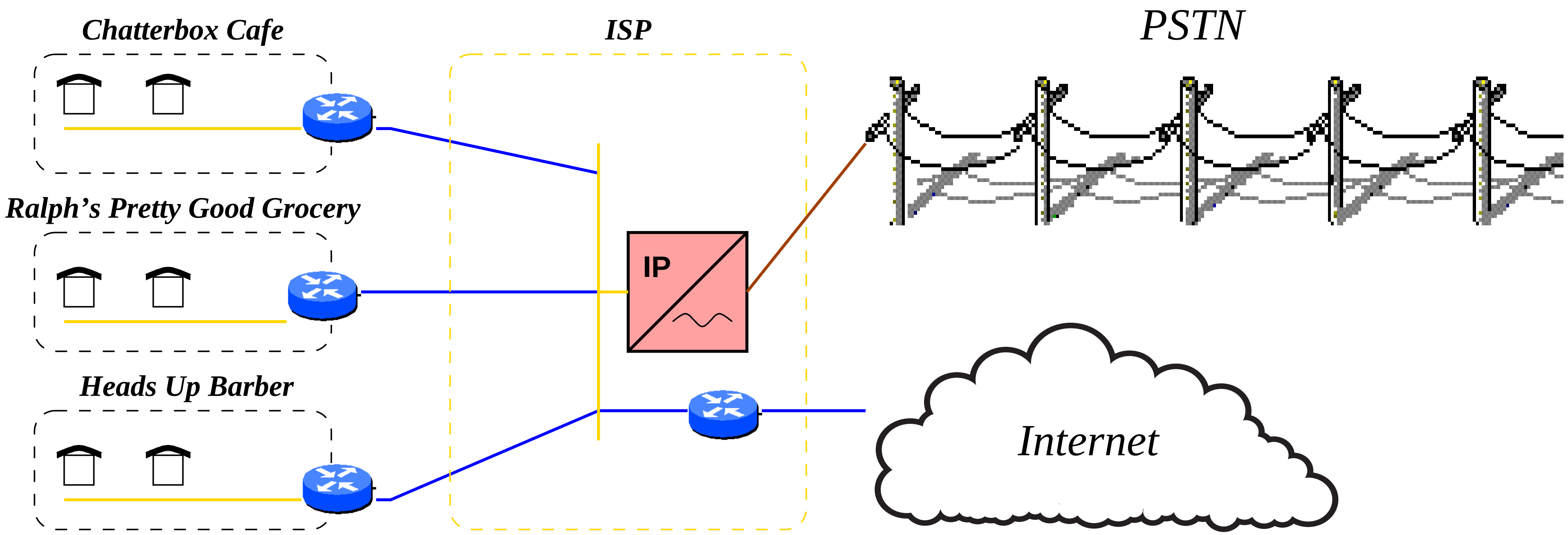
- Internet “PBX”
- Internet Centrex
- Internet Carrier

➠ same basic equipment, but size of gateway varies

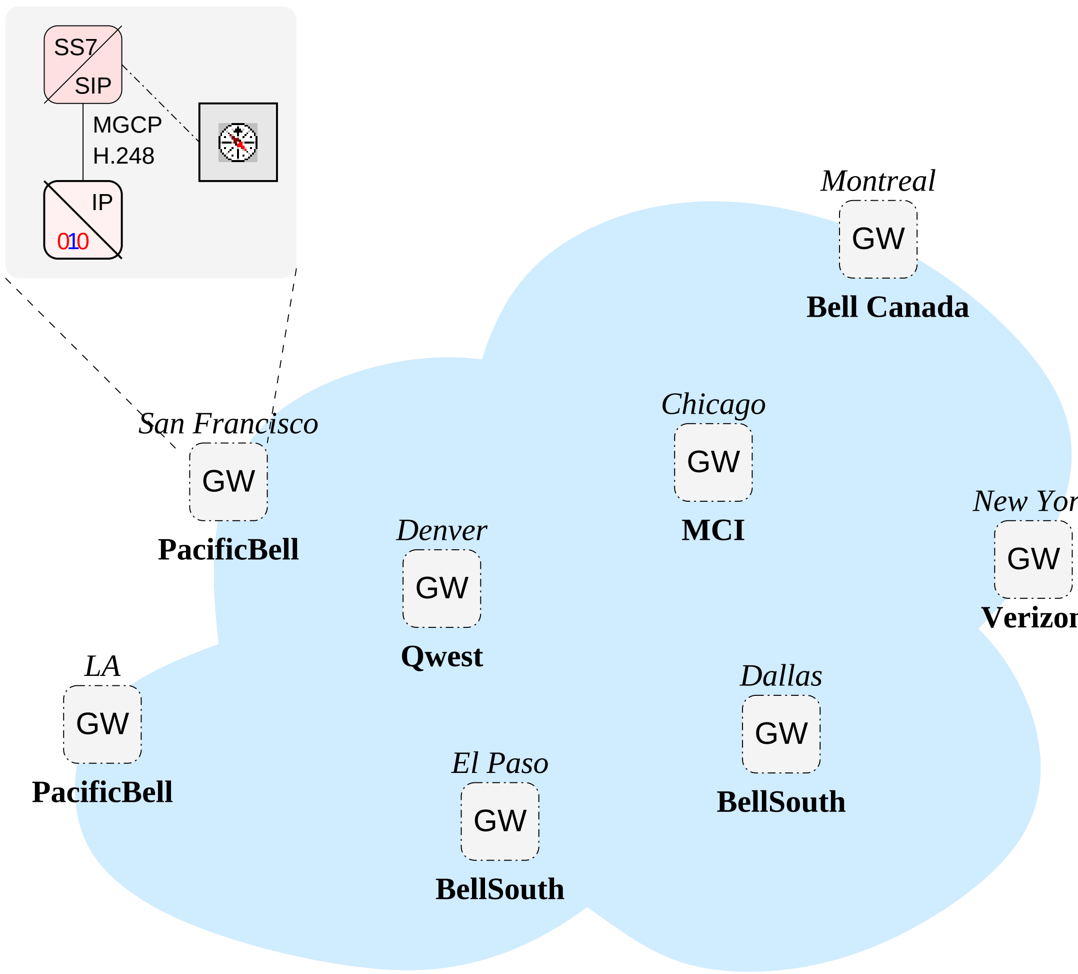
# Internet PBX



# 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
San Francisco
GW
PacificBell
Chicago
GW
MCI
New York
GW
Verizon
Denver
GW
Qwest
LA
GW
PacificBell
Dallas
GW
BellSouth
El Paso
GW
BellSouth

# What is SIP good for?

- replicate functionality of traditional PSTN services:
  - caller id
  - call forwarding
  - call transfer
  - 800/900# services
  - find me/follow me
  - conference calls
- create new services:
  - Internet integration
  - programmable services
  - multi-destination routing
  - multimedia
  - event notification

# New SIP services: Internet integration

- typically, SIP URL $\equiv$ email address, e.g., `sip:joe@net2phone.com` or `tel:+1201-555-1212`
- URLs everywhere:
  - forward calls to email
  - forward calls to web page
  - forward calls to recordings
  - pager, cell phone numbers
  - IM addresses
- SIP messages can contain HTML and other web objects:
  - menu pops up when calling restaurant
  - error messages: "not here, but please choose from ..."
  - visual caller id – photos of callee

# New SIP services: programmable services

- three sources of services:

  **Vendor:** program into software ➠ efficient, robust, but long cycles, inflexible

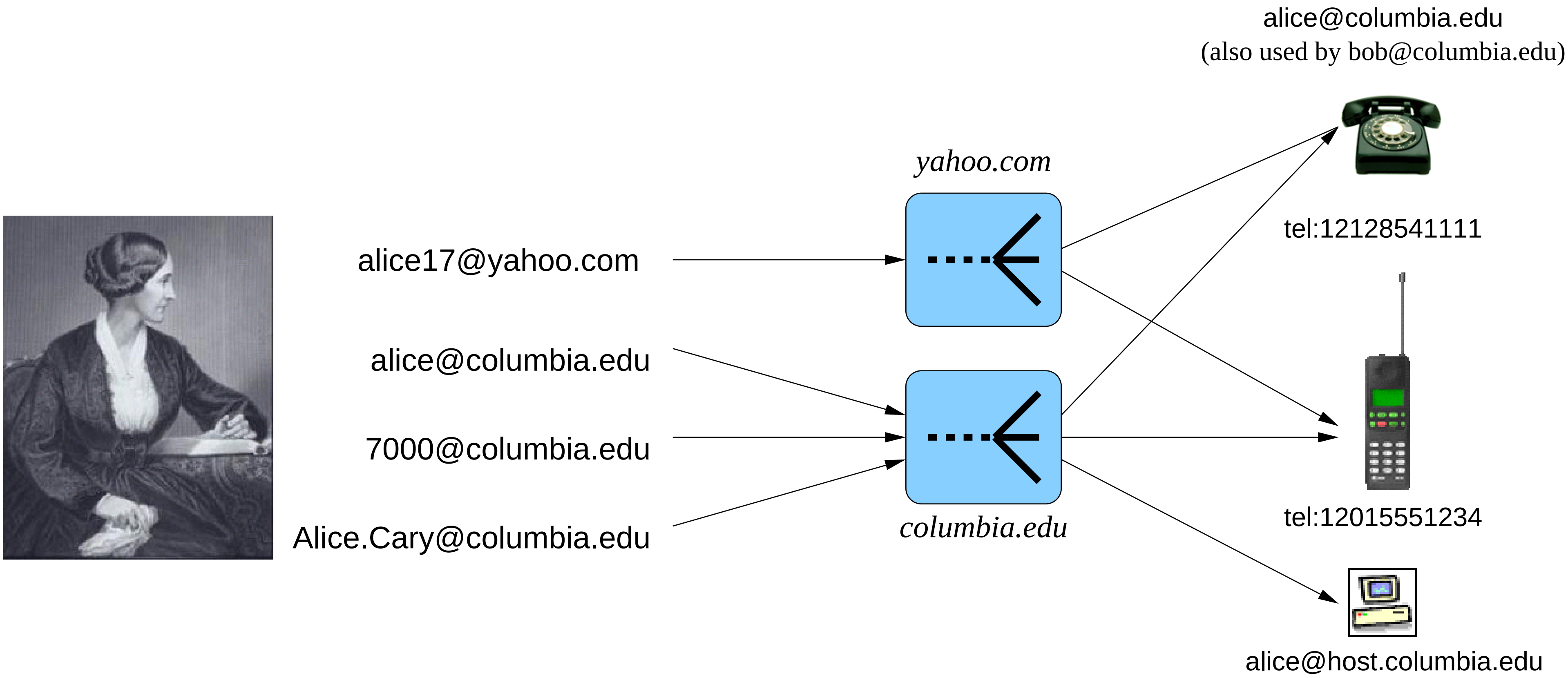
  **Service provider:** differentiation, vertical markets, but limited set

  **User:** customized and personalized

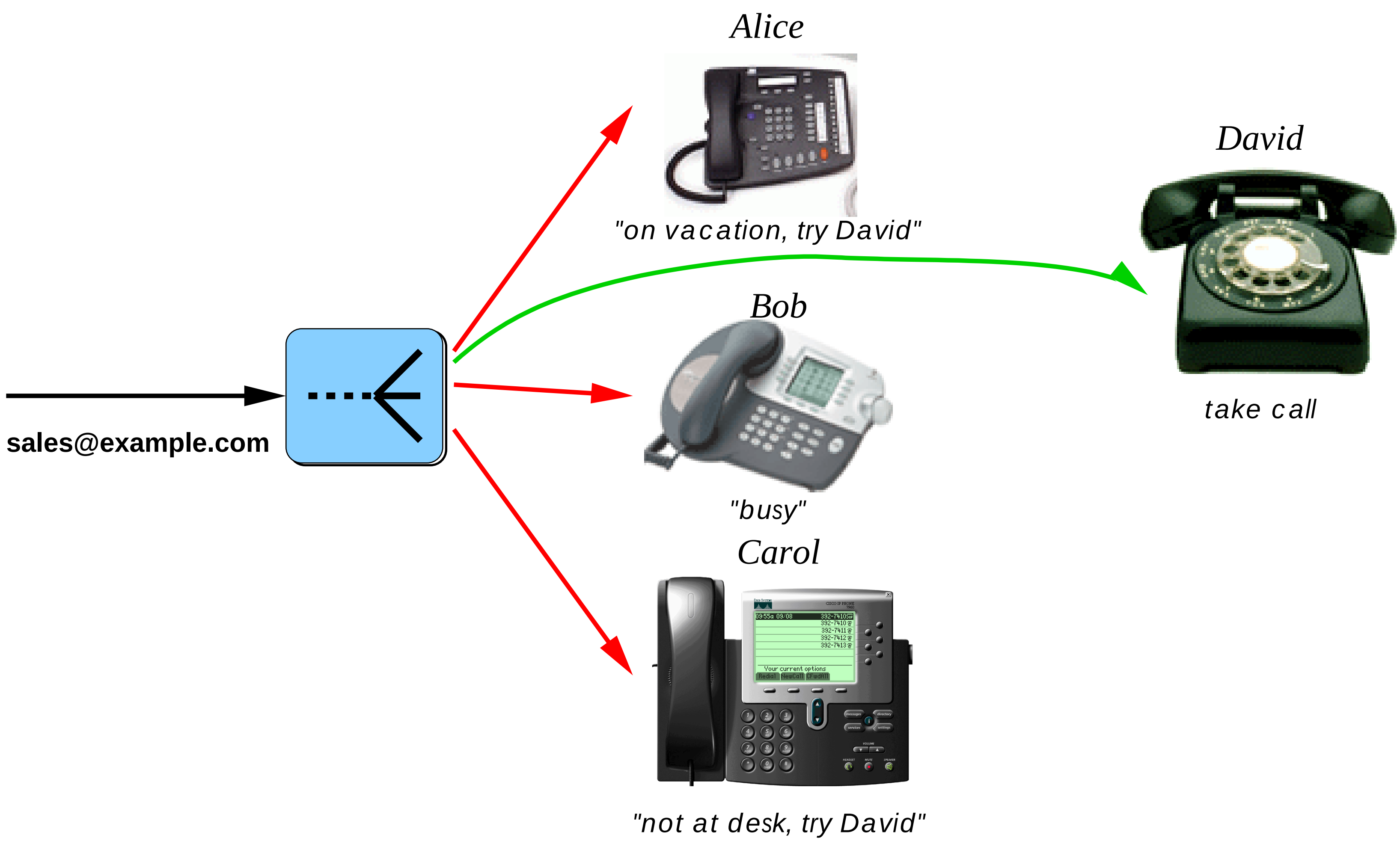
# SIP mobility

| | |
|---|---|
| terminal | cross-provider |
| personal | different terminals, same address |
| service | different terminals, same services |
| session | move sessions across terminals |

# SIP personal mobility



# New SIP services: multi-destination routing



# New SIP services: event notification

- many telecom services are really events:
  - voicemail notification
  - call supervision
  - automated call back
  - call waiting
- generalizes to
  - physical events: “water in basement”
  - communication events: “email has arrived”
  - network events: “print job is done”

# What kind of SIP products are emerging?

**SIP libraries:** for building end systems

**SIP “clients”:** also known as user agents; PC-applications

**SIP proxy servers:** call routing and applications

**SIP unified messaging servers:** record voice calls

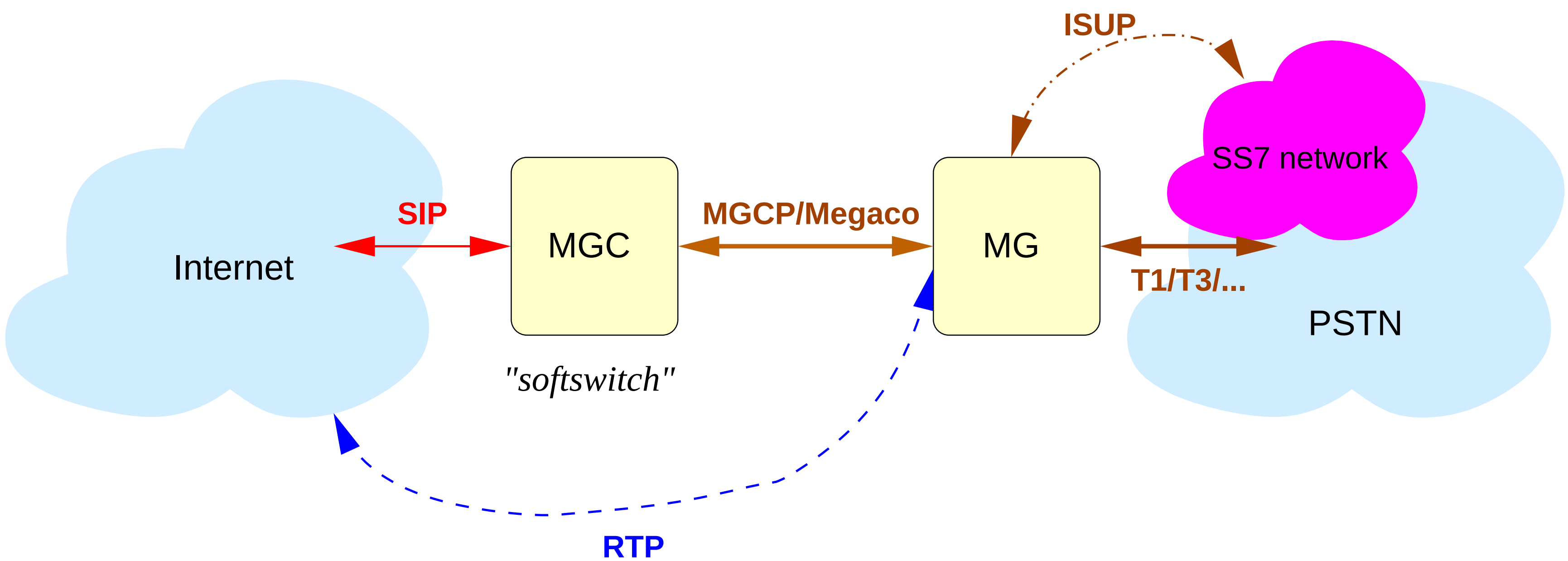
**SIP conference servers:** multipoint control units

**SIP testers:** debug applications, load testing

**SIP-enabled firewalls:** get voice through firewalls

# Marketing terms: softswitches

Commonly used in marketing, but pretty vague:




- Software version of class-4/class-5 switch?
- doesn't really "switch" voice

# Marketing term: application server

- supports Internet telephony applications
- typically, programmable:
  - APIs, such as JAIN and Parlay
  - Java servlets
  - Call Processing Language
- may be able to initiate calls or just route calls

# SIP-enabled networks

- Chunghwa Telecom, Taipei
- Level 3
- MCI Worldcom
- VONage

Others to follow soon.

# Example: Pingtel SIP phone

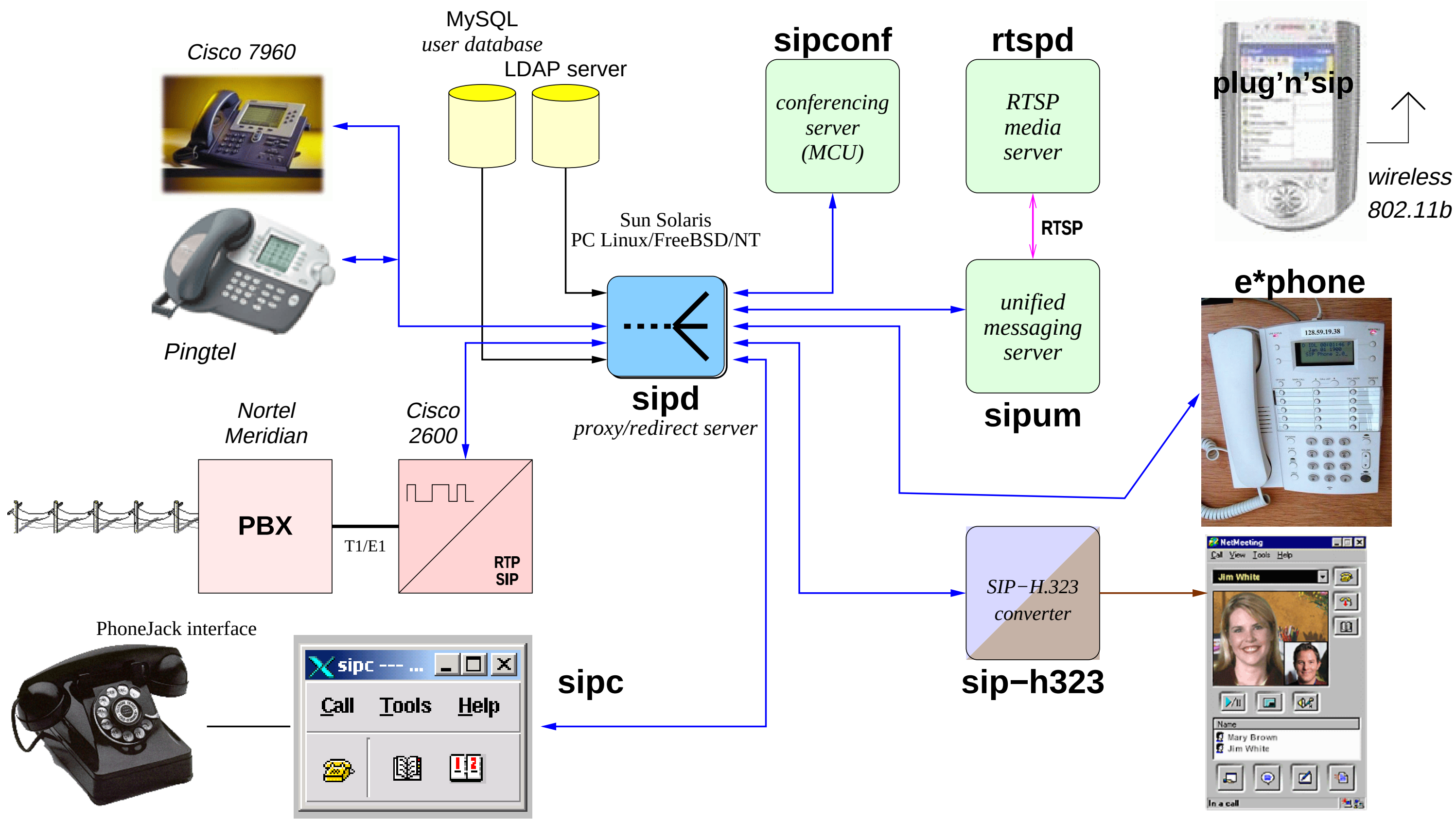
# Example: Cisco and 3Com SIP phones

Cisco

3Com ($395 list)

# Example: Columbia CS Phone System

Expand existing PBX via IP phones, with transparent connectivity



# Status of SIP in the market

- all major manufacturers of telephony-related equipment appear to be working on SIP
- generally, first-generation products
- not yet widely available as consumer products
- but largely interoperable
- use of SIP in Windows XP (MS Messenger) will accelerate uptake

# Differences: Internet Telephony $\leftrightarrow$ 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” $\longrightarrow$ 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

# 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

# Example cgi script: priority

```
#! /usr/bin/env perl -w

# Prioritize messages whose 'From:' matches 'sip:hgs@' by proxying them
# with 'Priority: urgent'.

# Translate the REGISTRATIONS env variable into a list of
# registration addresses, without name-addr forms or parameters.
sub get_regs {
    my($reg_str, @regs);

    if (!defined($ENV{REGISTRATIONS})) { return (); }

    $reg_str = $ENV{REGISTRATIONS};
    ...
    @regs = split(",", $reg_str);

    grep {
        # Eliminate parameters, then strip <> forms.
        s/;.*//;
        if (/\<(.*)\>/) { $_ = $1; }
    } @regs;

    return @regs;
}
```

```
if (defined $ENV{SIP_FROM} && $ENV{SIP_FROM} =~ /sip:hgs@/) {
    foreach $reg (get_regs()) {
        print "CGI-PROXY-REQUEST $reg SIP/2.0\n";
        print "Priority: urgent\n\n";
    }
}
```

# Example CPL script: lookup

```
<?xml version="1.0" ?>
<!DOCTYPE call SYSTEM "cpl.dtd">

<cpl>
  <incoming>
    <lookup source="http://www.example.com/cgi-bin/locate.cgi?user=jones"
            timeout="8">
      <success>
        <proxy />
      </success>
      <failure>
        <mail url="mailto:jones@example.com&Subject=lookup%20failed" />
      </failure>
    </lookup>
  </incoming>
</cpl>
```

# Potential stumbling blocks

**NATs:** VoIP violates new Internet “architecture” of TCP client-server ➠

- application-layer gateways
- avoid unnecessary NATs (get addresses)
- IPv6

**Firwewalls:** Out-of-band = control + data ➠ need modifications

**Walled-garden:** only carrier-approved services

- escalating attempts to try to prevent people from “bypassing” services
- probably violates common-carrier status
- WAP as warning

# Where is SIP being defined?

| | |
|---|---|
| IETF (Internet Engineering Task Force) | SIP core and extensions |
| 3GPP (3rd Generation Partnership) | mobile networks (SIP for signaling) |
| Softswitch Consortium | profiles for soft switches |
| PacketCable | profiles for cable modems |
| SIP Forum | evangelism, operational guidelines |

# For more information...

**SIP:** http://www.cs.columbia.edu/sip

**SDP:** http://www.cs.columbia.edu/˜hgs/internet/sdp.html

**RTP:** http://www.cs.columbia.edu/˜hgs/rtp

**Papers:** http://www.cs.columbia.edu/IRT