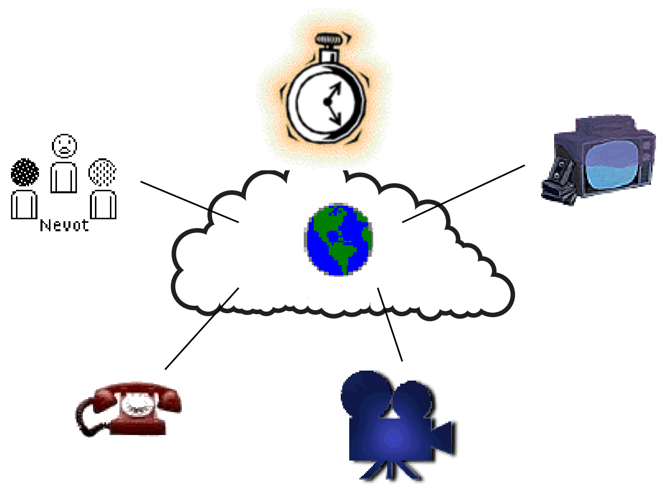
Internet Real-Time Laboratory

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

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

IBM – October 12, 1999



IRT Research Topics

- network infrastructure
  - scalable resource reservation protocols
  - QOS measurements and management
  - adaptive services and pricing
- Internet telephony (VoIP)
  - signaling: SIP, translation to ISUP/H.323/MGCP
  - VoIP services and service creation
  - Internet voice mail
  - e*phone Internet "appliance"
- applications
  - "civil society" computing
  - Internet TV and radio
  - distance learning

IRT Members: Full-Time PhD Students

Student	CU	IRT	research topic
Jonathan Lennox	F1994	S1997	call processing language
Wenyu Jiang	F1995	F1998	Internet QOS measurements
Xin Wang	F1996	F1996	multicast, QOS, pricing
Maria Papadopouli	F1996	F1997	ad-hoc networks
Fatima Al-Garawi	F1997	F1997	distance learning
Weibin Zhao	F1997	F1997	directory services
Xiaotao Wu	S2000		service creation

IRT Members: Part-Time PhD students

Student	employer	research topic
Lisa Amini	IBM	video on demand
Mandis Beigi	IBM	resource reservation
Ethen. Bommaiah	Bell Labs	media on demand
Ashutosh Dutta	Telcordia	MarconiNet
Christopher Kang	Bell Atlantic	signaling translation
Santosh Krishnan	Bell Labs	scheduling, QOS
Ping Pan	Bell Labs	resource reservation
Jonathan Rosenberg	Bell Labs	Internet telephony

IRT Members: Masters Students and Visitors

Gautam Nair		1999–	Internet telephony
Kundan Singh		1999–	Internet telephony
Jianqi Yin	BUPT, Beijing	1997–1999	Internet telephony
Elin Wedlund	Ericsson, Sweden	1998–1999	mobility
Tony Eyers	U. Wollongong	1999	signaling performance

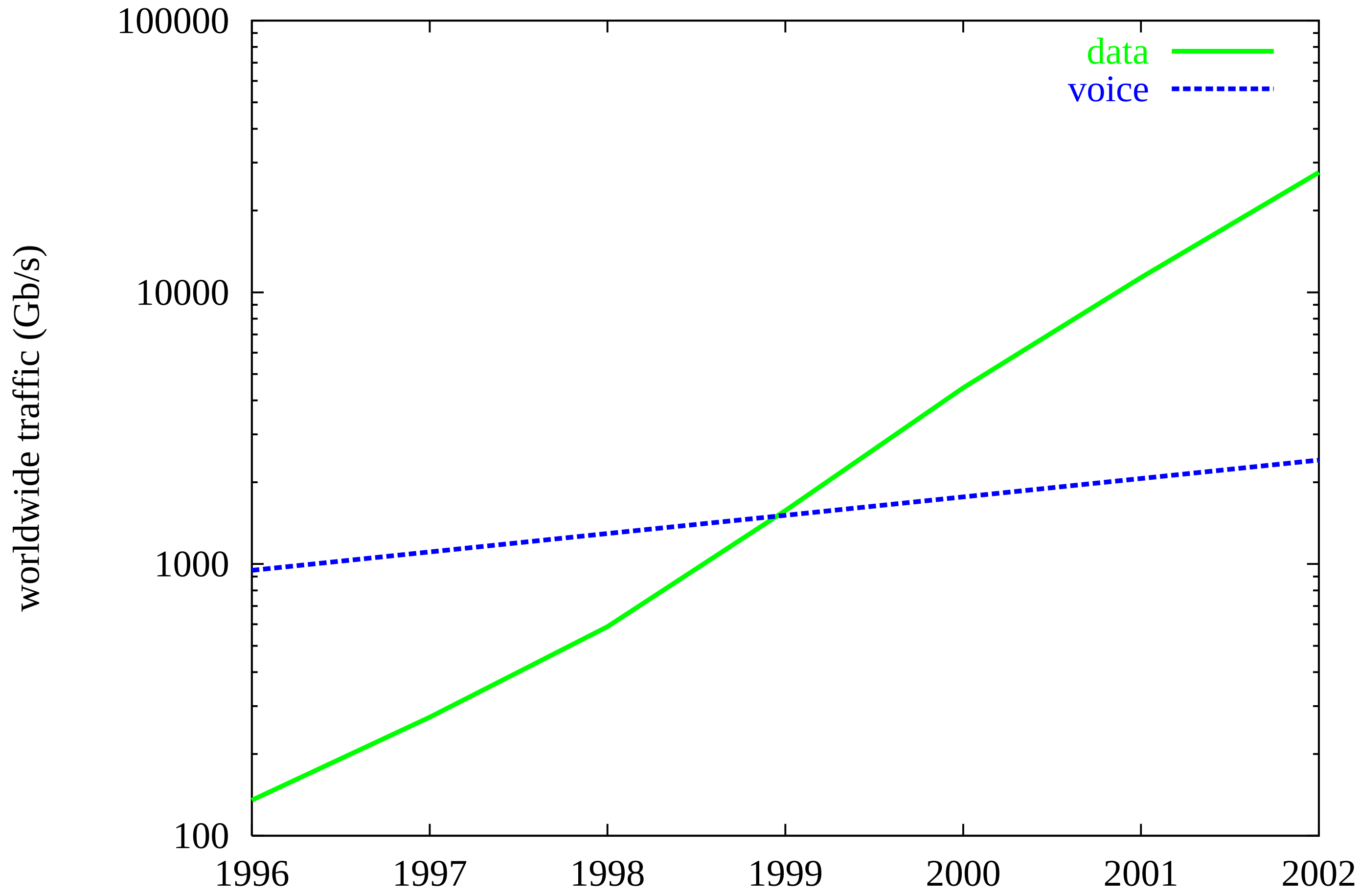
Internet Telephony and Multimedia

- Internet telephony: motivation and problems
- protocol architecture
- quality of service:
  - light-weight resource reservation
  - forward error control
- services ➠ signaling
- the “programmable” telephone
- Internet telephony “appliances”
- mobile services

Historical perspective

1876	invention of telephone
1915	first transcontinental telephone (NY–SF)
1920’s	first automatic switches
1956	TAT-1 transatlantic cable (35 lines)
1962	digital transmission (T1)
1965	1ESS analog switch
1977	4ESS digital switch
1980s	Signaling System #7 (out-of-band)

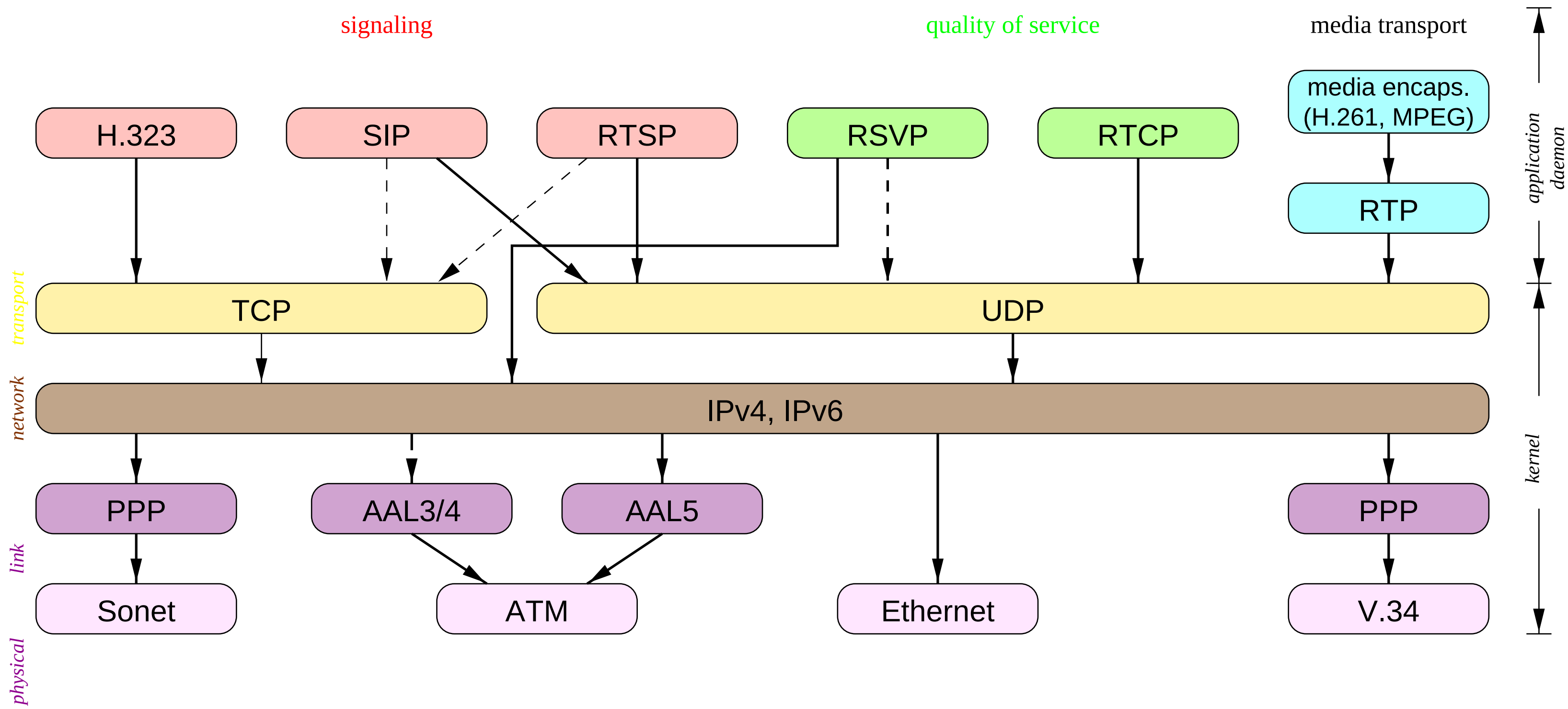
Data vs. Voice Traffic



The phone works — why bother with VoIP?

user perspective	carrier perspective
• variable compression: tin can to broadcast quality	• silence suppression ➠ traffic ↓
• security through encryption	• shared facilities ➠ management, redundancy
• caller, talker identification	• advanced services (email/web integration)
• better user interface (browser)	• cheaper switching ($0.005 vs. $5/kb/s)
• internat. calls: TAT transatlantic cable = $0.03/hr	• 9.6 kb/s fax as data
• no local access fees (3.4c)	
• easy: video, whiteboard, ...	

Internet multimedia protocol stack



YESSIR: Yet another Sender Session Internet Reservation

- RSVP: separate daemon, API
- ➠ integrate into application that needs it (embedded systems!)
- in-band ➠ easier firewall
- RTP: common data transport protocol for audio/video
- router alert option in RTCP packets
- resource demands: payload type, measurement, flow specs, …
- soft-state + RTCP BYE
- partial reservations: add links as session ages ↔ fragmentation

YESSIR

plain RTCP SRs or additional information:

IP Header with Router-Alert Option

UDP Header

RTCP message:

Sender Report:
- sender information
- detailed report for each source

YESSIR message:
- reservation command: active/passive
- reservation style, refresh interval
- reservation flow specification
- link resource collection
- reservation failure report

Profile-specific extensions

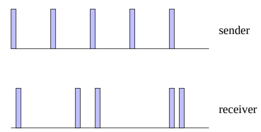
end-to-end refresh (vs. hop-by-hop)

RSVP and YESSIR performance

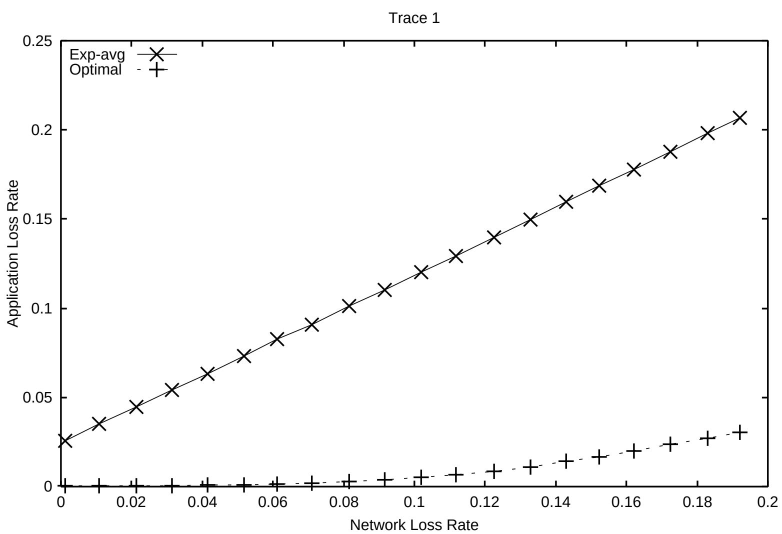
	setup	refresh
	μs	
RSVP	1,105	624
YESSIR	356	344

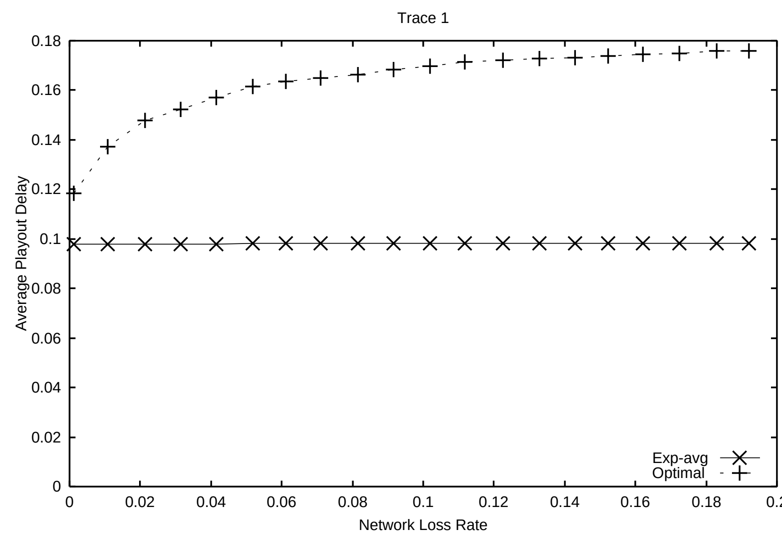
Integrating packet FEC into adaptive voice playout buffers

- playout buffer: trade loss (2…20%) for delay (50…500 ms)



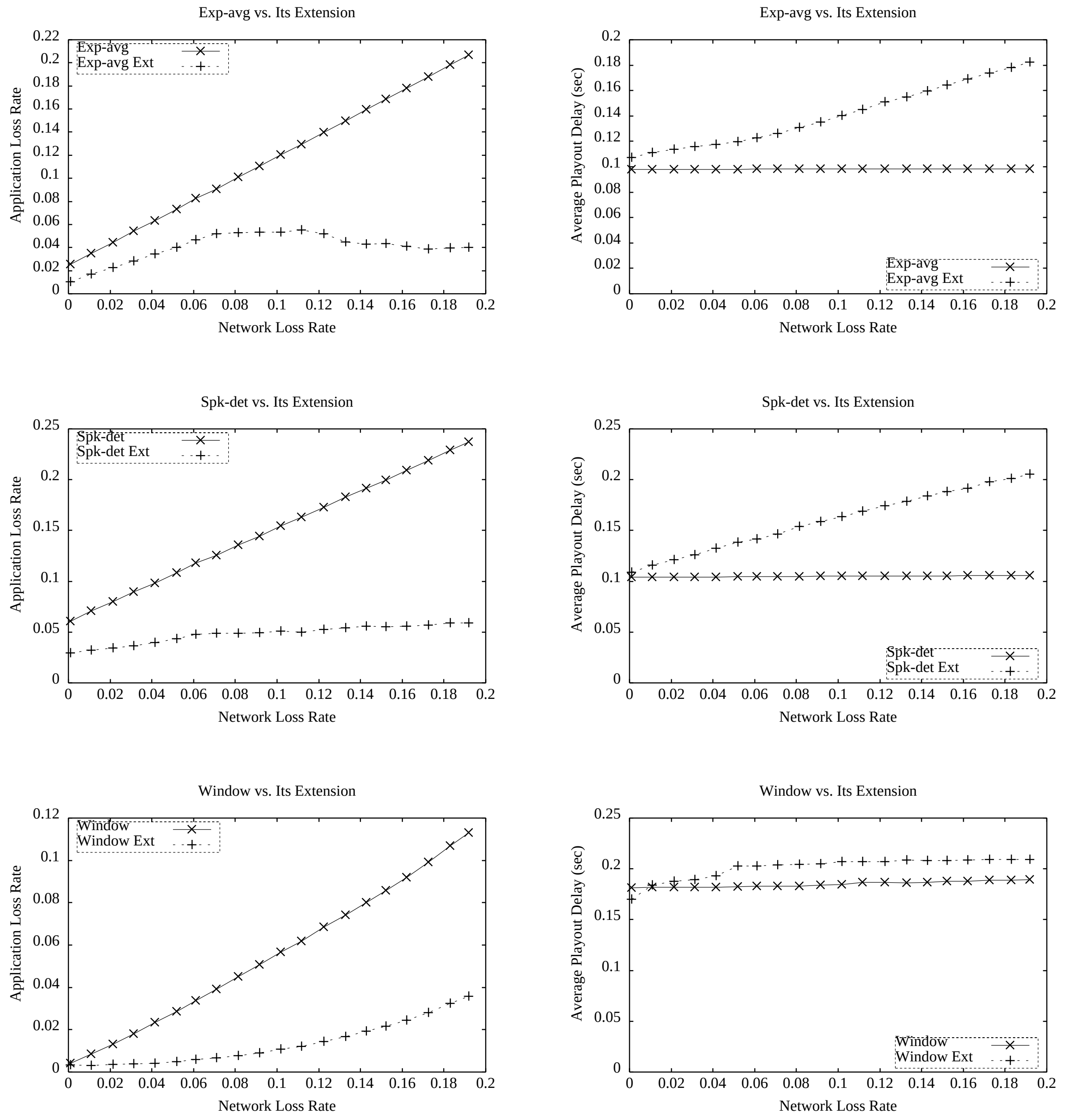
- (n, k) FEC: add $n - k$ additional packets for total of n





FEC: virtual delay

- virtual delay = min(arrival, recovery) $-$ departure
- playout delay $\approx \alpha \cdot \sigma$
- if loss $<$ target loss, $\alpha \leftarrow \alpha + \delta$
- recover lost and **late** packets
- 20% loss: application loss/5, delay * 2


Exp-avg vs. Its Extension
Application Loss Rate
Network Loss Rate
Exp-avg
Exp-avg Ext
Exp-avg vs. Its Extension
Average Playout Delay (sec)
Network Loss Rate
Exp-avg
Exp-avg Ext
Spk-det vs. Its Extension
Application Loss Rate
Network Loss Rate
Spk-det
Spk-det Ext
Spk-det vs. Its Extension
Average Playout Delay (sec)
Network Loss Rate
Spk-det
Spk-det Ext
Window vs. Its Extension
Application Loss Rate
Network Loss Rate
Window
Window Ext
Window vs. Its Extension
Average Playout Delay (sec)
Network Loss Rate
Window
Window Ext

New playout algorithm: delayed optimal

- after talkspurt, one knows optimal delay
- use optimal combination for next talkspurt
- $D_i = \alpha D_{i-1} + (1 - \alpha) D_{\mathrm{opt}}$
- user perception function: minimal delay that achieves loss target

New playout algorithm: Analytical

- assume independent loss p, delayed randomly d
- compute playout probability
- tabulate delay distribution histogram
- at end of talkspurt, find matching playout delay for target loss rate

SIP: Session Initiation Protocol

- call user
- re-negotiate call parameters
- manual and automatic forwarding
- call center: reach first (load distribution) or reach all (department conference)
- *personal mobility* (complements data link/IP mobility) ➠ change of terminal (PC, digital cordless, palmtop), location
- “forking” of calls
- terminate and transfer calls
- web security, cookies

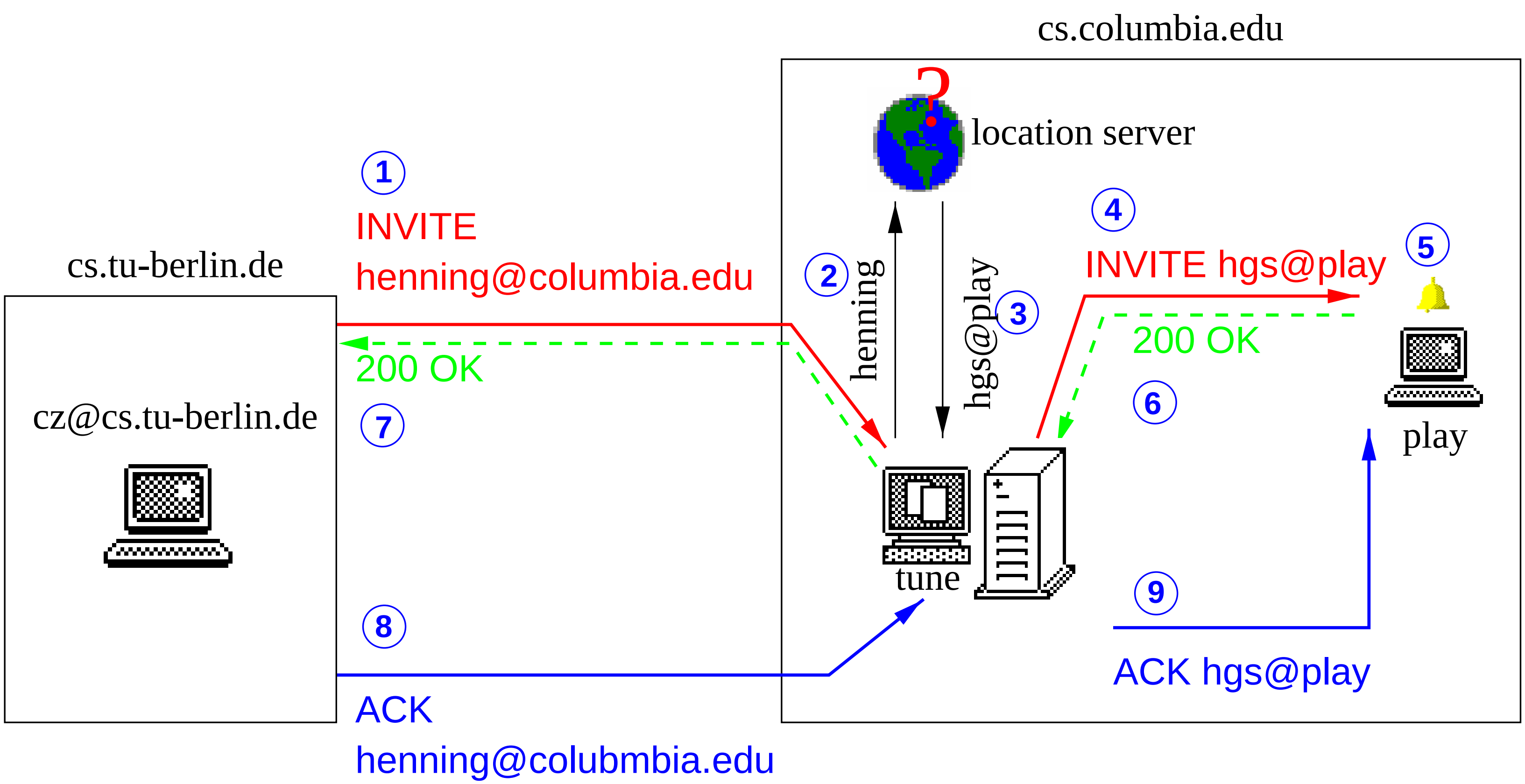
SIP addresses food chain

yellow pages		"president of the United States"
		↓ *WWW search engines*
common names		"Bill Clinton, Whitehouse"
		↓ *directory services*
host-independent		president@whitehouse.gov
	↙ *SIP*	↘ *SIP*
host-specific	sip:bubba@oval.eop.gov	sip:+1-202-456-1111@net2ph.com
	↓ *DNS*	
IP address	198.137.241.30	

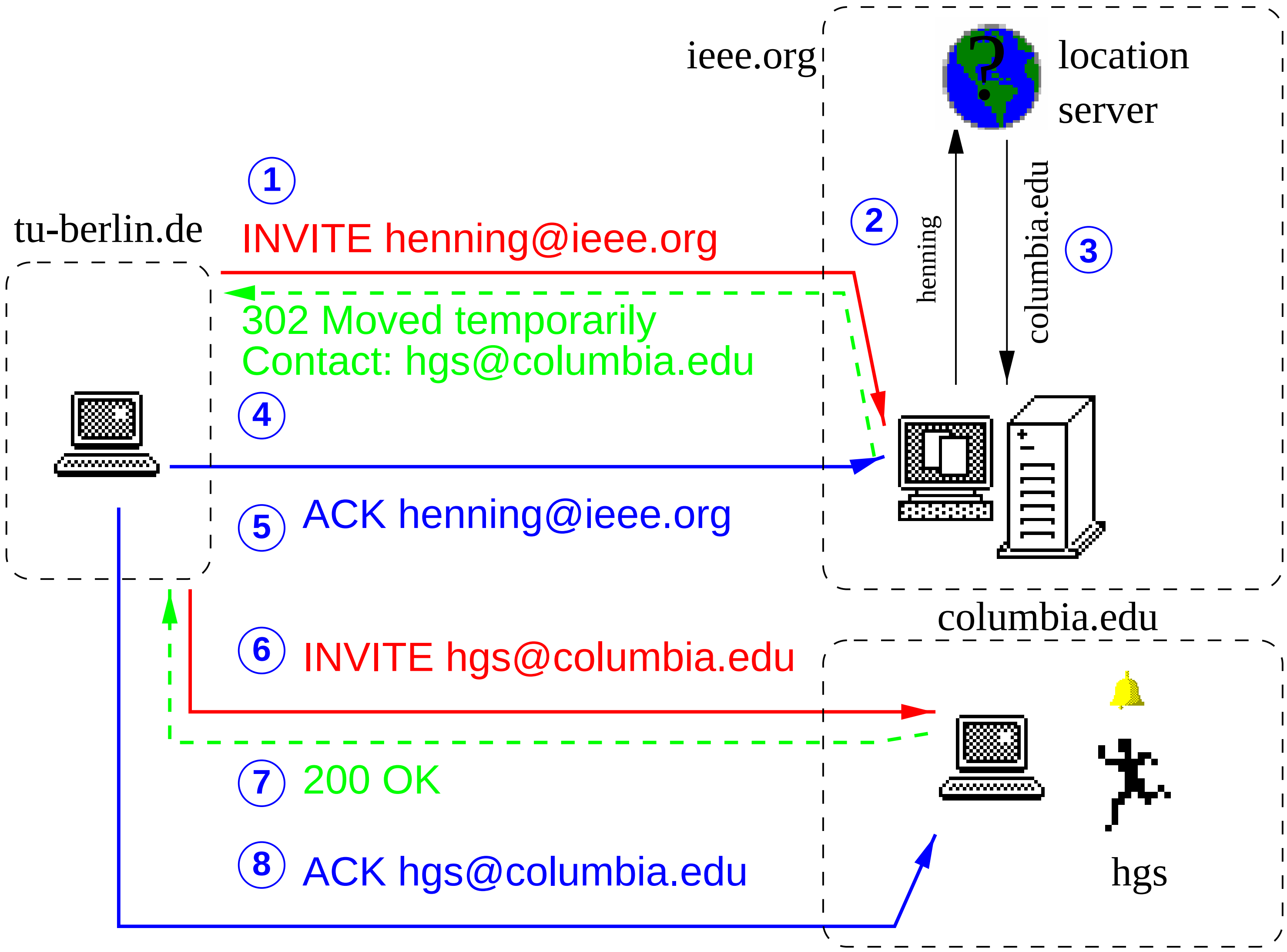
SIP: basic operation

1. use directory service (e.g., LDAP) to map name to *user@domain*
2. locate SIP servers using DNS SRV, CNAME, A
3. called server may map name to *user@host*
4. callee accepts, rejects, forward ($\rightarrow$ new address)
5. if new address, go to step 2
6. if accept, caller confirms
7. . . . conversation . . .
8. caller or callee sends BYE

SIP operation in proxy mode



SIP operation in redirect mode



SIP protocol design: robustness

SIP is designed to be robust against server failures:

- no state in proxy servers during call
- responses are “self-routing”
- subsequent requests and retransmissions can take different path (backup server)
- proxy servers can “lose memory” any time ➠ still function
- UDP ➠ less state than TCP, no time-wait

Invitation modes

invitation	conference	
	unicast	multicast
unicast	telephony	Internet TV session
multicast	reach first	dept. conference

SIP user location

- SIP is independent of mechanism to locate user
- examples:
  - local multicast of invitation
  - login-based via NFS
  - recursive “finger”-traversal
  - name translation: *Alexander.G.Bell* ➠ *agb*
  - active badges
  - SIP:
    * REGISTER announces location, with time limit
    * REGISTER + Location sets new location
    * forwarding within host ($\neq$ standard port)

Implementations

- 18 implementations at Columbia SIP "bake-off"
- Columbia sipd:
  - registration via unicast and multicast
  - handles mailing lists (ug-students@cs), ambiguous names (lee@cs)
  - maps names (b.clinton@whitehouse)
  - Apache (httpd)-style configuration and logging
  - "basic" authentication
  - how many servers for 2300 requests/second?

Signaling $\longleftarrow$ event notification

- call queueing . . . buddy lists . . . event notification
- SUBSCRIBE to events
- server NOTIFY
- can use forking
- handle subscriptions using CPL
- transition to multicast if large group of subscribers

Internet Telephony Signaling

- touch-tone transmission
- interoperation of SIP with SS7, ISDN and POTS
- large-scale IPtel gateways
- locating IPtel gateways (and other wide-area resources)
- charging for (adaptive) services and resources
- Internet voice mail

Programmable phone service

- “caller proposes, callee disposes, administrator decides”
- web = static pages $\longrightarrow$ cgi-bin $\longrightarrow$ Java
- “if somebody is trying to call for the 3rd time, allow mobile”
- “try office and lab in parallel, if that fails, try home”
- “allow call to mobile if I’ve talked to person before”
- “if on telemarketing list, forward to dial-a-joke”
- phone: CTI = complex, not generally for end users
- “cgi-bin” for Internet telephones: generate requests, proxy, responses ➠ sip-cgi, complete control

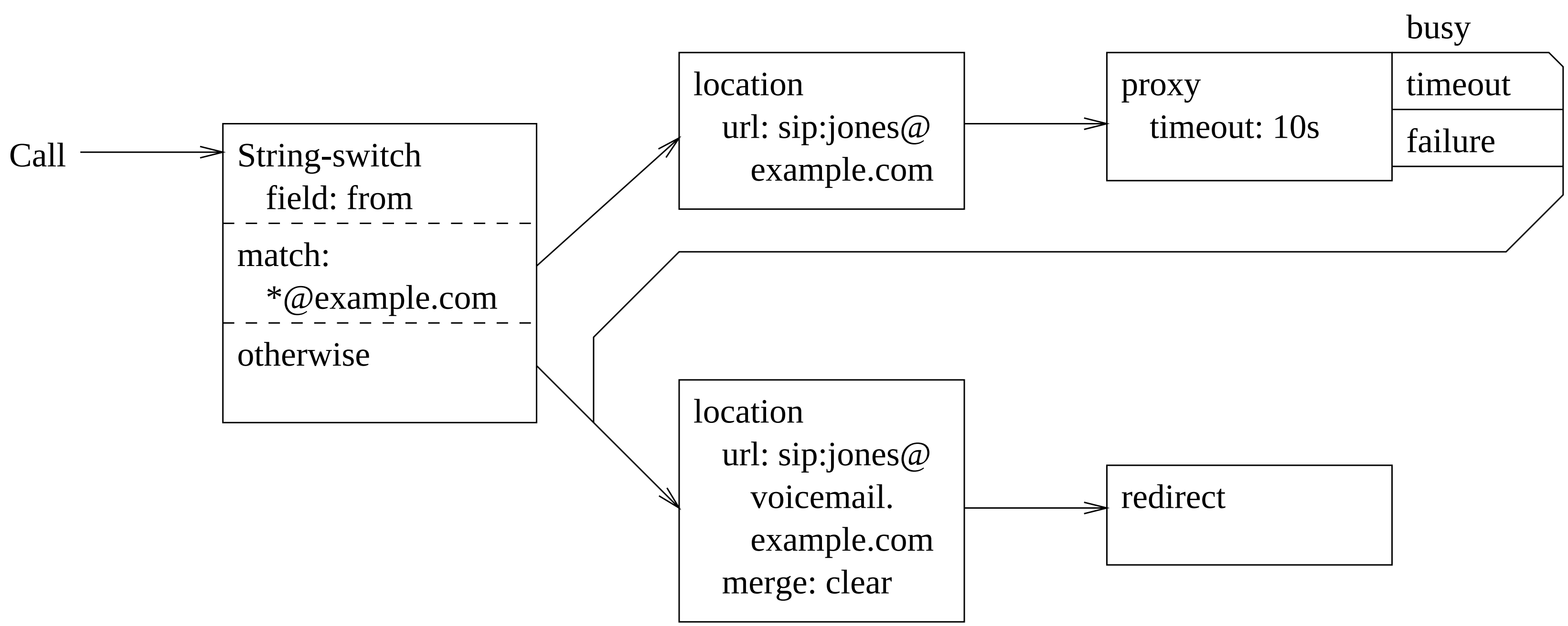
“Active Phone Networks”

language:

- don’t want Turing-complete language
- fail safe: make phone calls even if crashes
- predictable resource consumption
- hide parallelism (searches)
- hide timers
- execute in callee’s proxy server or end system (or phone button)

➠ CPL, an XML-based language

CPL example


busy
location
url: sip:jones@
example.com
proxy
timeout: 10s
timeout
failure
Call
String-switch
field: from
match:
*@example.com
otherwise
location
url: sip:jones@
voicemail.
example.com
merge: clear
redirect

CPL example

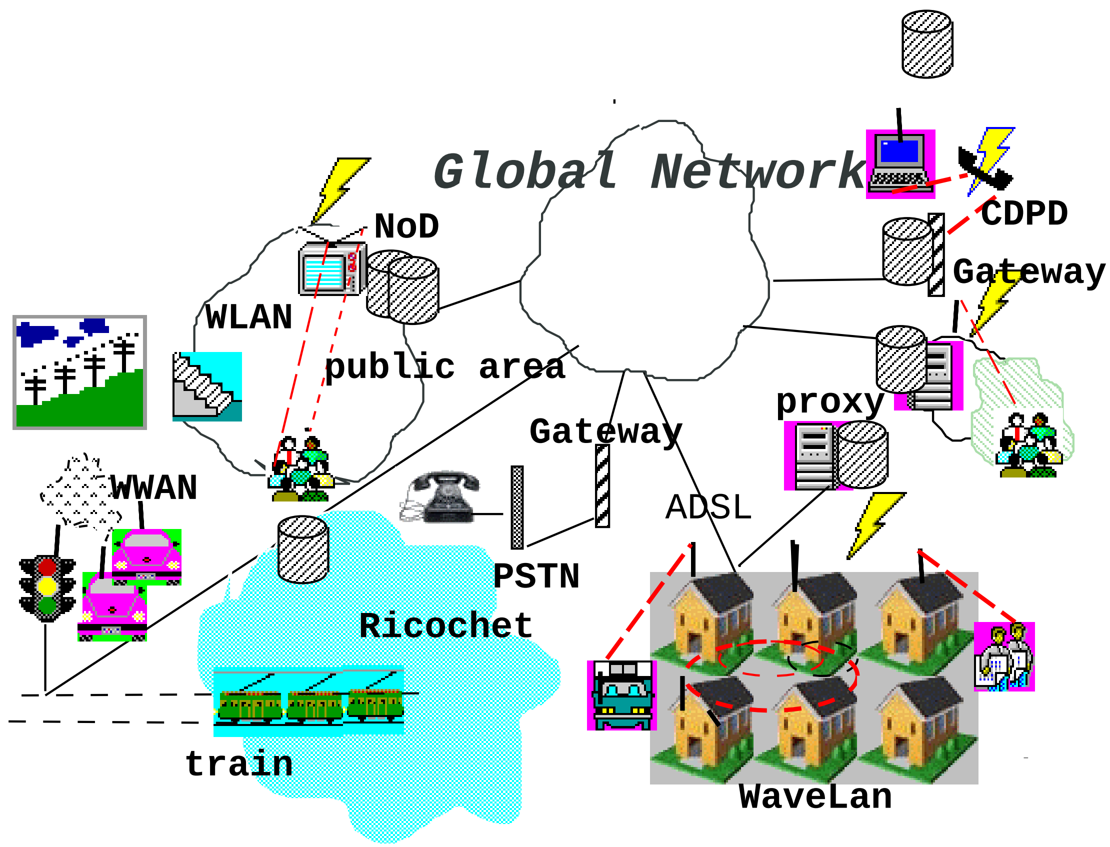
```
<call>
  <location url="sip:jones@jonespc.example.com">
     <proxy timeout="8s">
       <busy>
         <location url="sip:jones@voicemail.example.com" merge=
                   id="voicemail" >
            <proxy />
         </location>
       </busy>
       <noanswer>
         <link ref="voicemail" />
       </noanswer>
     </proxy>
  </location>
</call>
```

Internet phone “appliance”

- phone = $49.99; PC > $600 (GPF included)
- *Ethernet phone* ➠ no PBX for switching
- DSP for voice coding ➠ limited memory (256 kB SRAM!)
- implemented minimal IP stack (IP/UDP/RTP, DHCP, SIP, tftp, DNS)
- no TCP
- MP3 radio
- sensor interfaces to the world: chair, IR, temperature, …


128.59.19.38

Internet mobile services



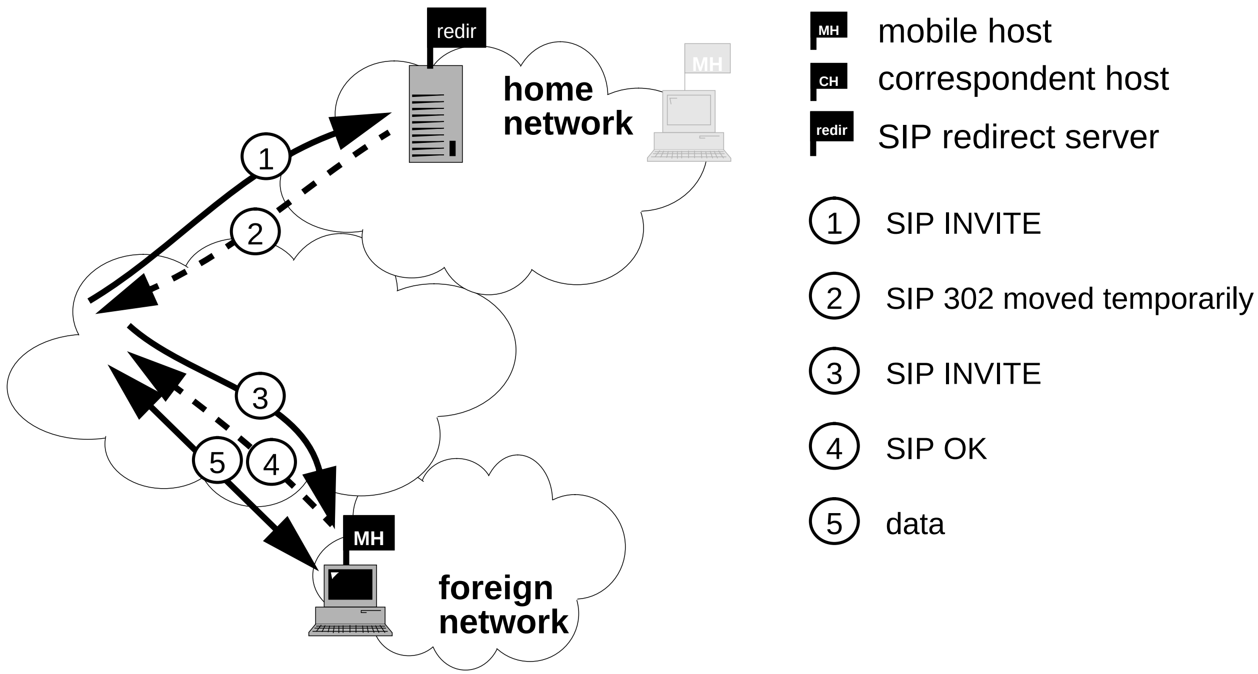
Internet mobile services: “social” ad-hoc networks

- connection sharing
  - multiple wireless networks: 2–10 Mb/s (WL Ethernet, IR) to 28 kb/s (Ricochet, CDPD)
  - share wide-area connections with neighbors
  - load sharing with mobile or in-home gateways
- social caching
  - subway model: in-car high-speed receiver updated in stations
  - socially optimal retrieval
  - anticipatory caching of streaming media
  - “leave the newspaper behind”

Internet cellular phone

- mobile IP: mask mobility to TCP connections
- imposes overhead:
  - all registrations to home agent
  - triangle routing (mostly)
  - encapsulation, address filtering problems
- ➠ use SIP and RTP for mobility management

SIP mobility



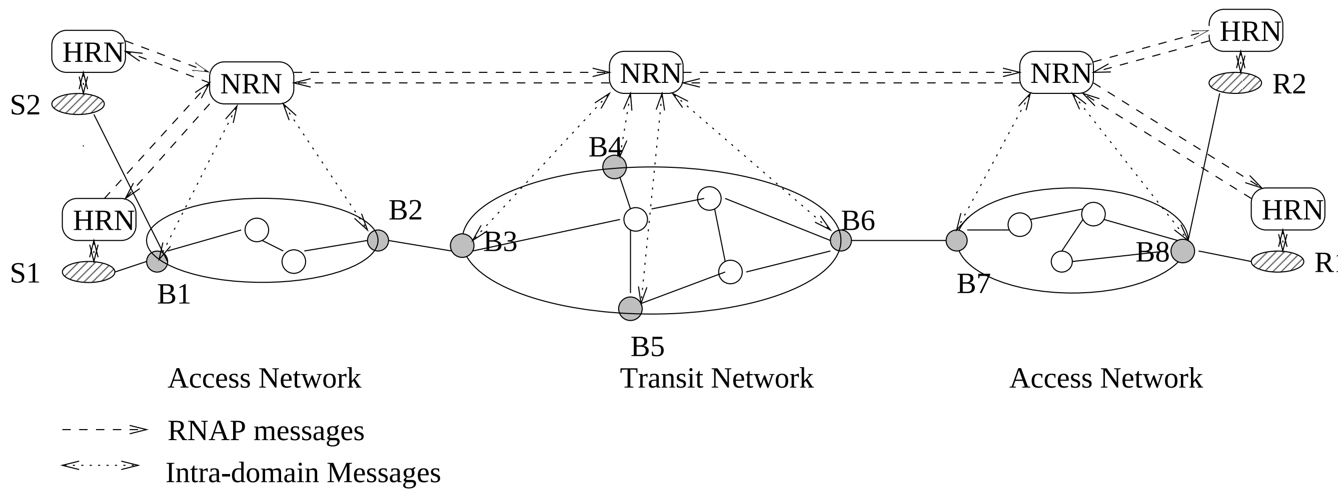
QOS

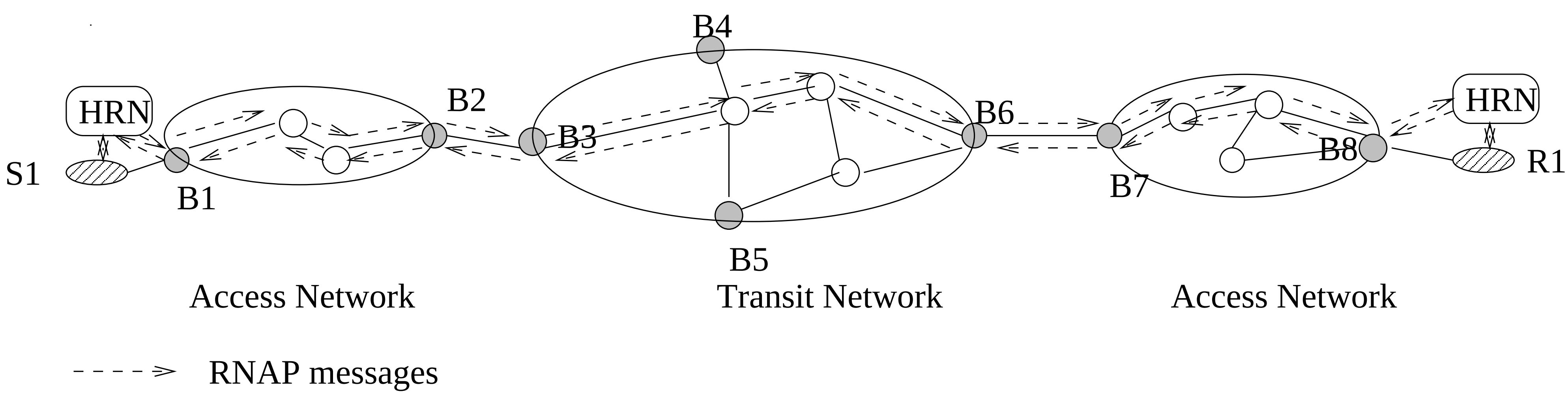
- RSVP refresh mechanisms
- aggregation of reservations (BGRP)
- quality-of-service characterization for IP telephony
- TCP-friendly adaptive services

Pricing Adaptive Multimedia Services

- pricing schemes that consider the holding cost, usage cost, and congestion cost as well as supports differential quality and advance reservation in the Internet.
- develop a network infrastructure that supports dynamic resource negotiation and pricing across Internet domains and users ➠ RNAP

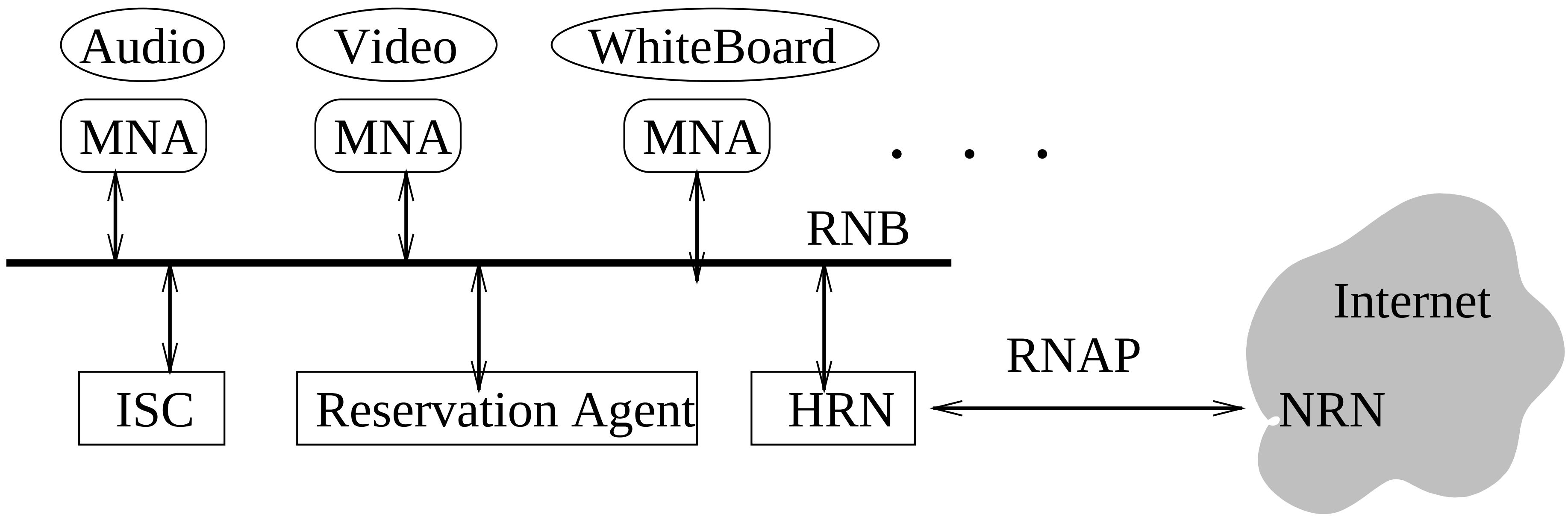
RNAP Architecture





Adaptive Multimedia Services

- design an incentive-compatible adaptation framework that optimizes user satisfication and network revenue, minimizes connection rejections.
- cross-media adaptation scheme to optimize system wide performance;
- designing test methods for user utility functions for different tasks


Audio
Video
WhiteBoard
MNA
MNA
MNA
RNB
Internet
RNAP
ISC
Reservation Agent
HRN
NRN

IP Multicast and Network Reliability

- performance analysis of different multicast protocols (DVMRP, MOSPF, CBT, PIM-DM, PIM-SM)
- failure tests of routing protocols PIM-DM and PIM-SM
- implementation of multicast protocols (PIM-DM, PIM-SM) in Opnet environment.

Reservation Aggregation: BGRP

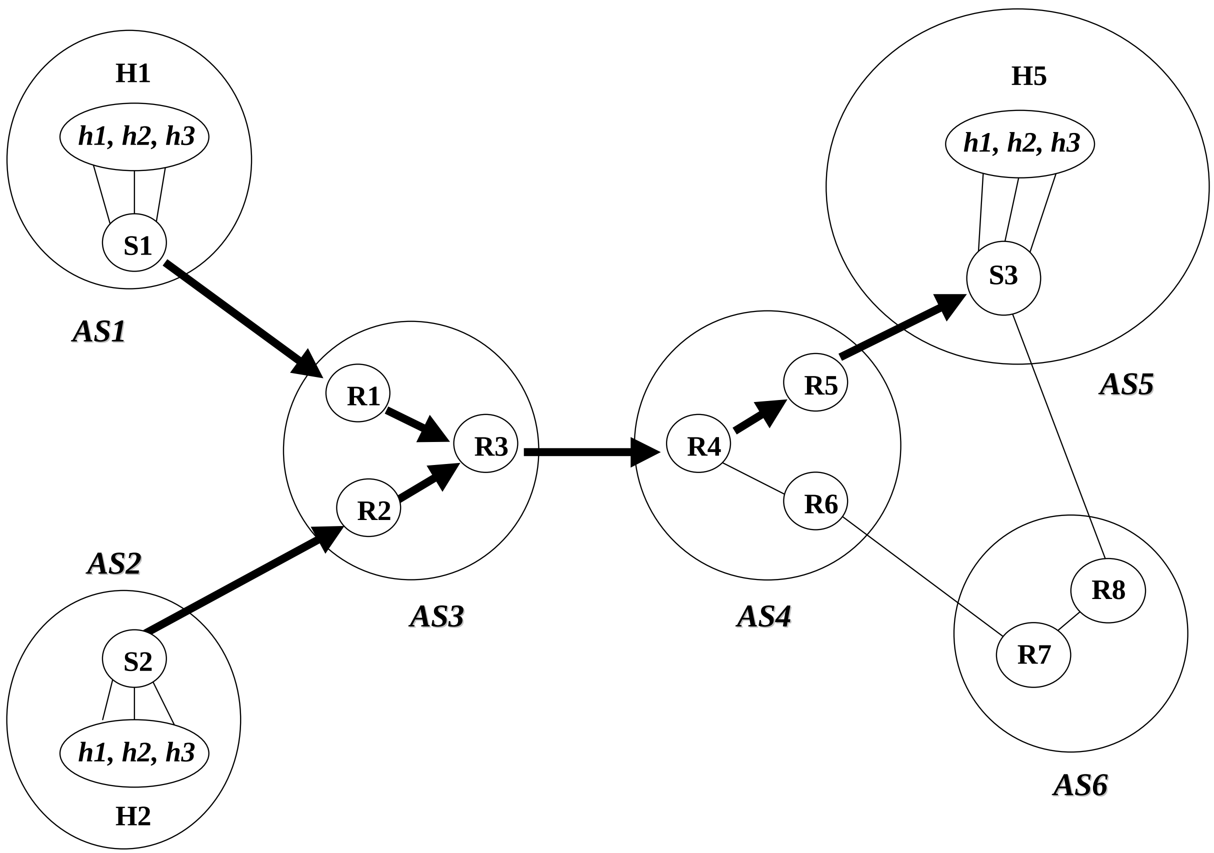
- RSVP: per-flow reservation $\longrightarrow$ scales N^2 with users
- need to bundle reservations
- BGRP: construct *sink tree* of data to common destination
- use hysteresis and over-reservation on trunks

Reservation Scaling

Flows and aggregations based on 90 s packet trace from MAE-West

Granularity	flow discriminators	flows
Application	source address, port	143,243
	dest. address, port, proto.	208,559
	5-tuple	339,245
IP Host	source address	56,935
	dest. address	40,538
	source-dest. pairs	131,009
Network	source network	13,917
	dest. network	20,887
	source-dest. pairs	79,786
AS	source AS	2,244
	dest. AS	2,891
	source-dest. pairs	20,857

BGRP Model


H1
h1, h2, h3
S1
AS1
H5
h1, h2, h3
S3
AS5
R1
R2
R3
R4
R5
R6
AS2
S2
h1, h2, h3
H2
AS3
AS4
R8
R7
AS6

BGRP

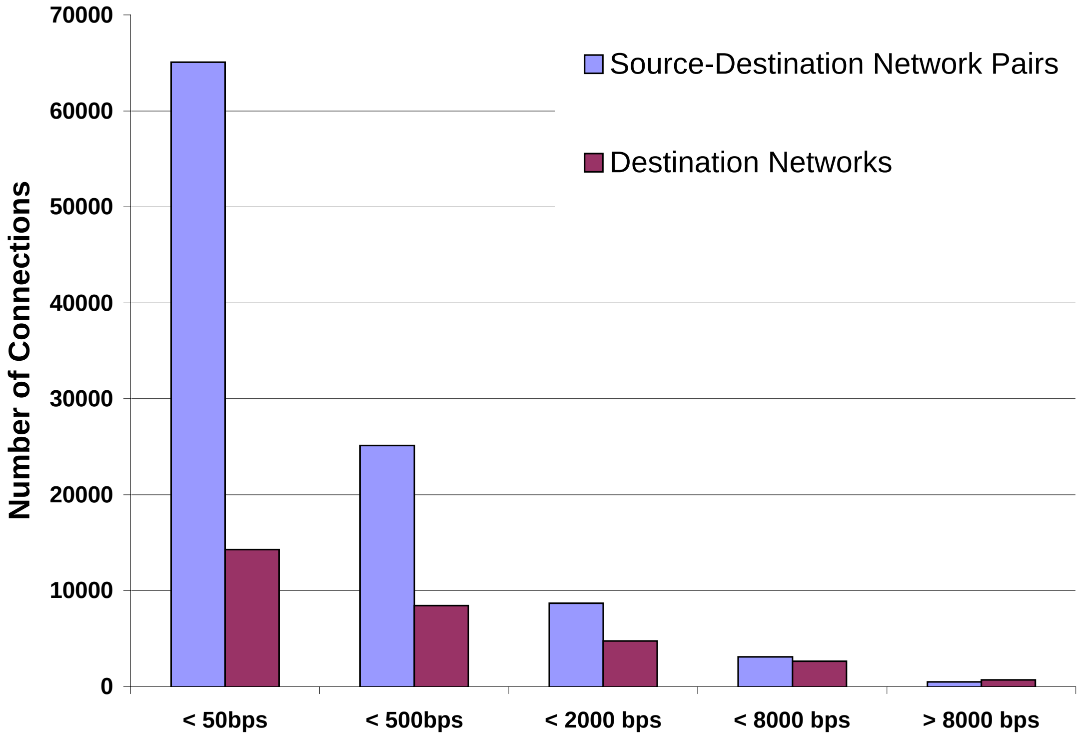
- *inter*-domain routing, soft-state
- PROBE from source to sink, stops when it meets tree
- GRAFT back to source(s), source-routed
- REFRESH between peers
- TEAR to accelerate tear-down
- RSVP: *shared* reservations; BGRP: *summed* reservations

Hysteresis

- allocate in reservation quantums Q to reduce signaling traffic

ρ	Q					
	2	5	10	15	20	25
10	4.1	30	830	3,000	4,000	63,000
100	4.7	26	110	270	590	1,400

BGRP Load Effect



Internet radio & TV

- control of streaming services (➠ RTSP)
- “trunking” of RTP streams
- reliability and recovery of multicast routing protocols
- program insertion, “networks” for Internet radio and TV
- large-group scaling of RTCP feedback

More information

Internet and telecom statistics:
http://www.cs.columbia.edu/˜hgs/internet

Papers: http://www.cs.columbia.edu/˜hgs/research/irt

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

RTSP: http://www.cs.columbia.edu/˜hgs/rtsp

SIP: http://www.cs.columbia.edu/˜hgs/sip