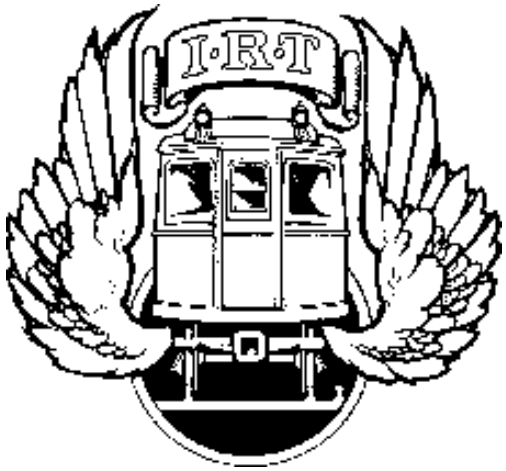
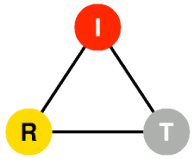
Managing (VoIP) Applications – DYSWIS



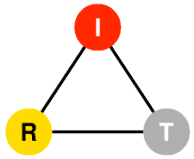
Henning Schulzrinne
Dept. of Computer Science
Columbia University
July 2005



CS@CU

Overview

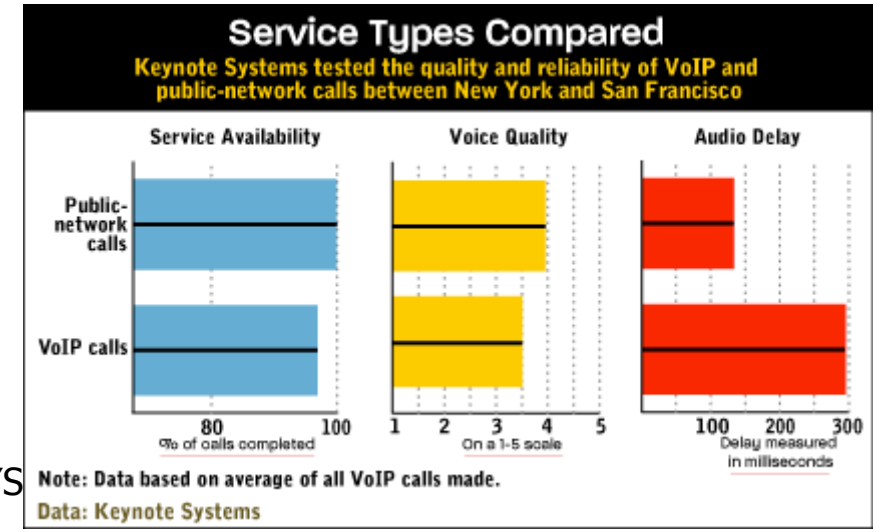
- User experience for VoIP still inferior
- Existing network management doesn't work for VoIP and other modern applications
- Need *user-centric* rather than *operator-centric* management
- Proposal: peer-to-peer management
  - "Do You See What I See?"
- Also use for reliability estimation and statistical fault characterization



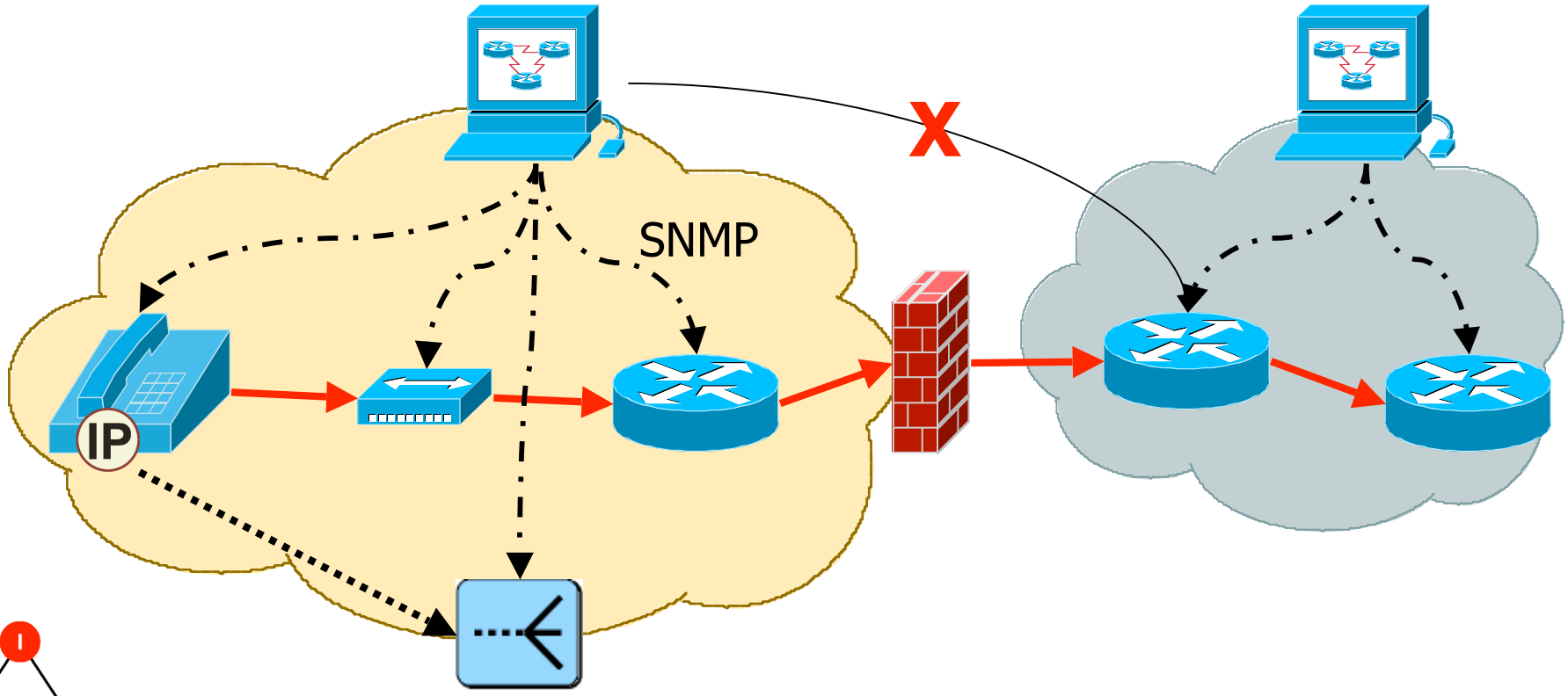
CS @CU

VoIP user experience

- Only 95-99.5% call attempt success
  - "Keynote was able to complete VoIP calls 96.9% of the time, compared with 99.9% for calls made over the public network. Voice quality for VoIP calls on average was rated at 3.5 out of 5, compared with 3.9 for public-network calls and 3.6 for cellular phone calls. And the amount of delay the audio signals experienced was 295 milliseconds for VoIP calls, compared with 139 milliseconds for public-network calls." (InformationWeek, July 11, 2005)
- Mid-call disruptions
- Lots of knobs to turn
  - Separate problem: manual configuration

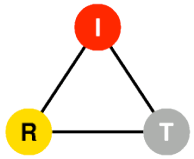


Traditional network management model



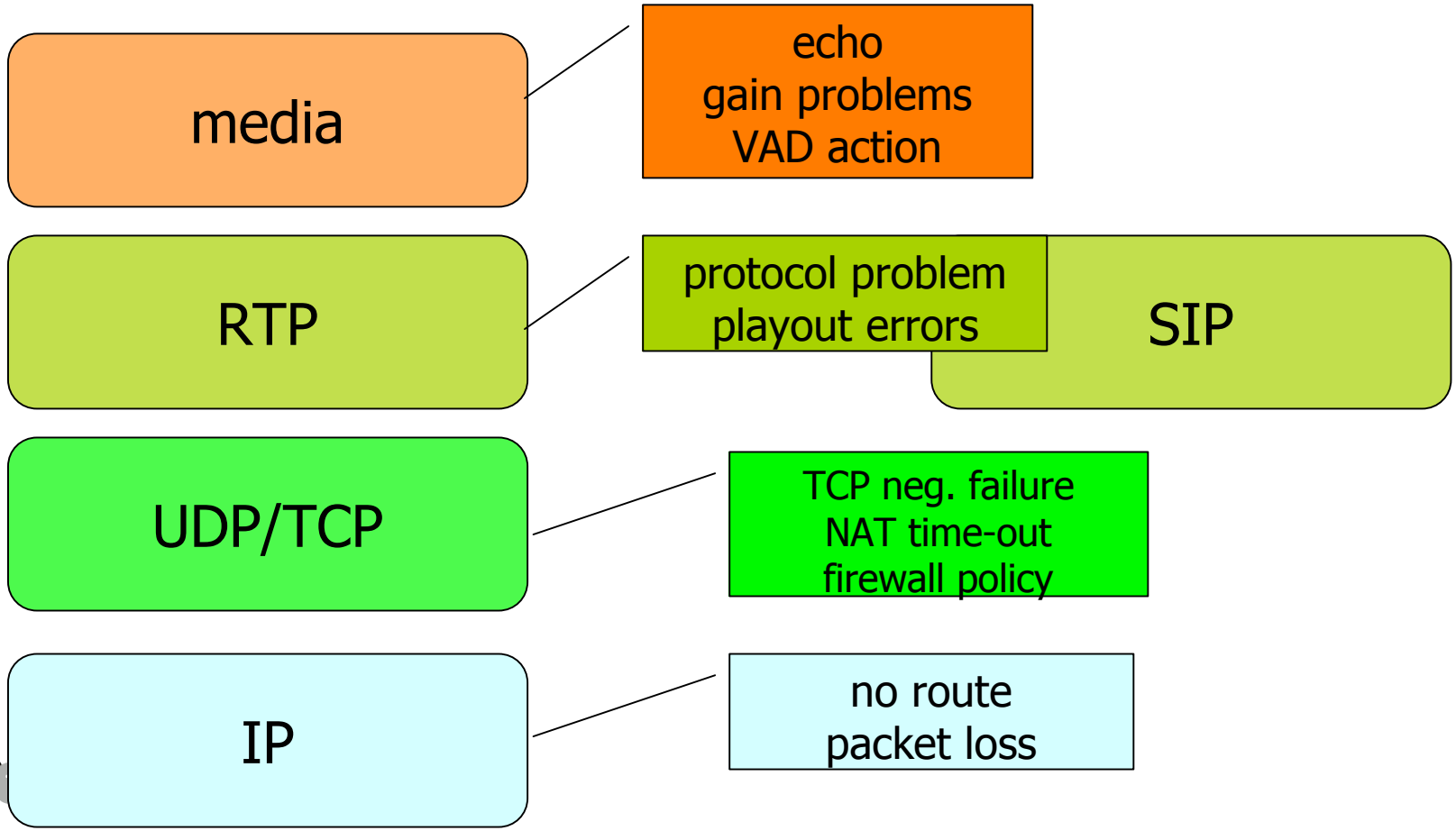
Assumptions

- Single provider (enterprise, carrier)
  - has access to most path elements
  - professionally managed
- Typically, hard failures or aggregate problems
  - element failures
  - substantial packet loss
- Mostly L2 and L3 elements
  - switches, routers
  - rarely 802.11 APs
- Indirect detection
  - MIB variable vs. actual protocol performance

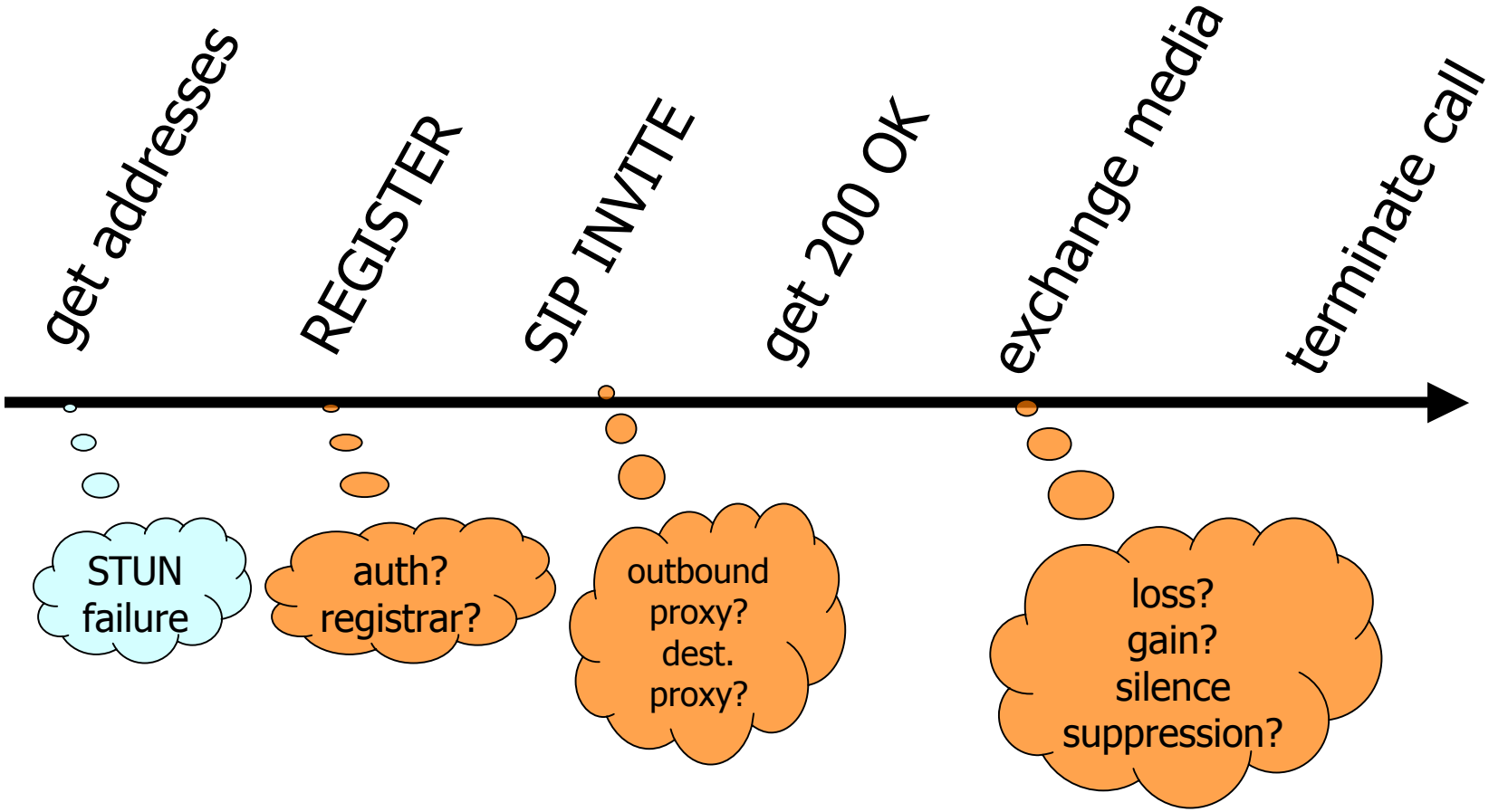


CS @CU

Managing the protocol stack

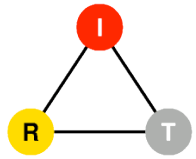


Call lifecycle view



Types of failures

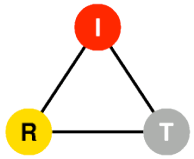
- Hard failures
  - connection attempt fails
  - no media connection
  - NAT time-out
- Soft failures (degradation)
  - packet loss (bursts)
    - access network? backbone? remote access?
  - delay (bursts)
    - OS? access networks?
  - acoustic problems (microphone gain, echo)



CS @CU

Diagnostic undecidability

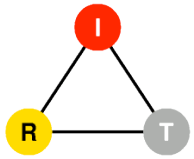
- symptom: "cannot reach server"
- more precise: send packet, but no response
- causes:
  - NAT problem (return packet dropped)?
  - firewall problem?
  - path to server broken?
  - outdated server information (moved)?
  - server dead?
- 5 causes → very different remedies
  - no good way for non-technical user to tell
- Whom do you call?



CS @CU

Additional problems

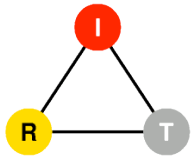
- `ping` and `traceroute` no longer works reliably
  - WinXP SP 2 turns off ICMP
  - some networks filter all ICMP messages
- Early NAT binding time-out
  - initial packet exchange succeeds, but then TCP binding is removed ("web-only Internet")



CS @CU

"Do You See What I See?"

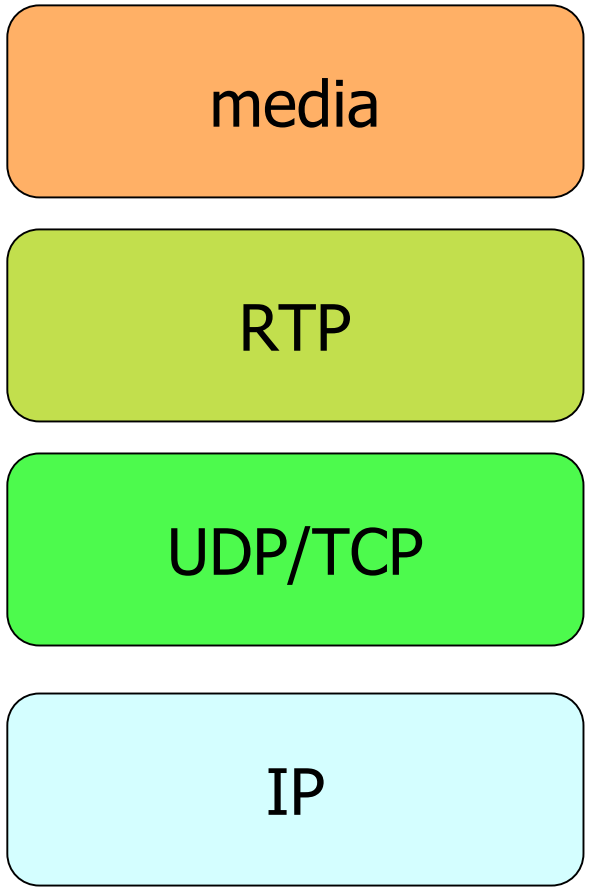
- Each node has a set of active measurement tools
- Nodes can ask others for their view
  - possibly also dedicated "weather stations"
- Iterative process, leading to:
  - user indication of cause of failure
  - in some cases, work-around (application-layer routing) → TURN server, use remote DNS servers
- Nodes collect statistical information on failures and their likely causes

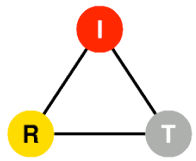


CS @CU

Failure detection tools

- STUN server
  - what is your IP address?
- ping and traceroute
- Transport-level liveness
  - open TCP connection to port
  - send UDP ping to port

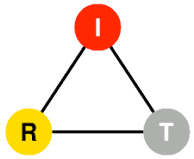




CS @CU

Failure statistics

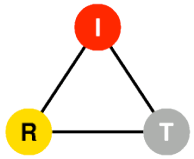
- Which parts of the network are most likely to fail (or degrade)
  - access network
  - network interconnects
  - backbone network
  - infrastructure servers (DHCP, DNS)
  - application servers (SIP, RTSP, HTTP, …)
  - protocol failures/incompatibility
- Currently, mostly guesses
- End nodes can gather and accumulate statistics



CS @CU

How to find management peers?

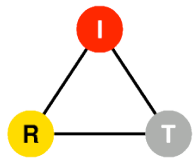
- Use carrier-provided bootstrap list
- Previous session partners
  - e.g., address book
- Watcher list



CS @CU

What's missing?

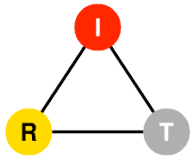
- Request diagnostic
  - "send this message"; return result
  - do specific high-level operation (ping, traceroute, DNS resolution)
- Failure statistics protocol and data exchange format
- Algorithm specification for steps
  - "if no response to REGISTER, check server liveness"
  - "if bad voice QoS, ask subnet neighbor; then ask somebody close to destination"



CS @CU

Security issues

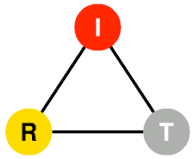
- Indirect denial-of-service attacks
  - limit per-requestor rate
  - return cached results to querier
- Lying
- Non-participation ("leechers")
  - usual P2P mechanisms such as blacklists



CS @CU

Conclusion

- Existing management mechanisms ineffective
- Outline of user-centric management approach
- Next steps
  - what protocols are needed?
  - trust and security issues?



CS @CU