Integration of Resource Management and Call Signaling for IP Telephony

W. Marshall, K. K. Ramakrishnan, E. Miller, G. Russell, B. Beser,
M. Mannette, K. Steinbrenner, D. Oran, W. Guckel, J. Pickens,
P. Lalwaney, J. Fellows, D. Evans, K. Kelly, F. Andreasen

AT&T, CableLabs, 3Com, Cisco, Com21, General Instrument, Lucent Cable, NetSpeak, Telcordia

November 1999

IETF Presentation

Outline of talk

- Overview of Resource Management
  - Critical from service provider's point of view
  - call blocking vs. call defects
  - theft of service
  - requirements for two-stage resource reservation model
- Post-pickup delay
  - why it is important, how clipping happens (both originator and of destination)
- Two-stage invite model
- Conversion of two-stage invite model to single-stage model
- Changes needed in SIP to support a two-stage Invite model

High Speed Data Service vs. Telephony Service

- Customer with full access to High Speed Data Service:
  - Initiates his/her own connection directly to called party
  - Requests bandwidth via RSVP
  - Uses connection to exchange voice packets
- Service Provider's perspective:
  - ➪ Flat rate/month data+voice service --> go broke
- Can we do anything to stop this? No, but….
- High Quality of Service is fundamental to voice communication
  - reduced packet loss probability
  - reduced round-trip latency

Telephony Service over IP Networks

- Service Provider must implement Policy Restrictions on RSVP
  - How about if only telephony subscribers can do the RSVP?
    - ➪ Flat rate/month data service, flat rate/month voice service
  - How about if RSVP required per-call authorization?
    - ➪ Flat rate/month data service
    - ➪ Voice service billed per-call, or per-minute, any-distance
- To achieve current telephony billing model
  - RSVP only accepted when pre-authorized for this specific call
  - RSVP only accepted to pre-authorized destination

Call Blocking vs. Call Defect

- Blocked call: a call that does not complete due to
  - overload/congestion
  - traffic management policies
- Network engineered for a specific blocking probability
  - typically 0.1%
- Customers receive “fast busy” or recorded message

- Call defect: a call that fails after ring/ringback
  - link or switch failure
  - excessive dropped signaling packets
- Maximum defect rate (due to FCC regulations)
  - typically 0.00001%
- Customers may hear ring turn into fast-busy, or may hear nothing

Prevention of Call Defects vs. Theft of Service

- When to allocate resources needed for the call?
  - After answer - Causes a blocked call to become a call defect.
  - Prior to ringing - Opens Theft of Service possibility

- Can we charge customer for the time the phone is ringing?
  - Business: yes, we do
  - Residential: no.

- Can we trust the customer to tell us when called party answers?
  - No. Why should they?

Service Provider Perspective: Theft of Service is a Major Problem

- Customers have access to High Speed Data Service
- Access is via box within customer premises
- Via a box that we encourage others to program

- Experience with Cellular phones, Set Top Boxes, etc
  - Customers will open the box
  - Customers will load their own versions of software
  - Some customers will reverse engineer the box
  - Some customers will build their own variations of the box

- Expect the box in customer's home to do exactly what is in the customer's best interest to do, not what is in the network or service provider's interests.

Theft of Service Examples, Continued

- Can we trust the customer to tell us when called party answers? No.
- Can we place a time limit on ringing?
  - Intelligent Customer box can easily make long series of short no-answer calls
  - Note that (1/N) * N = 1
- Other alternatives?

- Conclusion: we only give the resources for the connection when the endpoint says call answered

Theft of Service Examples, Continued

- But what if called party says they answered, but calling party (payor) didn't?
- But what if calling party (payor) says to disconnect, but called party doesn't?
  - These cases leads to high QoS flow in one direction but not the other.

- If two cooperating users place calls to each other, each getting a free half-call ---
  - ➪ Note that 2 * 1/2 = 1

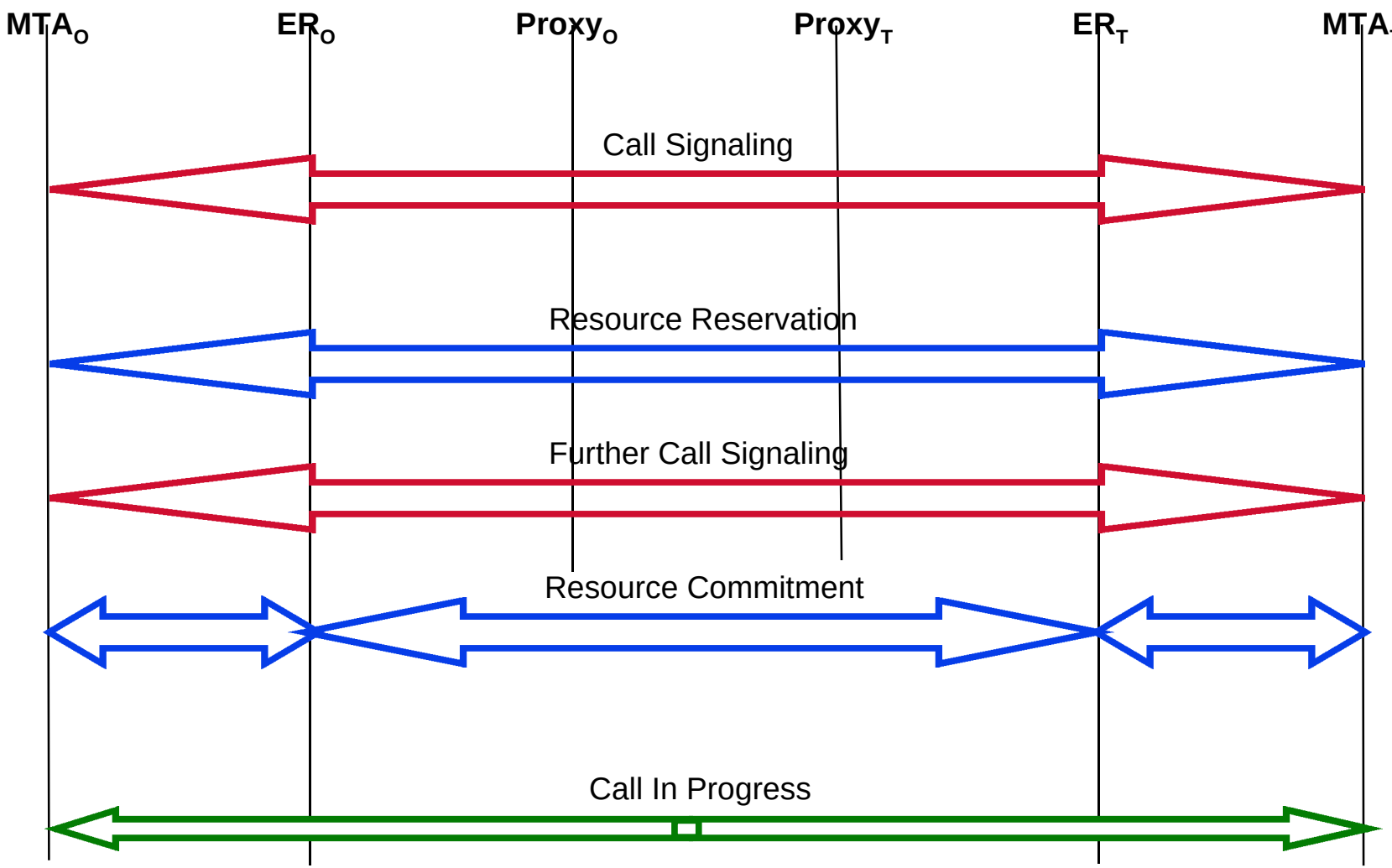
Theft of Service, Conclusion

- Must have a control point inside the trusted network to police requests
- Must limit packets to only the destination authorized, and limit bandwidth to only the amount authorized.
- Must coordinate with the control point at other end of connection

Basic Requirements for Resource Management

- Resources authorization
  - per-call
  - to a single destination
  - policed at the edge of the service provider's network
- Resources reserved prior to alerting
- Resources committed when billing is to begin

Two-Stage Resource Reservation



Post-Pickup Delay Requirements

- Measurement starts the instant the called party answers

- When can called party start talking?
- When can calling party start talking?
- How long will the called party wait before hearing a response?

- Need to consider Operators (especially those using headsets), Voice Response systems, etc.

- All requirements are based on customer and system expectations
- Expectations currently based on PSTN experience
- PSTN requirement today for cut-through of voice path is 100ms.

Post-Pickup Delay - Voice Path

0	Called party answers
	Assume: 20ms delay of MTA access of upstream
20ms	MTA sends Commit message to Edge Router
	Assume Edge Router schedules QoS in 4ms
24ms	Edge Router schedules QoS (UGS) and sends MAP
30ms	First Payload packet sent by MTA
	Assume: 40ms one-way delay across network
70ms	First Payload packet received by call originator

Post-Pickup Delay - Signaling Path

0	Called party answers
	Assume: 20ms delay of MTA access of upstream
20ms	MTA sends 200-OK message to its Proxy(1)
	Assume: Proxy load 80% (average queue of 4 messages), 10ms of processing for a request, 5ms for a response; delay=35ms
55ms	Proxy(1) sends 200-OK to Proxy(2)
	Assume: 33.3ms transmission delay Proxy-Proxy
88ms	Proxy(2) receives the 200-OK
	Assume: 4 Proxies between calling and called parties
123ms	Proxy(2) sends 200-OK to Proxy(3)
192ms	Proxy(3) sends 200-OK to Proxy(4)
260ms	Proxy(4) sends 200-OK to originating MTA

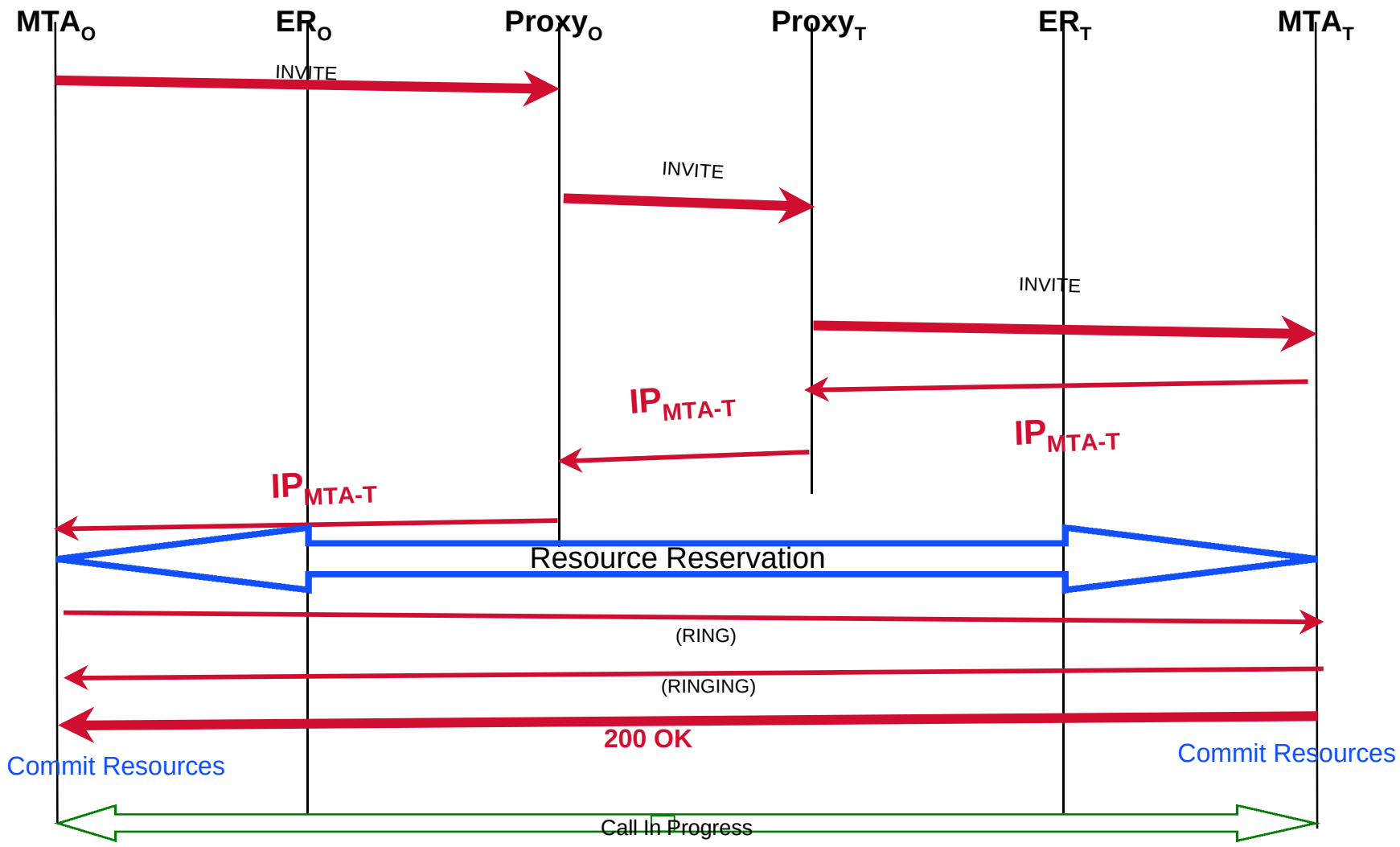
Clipping Choices at Call Originator

- What is the call originator to do when receiving payload before the 200-OK?
  - Discard it and wait for signaling message
    - ➪ Clip first 190ms of called party voice
  - Process it and play it out, but wait for signaling to start sending
    - ➪ Clip first 210ms of calling party voice
  - Treat the voice payload packet as call completion signaling information
    - ➪ yuck

Call Setup:
Fundamental Requirements

- First INVITE MUST go through proxies for authorization/translation
- Resources must be available PRIOR to ringing destination phone
- Answer must go direct, not via proxies

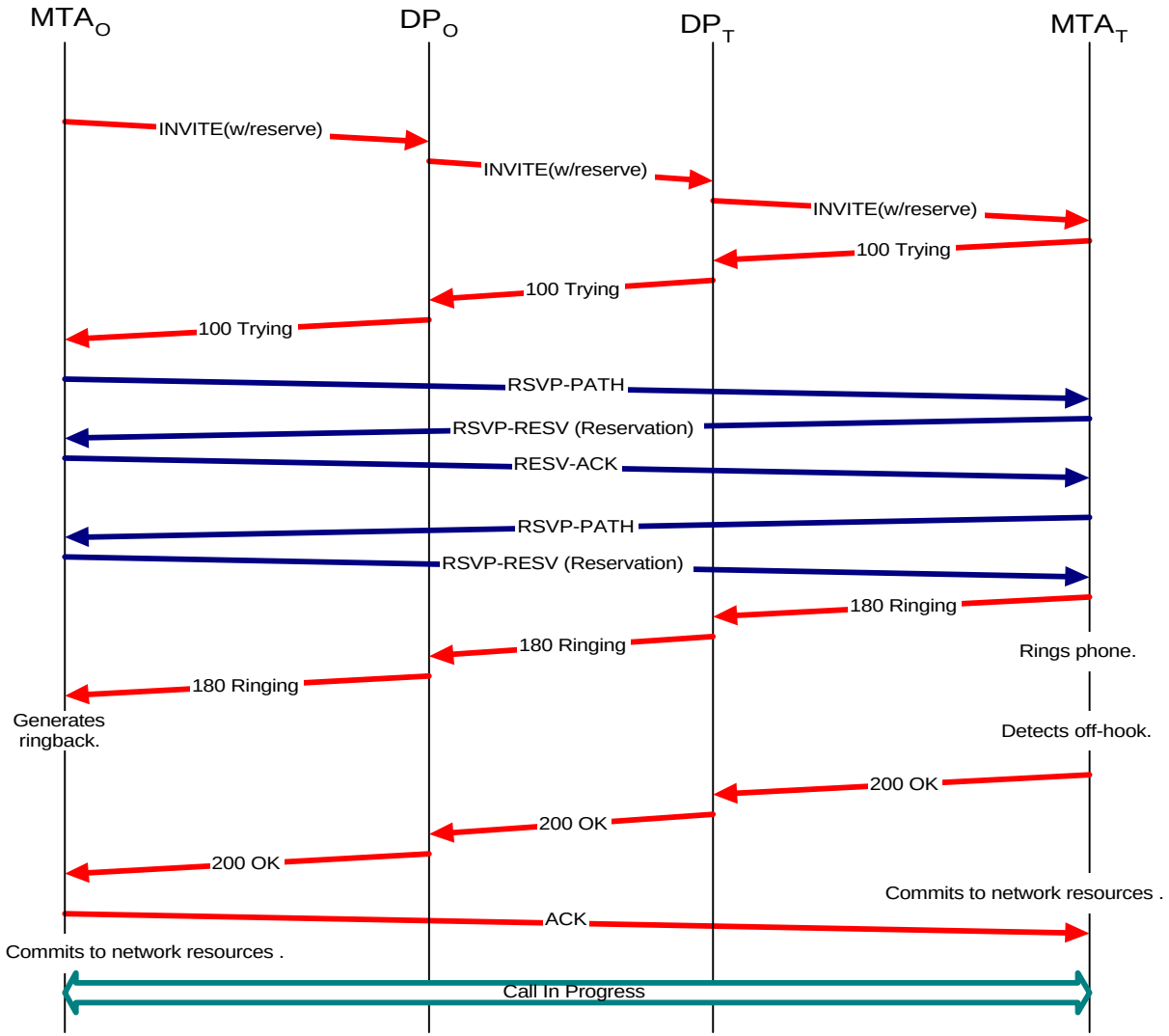
Critical DCS Signaling Messages and their Relationship to Resource Allocation



Single-stage Alternatives Examined

- Allocate resources in access prior to single traditional INVITE
  - RSVP? But don't know destination IP address of request
  - New protocol needed to do this
- Allocate resources after INVITE, before Ringing phone
  - E.g., Destination send 100 Trying in response to INVITE
    - 100 Trying (via GC) must include SDP
    - Caller does an end-end RSVP, Destination requests RESV-ACK
    - Called party also does an end-end RSVP exchange
    - On RESV-ACK + RESV, send 180 Ringing (via proxies)
  - 200 OK on pickup routed via Proxies
    - meanwhile, voice from destination is cut-through and may arrive at origin before 200 OK. User response is "clipped"

Single Stage Invite Sequence, with Resource Reservation

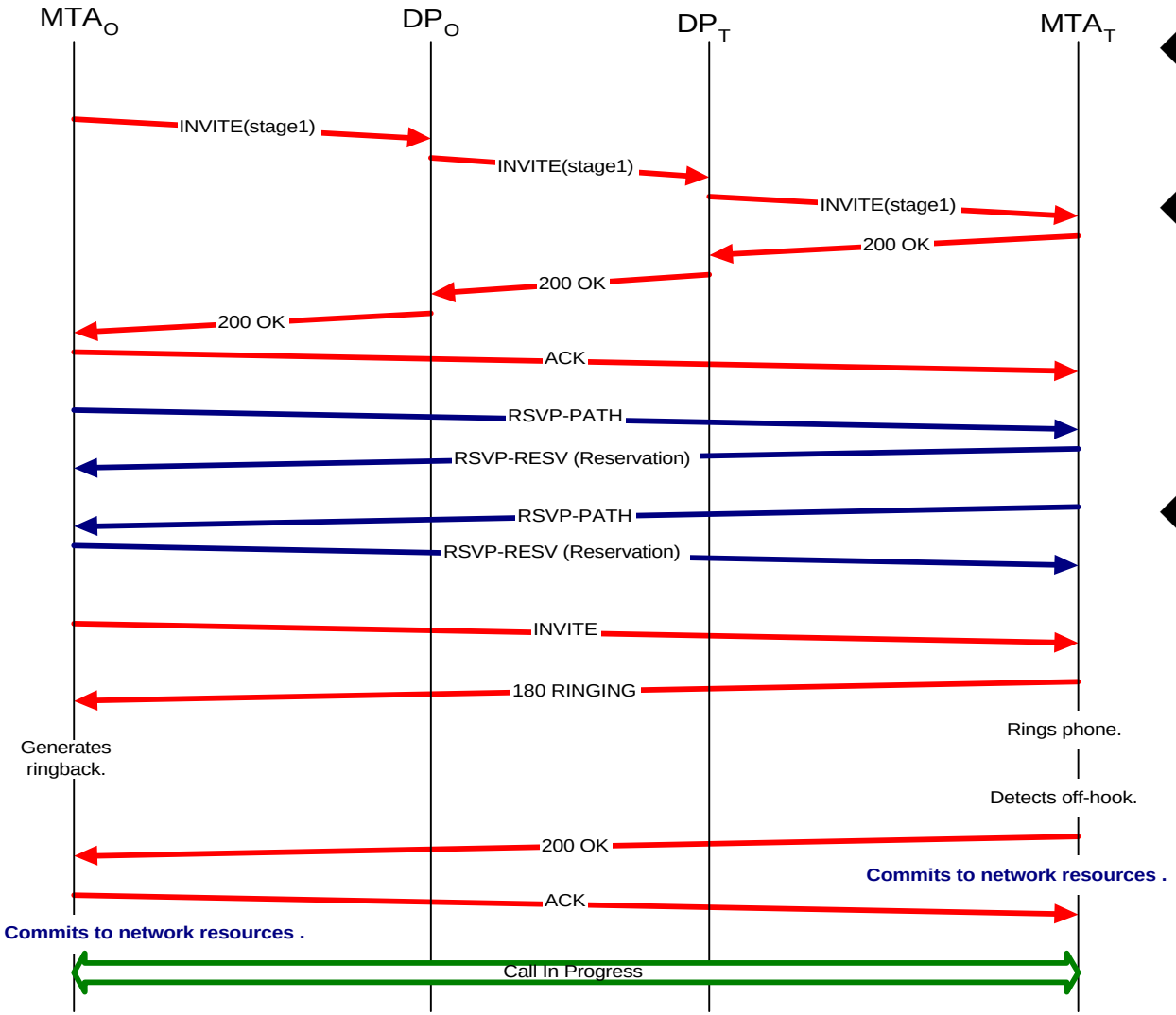


- INVITE(w/reserve)
- 100-Trying gives SDP for forward direction media
- RSVP initiated from both ends for reservation
- RESV-ACK requested by dest.
- 180-Ringing when all resources available
- May clip voice

Conclusion: Need for Two-Stage INVITE

- First Stage - INVITE but do not alert receiving customer
  - Sent via proxies to perform authorization, translation, etc.
  - Acknowledgement via proxies, with Contact: header
  - Call features invoked with first INVITE, e.g. Call Forwarding, etc.
- Second Stage - INVITE with alerting
  - Sent direct end-to-end
  - Interim and Final responses sent direct, low-latency path
- Tradeoff between post-pickup delay and No-Answer handling
  - Call Forwarding No Answer becomes a transfer operation
    - » Use call-control services, Replace: and Also: headers
    - » Assistance from proxies likely

Two-Stage Invite Sequence, with Resource Reservation

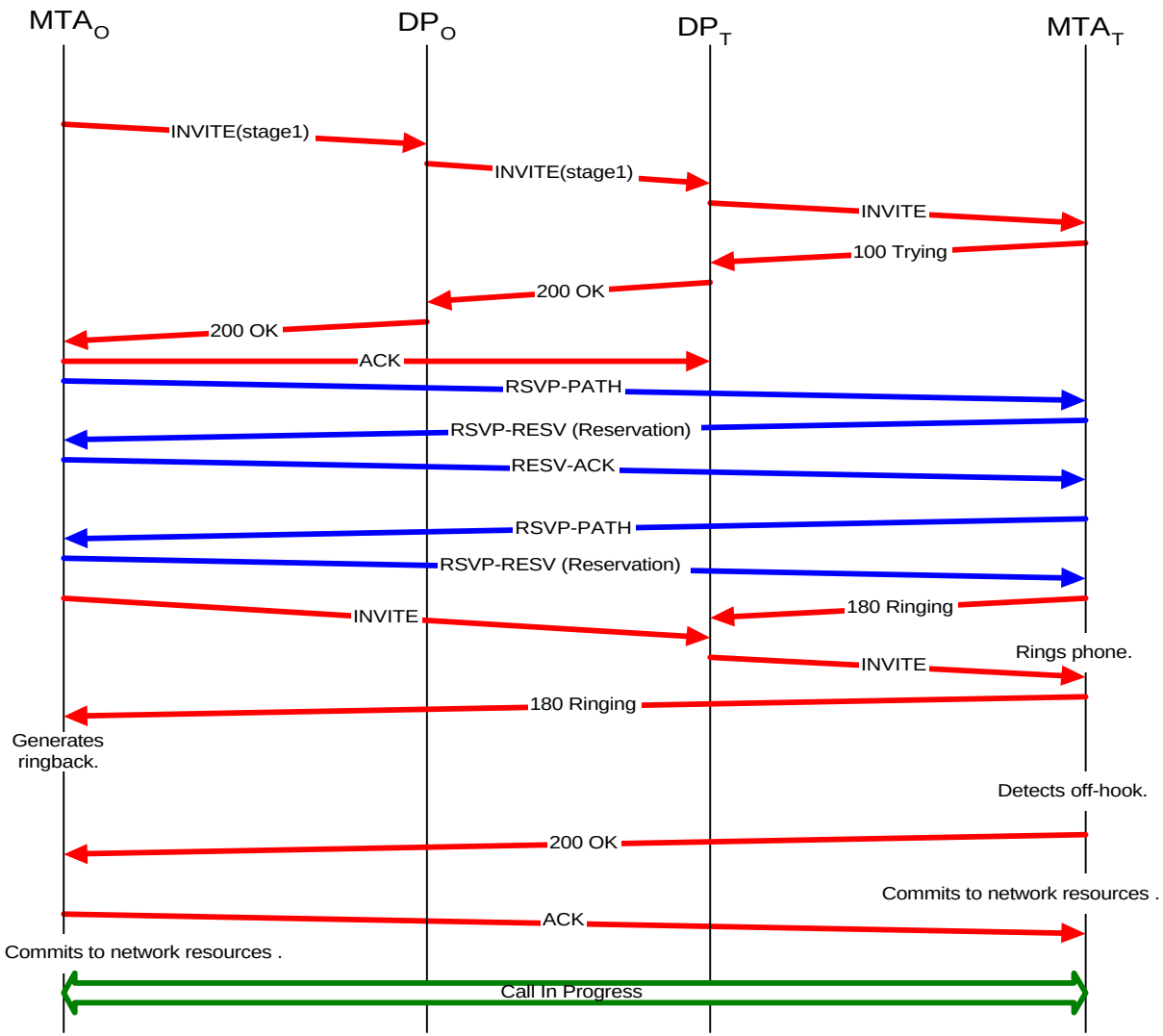


- INVITE(Stage1), 200-OK, ACK
- Resource reservation via RSVP (no RESV-ACK needed)
- Second INVITE causes alerting

Proxy conversion of signaling
Single-stage Invite to Two-stage Invite

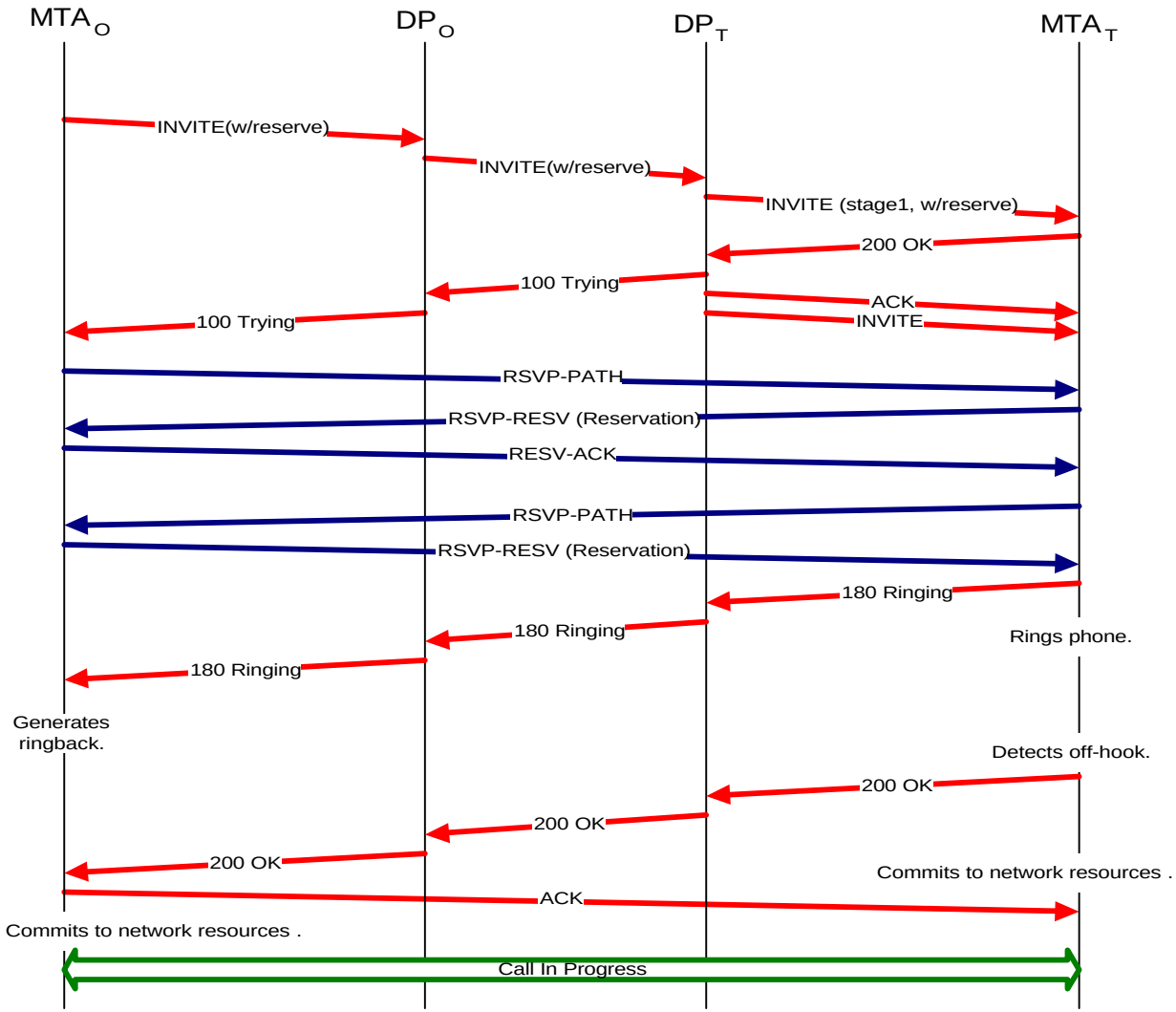
- A knowledgeable proxy can convert from one style to the other

Two-stage INVITE endpoint calling Single-stage INVITE endpoint



- INVITE(Stage1) becomes INVITE(w/reserve)
- 100-Trying becomes 200-OK
- ACK absorbed
- 180-Ringing absorbed
- Second INVITE passed through
- Potential race if answered quickly, may lead to clipping

Single-stage INVITE endpoint calling Two-Stage INVITE endpoint



- INVITE(w/reserve) becomes INVITE (stage1, w/reserve)
- 200-OK becomes 100-Trying, and generates ACK and second INVITE
- 180-Ringing and 200-OK both go hop-by-hop
- Potential clipping

SIP Support needed for Two-Stage INVITE Mechanism

- Additional header in initial INVITE message
  - Stage1 = “Stage1” “:”