VOIP with NATs and Firewalls

Robert Sparks

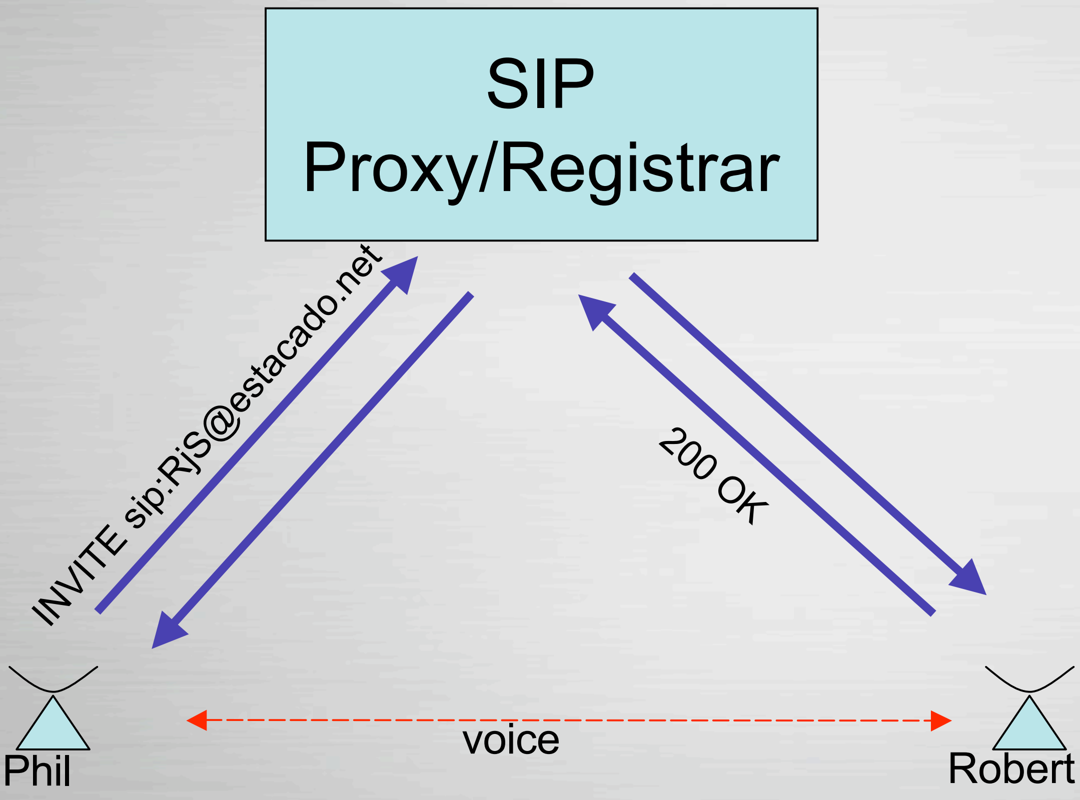
Protocol Barbarian

Estacado Systems

Voice (and more) over IP

- Signaling : Session Initiation Protocol (SIP)
  - Looks, but does not act, like HTTP
  - Defined over a variety of transports
  - Used for *rendezvous* (helping endpoints find each other
  - Used to negotiate media (addresses, formats)
- Media : Real-time Transport Protocol (RTP)
  - Carries media (voice, video, other)
  - Represented via standardized codecs
  - Internally sequenced and timestamped

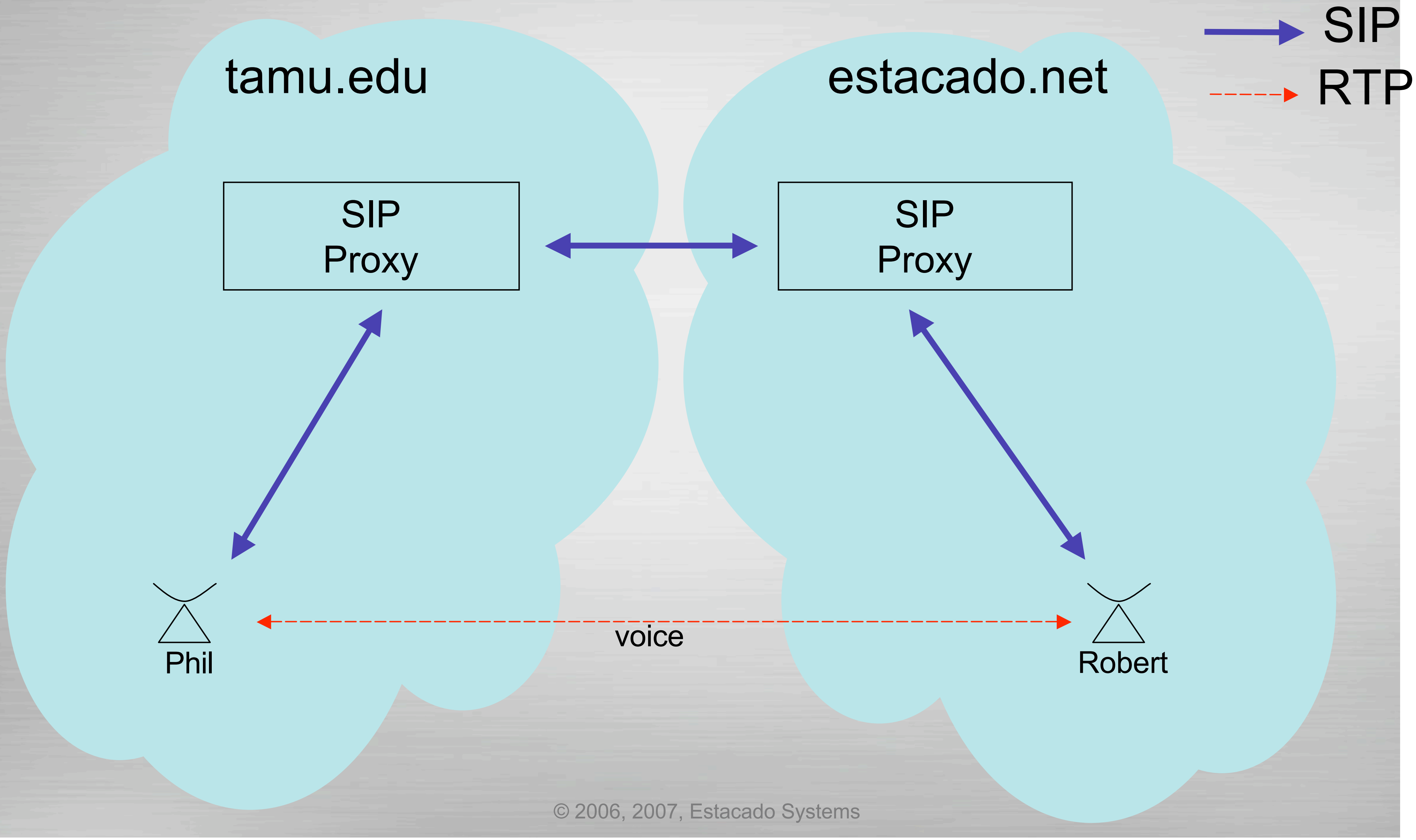
Basic Architectural Model



→ SIP over UDP, TCP, or TLS

⇢ RTP over UDP using ephemeral source and destination ports

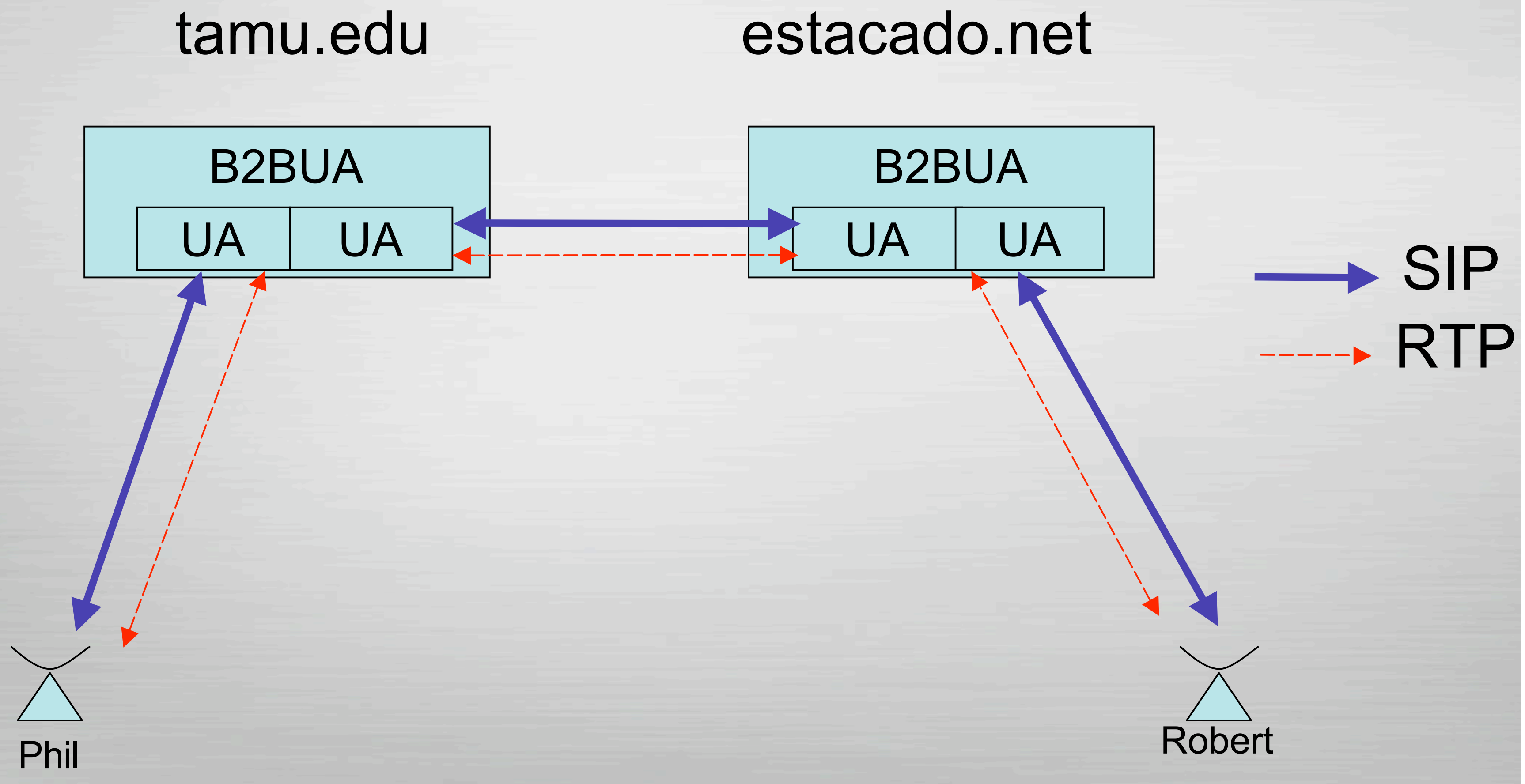
Trapezoid Model



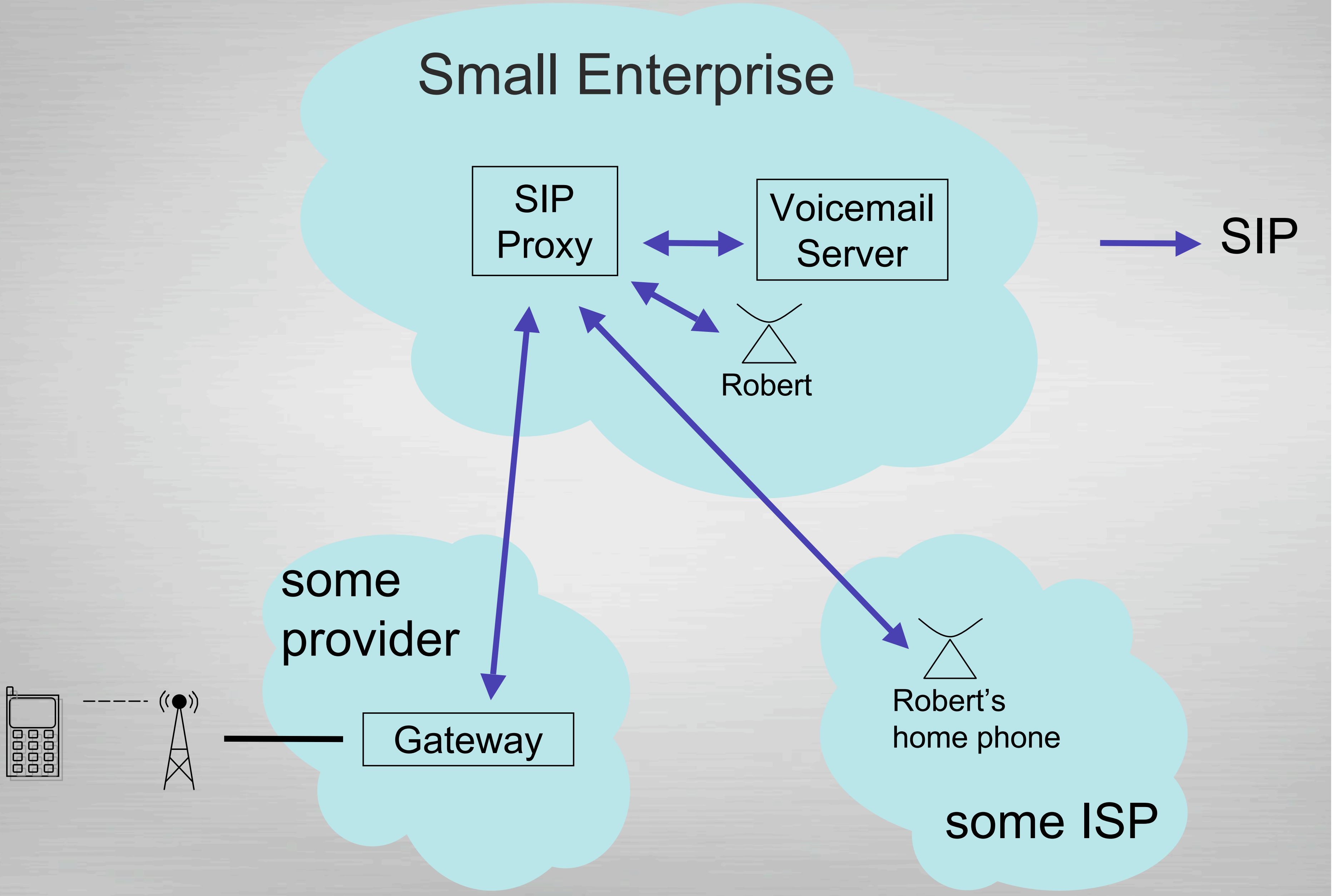
Bending the Architectures

Back-to-Back User Agents

(also called Session Border Controllers or SBCs)



Example Deployment



The Problem

- SIP and RTP were originally designed for an end-to-end transparent network
- NATs, Firewalls, and other elements (sometimes even SBCs) violate that assumption to the point that SIP or media fails

The Problem

- NATs
  - Change the apparent source address, and sometimes port, of packets
    - SIP puts these addresses in the IP packet payload, where NATs can’t “fix” them
      - Via, Contact, SDP c= lines
  - Prevent incoming TCP connections
  - Prevent incoming UDP unless you’ve sent traffic establishing a binding
    - Many different types of binding behavior

The Problem

- Firewalls
  - Tend to prevent all incoming traffic
  - Sometimes allow “pinholes”
    - no standard way (yet) to manage them
    - tend to close them without warning or notice
- SBCs
  - Have to be explicitly aware of any new places a protocol might need to be “fixed”
    - Tend not to forward *any* bits they haven’t been explicitly told to forward
    - Hinders deployment of new features

The Tools

- rport, received, symmetric rtp
  - Change protocol behaviors to avoid NAT damage
- STUN
  - Allows a client to discover what it looks like on the outside
- TURN
  - Reflects packets at globally reachable location
- outbound
  - “Nails up” a connection to something others can reach
- ICE
  - Allows endpoints to discover which of several alternative network traversal strategies work for each call

Within SIP

- Any SIP element receiving a request remembers (by marking the message) the IP address the request appeared to come from
  - UDP: responses will go back to that address
  - TCP: responses go back over the connection the request arrived on
    - But if the connection is gone, the UAS may attempt to open a new connection to that address (will almost never work if there’s a NAT)

Within SIP (rport extension)

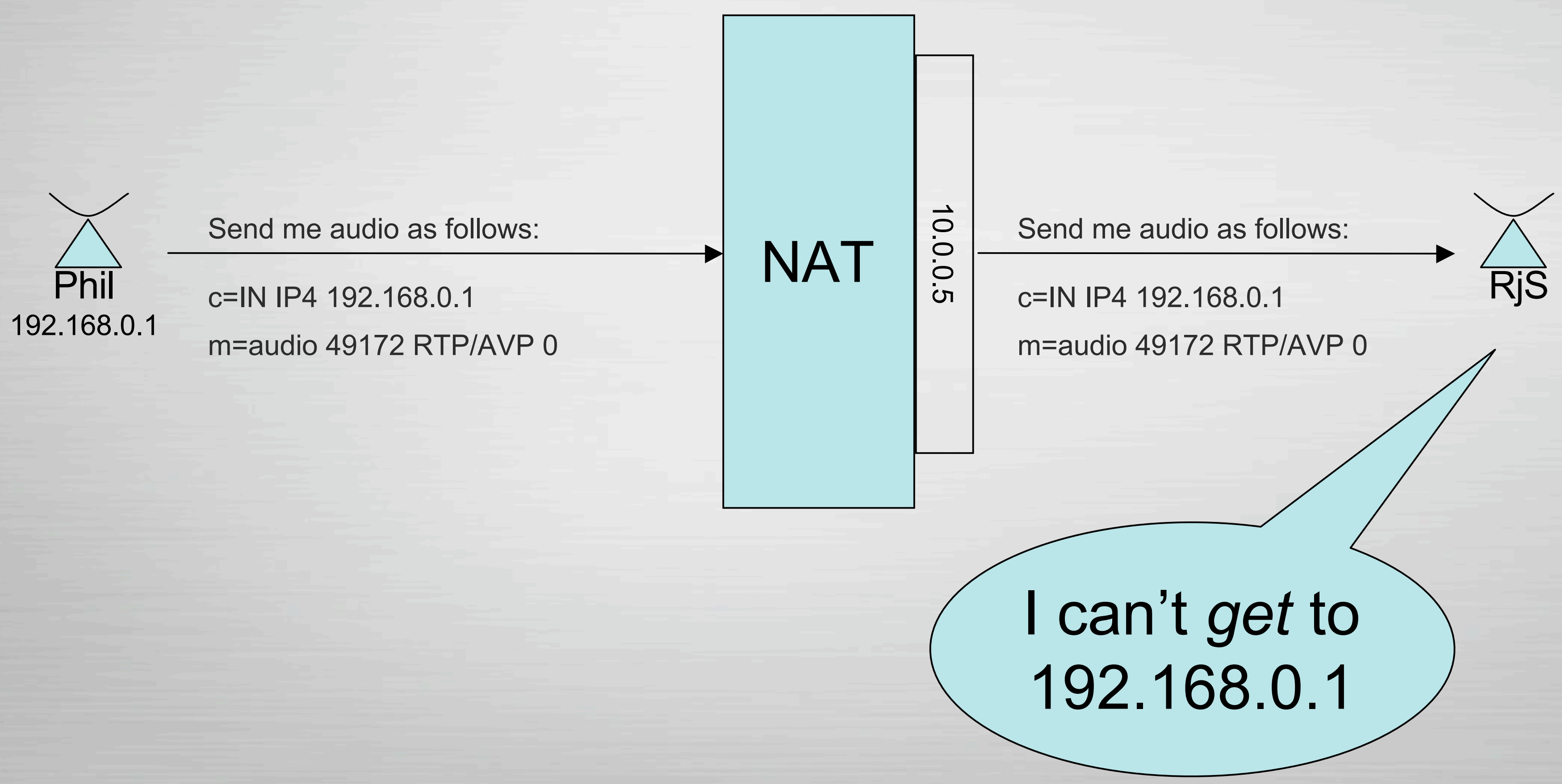
- The rport extension provides for ports what received provides for addresses
  - Requires support from both elements at each hop
  - Receiver remembers the port a request appeared to come from
    - Over UPD, response goes back to that port
    - Over TCP, response goes back to the connection the request arrived over

Media tools

- Symmetric RTP
  - Sending media packets from the same port you have agreed to receive media improves the likelihood of traversing certain NATs
  - This behavior is just done, not signaled

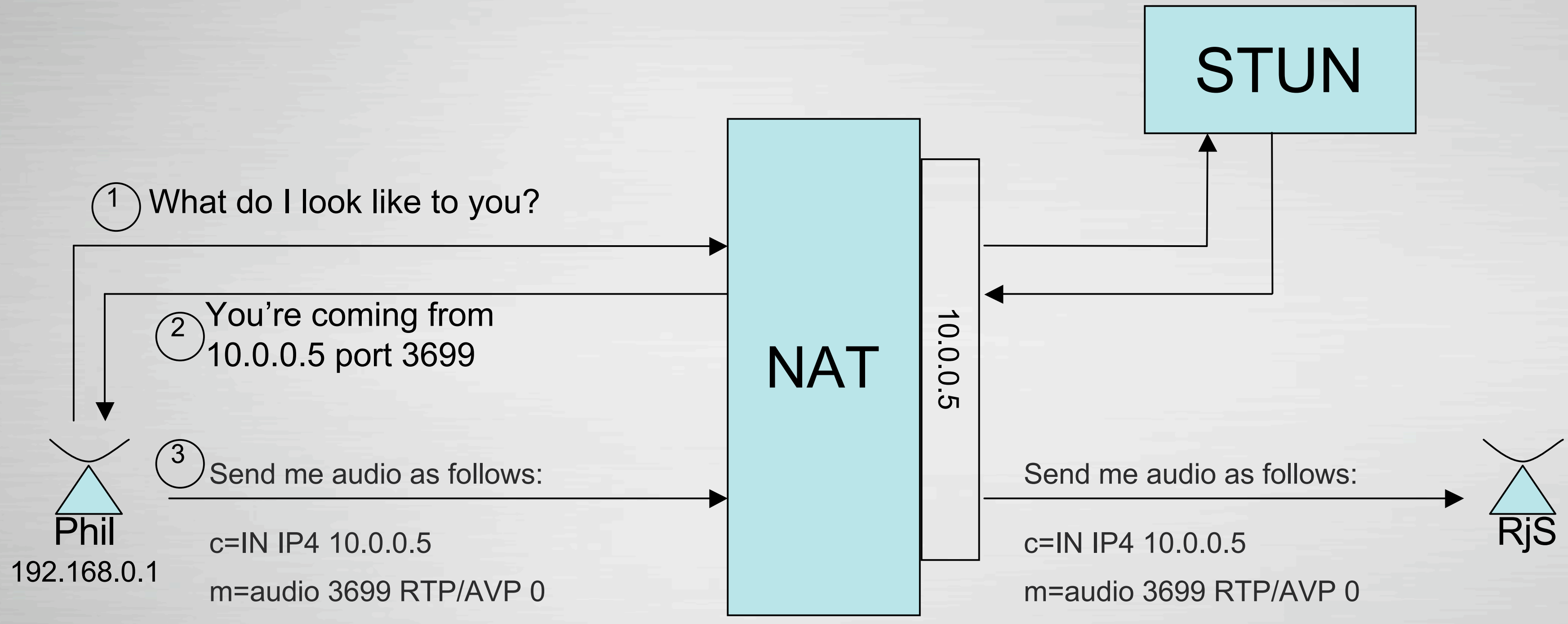
STUN

- SDP offers contain an address for receiving media



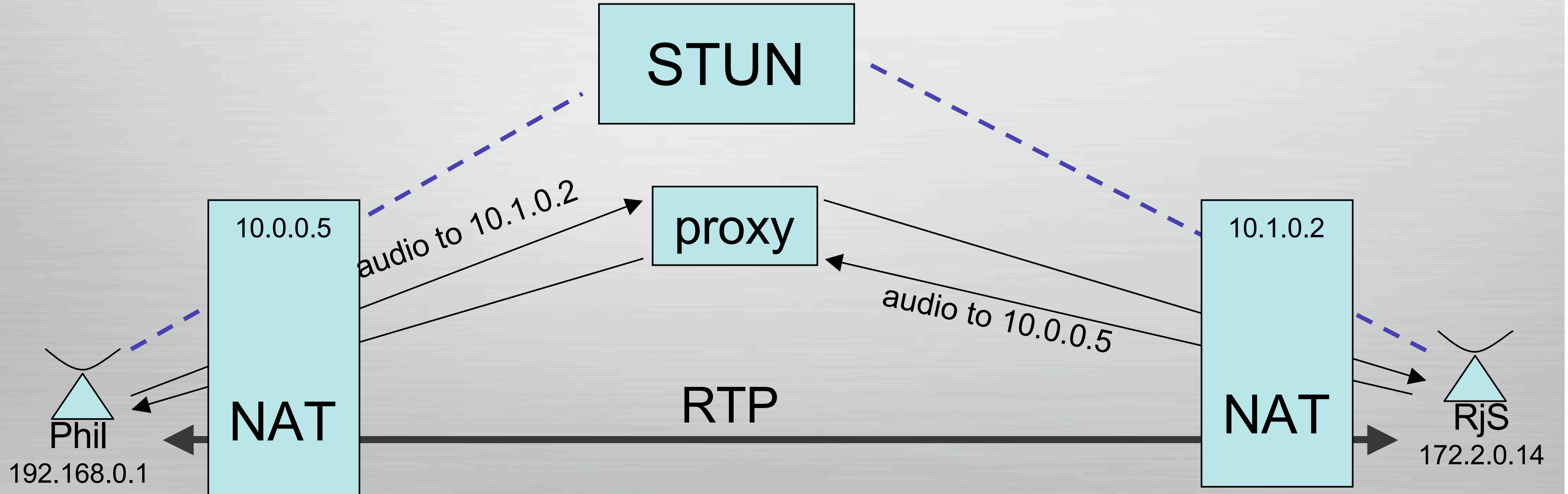
STUN

- STUN lets Phil discover what his address looks like to RjS



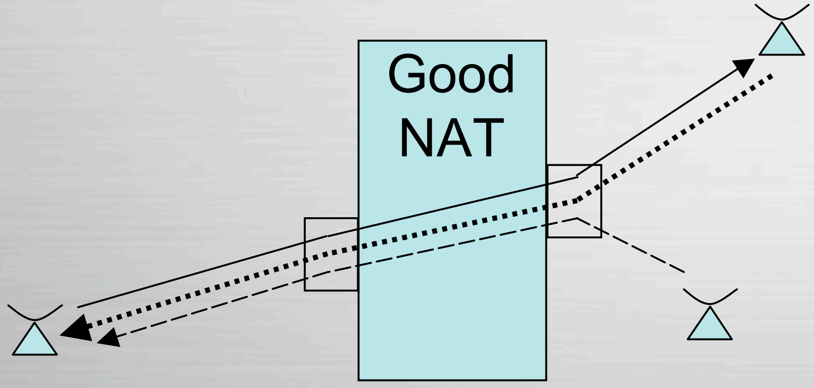
STUN

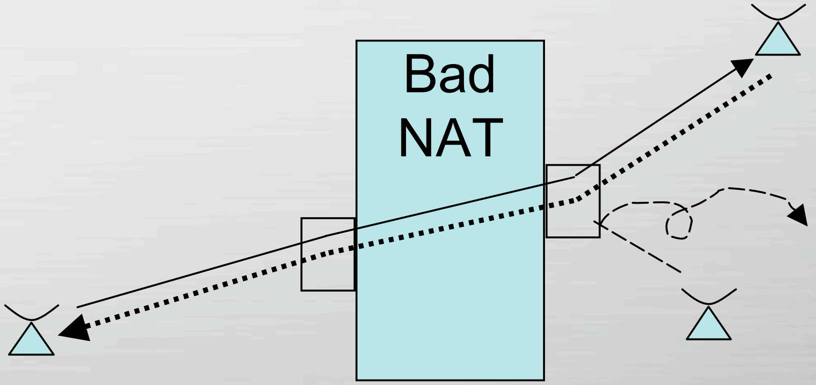
- Place the discovered address in
  - The SDP c= line
  - The Contact URI (if a domain name isn't appropriate)
- Allows traversal of a huge portion of NATed space
  - Even if both RjS and Phil are behind NATs



STUN

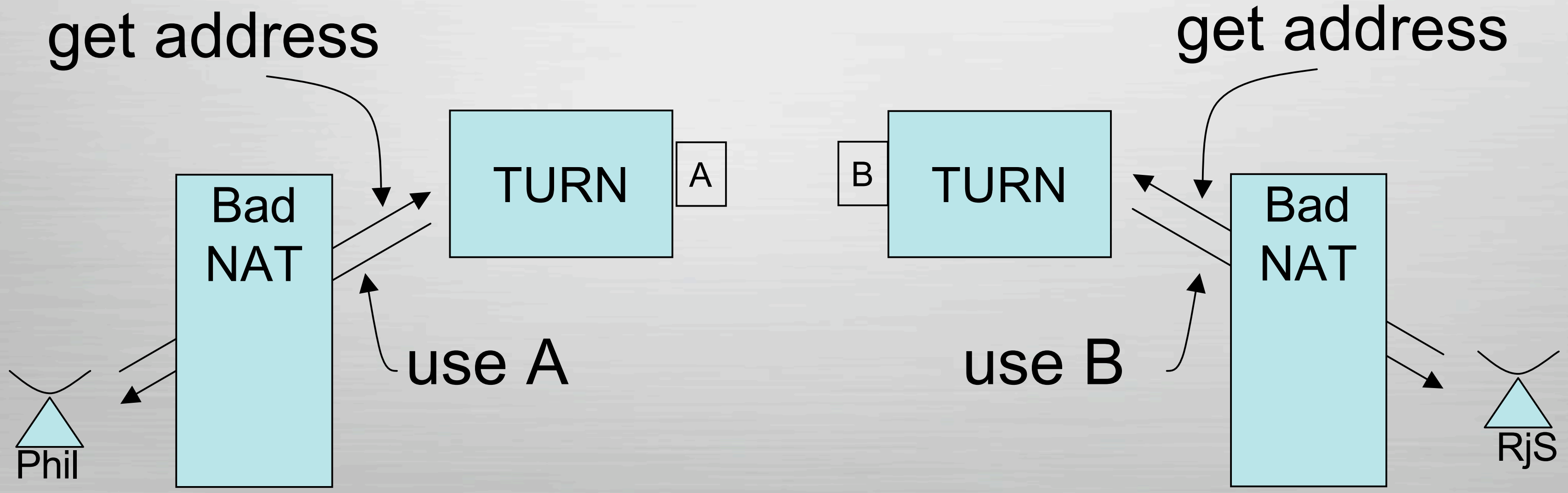
- Doesn’t help with NATs that bind so that only the destination of the of the packet creating the binding can send packets back to the source





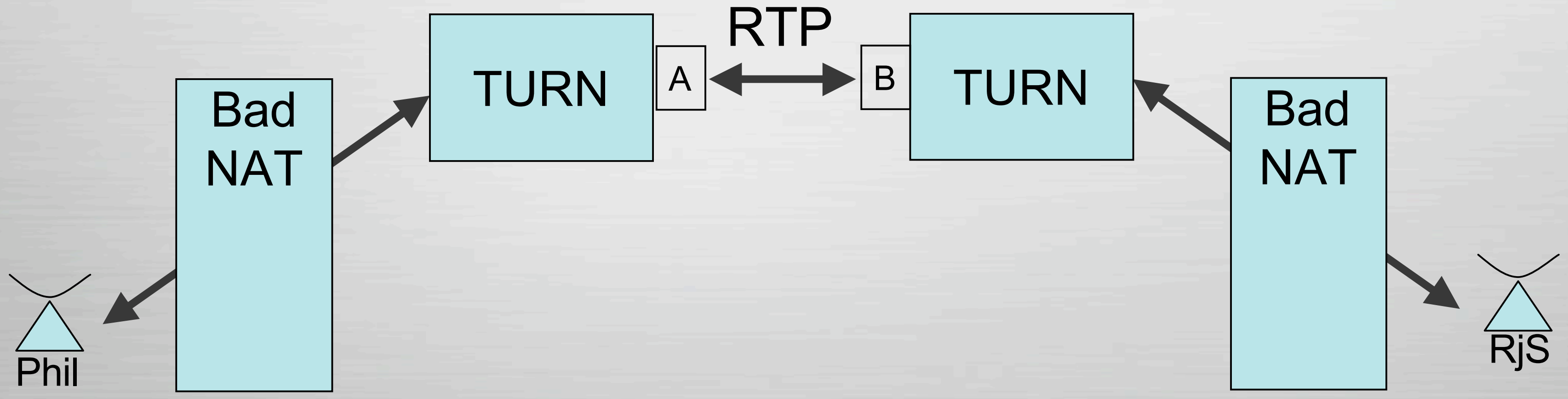
TURN

- Traversal using Relay NAT
  - Allows a client to request an address on a public interface and have media relayed to and from that address



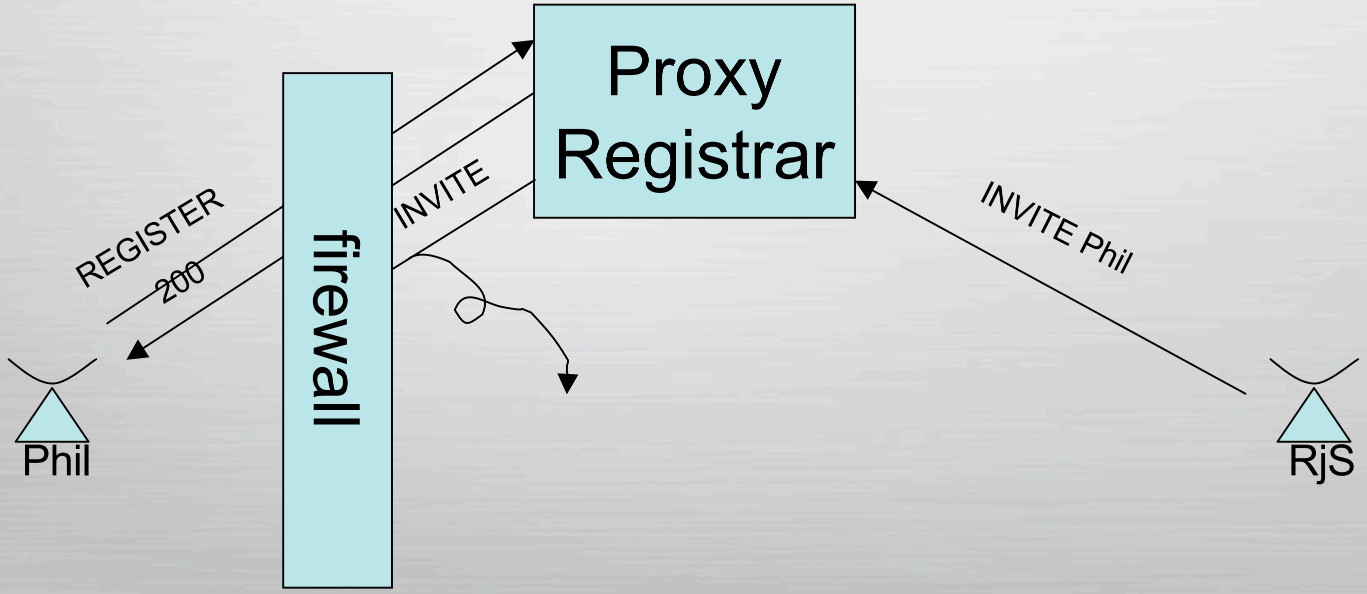
TURN

- Phil offers to receive media at A
- Robert answers asking to receive media at B



Outbound

- Clients behind some NATs and most firewalls can't accept a TCP connection or receive a UDP packet from a host they haven't sent to
  - Nobody can call Phil

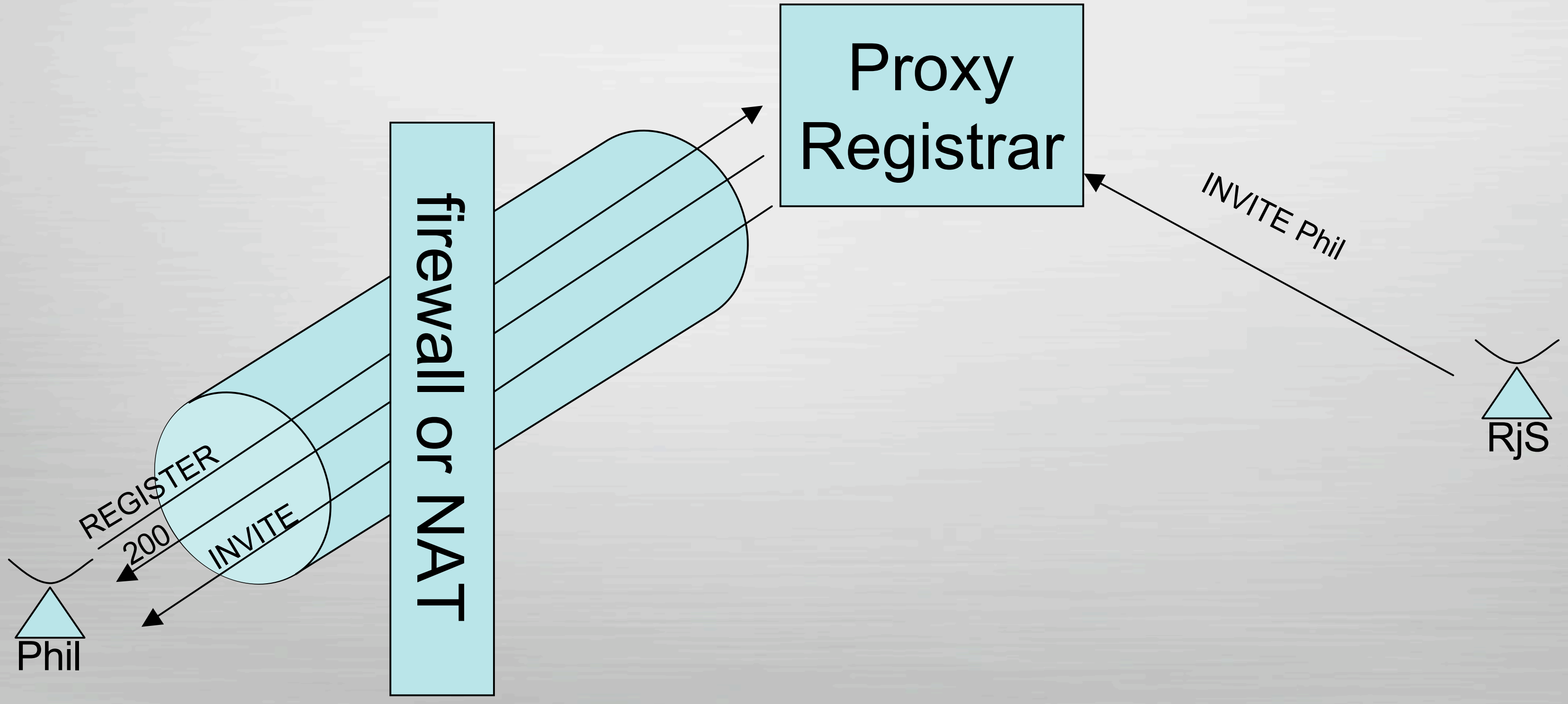


Outbound

- The outbound extension “nails up” a connection, or flow, between Phil and his proxy-registrar
  - Can be UDP or TCP (or anything else that carries SIP)
  - The proxy agrees to send all traffic for Phil down the outbound connection

Outbound

- Phil keeps his connection alive by periodically exchanging traffic with the proxy (STUN for UDP, CRLF for TCP)



ICE

- Phil may have many addresses to use as alternatives for media
  - Native interface address
  - VPN address
  - STUN discovered address
  - addresses acquired using TURN
- One might work with Ben, but only a different one with Robert

ICE

- The Interactive Connectivity Establishment Framework (ICE) allows Phil to
  - Offer all the addresses he has to Ben and Robert
  - Test the addresses they give back to see which one works the best

ICE

- Phil places all his addresses as alternatives in an SDP offer, ordered by preference
  - A direct connection is much better than one using a TURN relay
- Ben and Robert return all their alternatives in their answers
- Everyone starts testing the alternatives (ordered by preference) by trying STUN requests

ICE

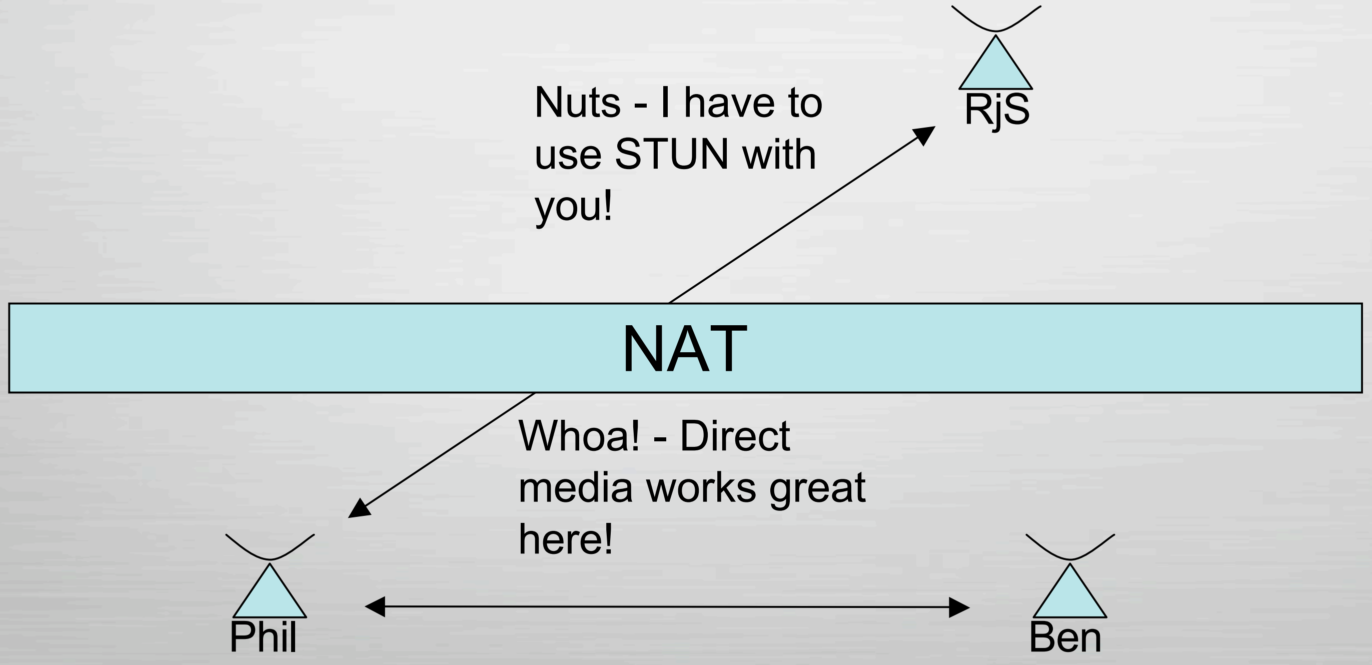
- Each alternative starts as a candidate, prioritized by the requested preference
- A successful STUN transaction between a local and remote candidate makes the pair “valid”
  - This transaction may expose a new address that should be considered as a candidate
- ICE can stop as soon as there is a valid pair for each media stream
  - It’s legal to keep trying to find a better pair

ICE

- There may be multiple valid pairs for a stream
- The ICE “controlling” endpoint indicates which pair to use for each stream by sending a STUN request with a nominating flag
  - The “controlling” endpoint is almost always whoever sent the offer
- ICE stops when all streams have a nominated pair
- ICE can be restarted for any stream at any time by issuing a new offer (changing certain media stream level attributes)

ICE

- Phil can place a single request
  - that forks to Robert and Ben
  - either of which (or both) can answer and have media work, even though their address requirements are wildly different



Nuts - I have to use STUN with you!

Whoa! - Direct media works great here!

Robert Sparks

mailto:RjS@estacado.net

mailto:RjS@nostrum.com

sip:RjS@estacado.net