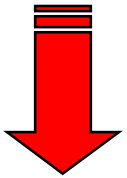
A Waypoint Service Approach to Connect Heterogeneous Internet Address Spaces

T. S. Eugene Ng Hui Zhang Ion Stoica
Carnegie Mellon University UC Berkeley

The Problem in a Nutshell

32-bit IP (IPv4) address space is too small



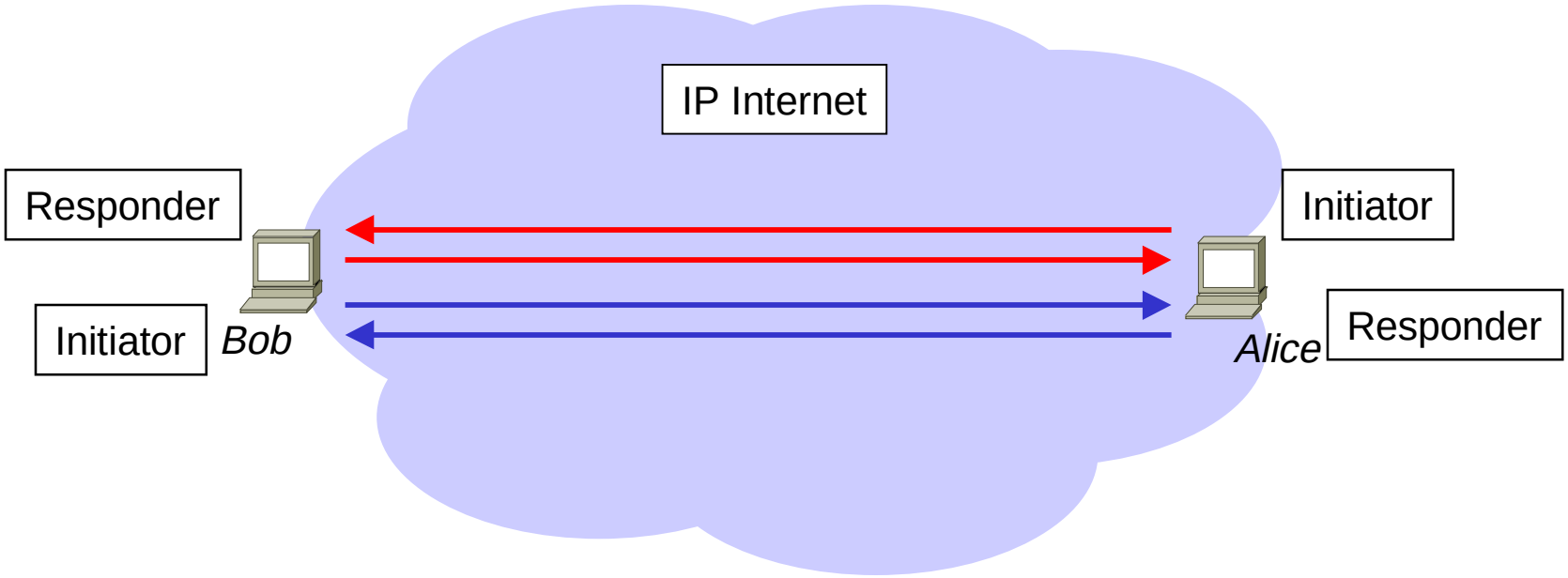
Internet becomes a heterogeneous network

Bi-directional connectivity between hosts is lost

Outline

- Formulate the problem
- Solution design goals
- Proposed solution: AVES
- Implementation and performance

The Original IP (IPv4) Internet



- Every host has a globally unique IP address
- Bi-directional connectivity is a fundamental property

32-Bit IP Address Space Is Too Small

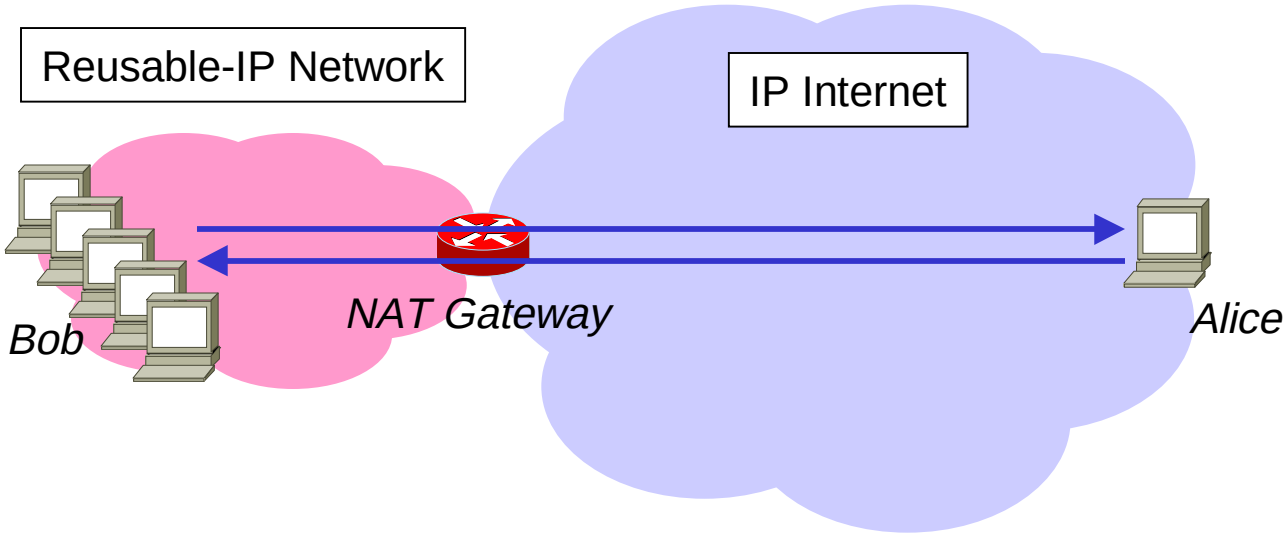
- Upper bound: 31% of IP address space is covered by aggregated routing table
- Poor utilization
- Increasing demand
  - Always-on access (e.g., DSL, cable modem)
  - Internet enabled devices (e.g., mobile phones, PDAs)

- Fear of exhaustion leads to aggressive conservation
- IP addresses are increasingly difficult to obtain

Two Trends

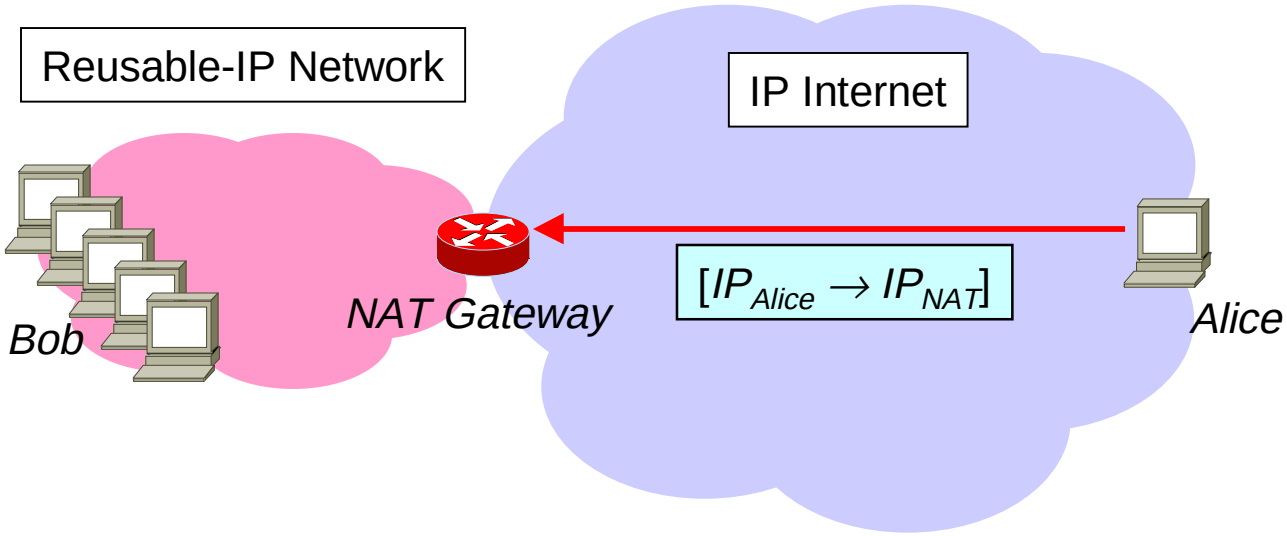
- Deploy networks using reusable-IP addresses
  - a.k.a. private-IP addresses
  - IP network prefixes 10/8, 172.16/12, 192.168/16
  - not globally unique, not routable
  - hosts and routers remain running IP
- Deploy networks using IPv6 addresses
  - enormous 128-bit address space
  - globally unique
  - hosts and routers run IPv6

Using Reusable-IP Addresses



- NAT (Network Address Translation) gateway may have only one IP address
  - shared by reusable-IP network hosts
- NAT provides reusable-IP to IP connectivity

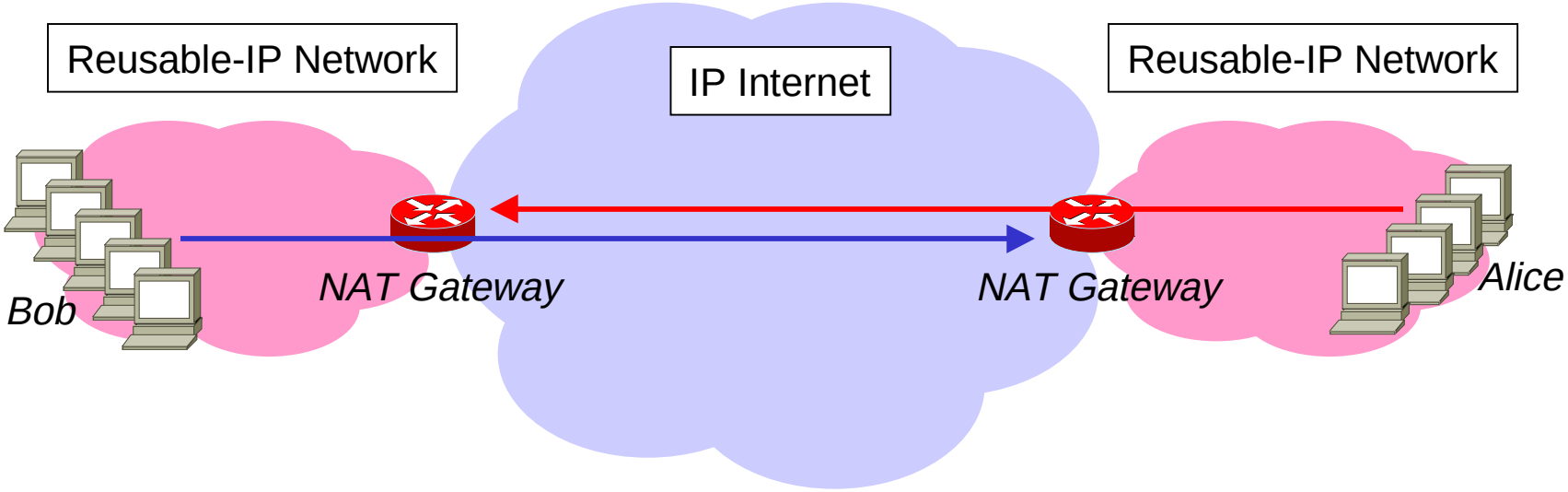
The Problem with NAT



- Bob has no globally unique IP address and so Alice cannot directly address Bob

NAT cannot provide bi-directional connectivity

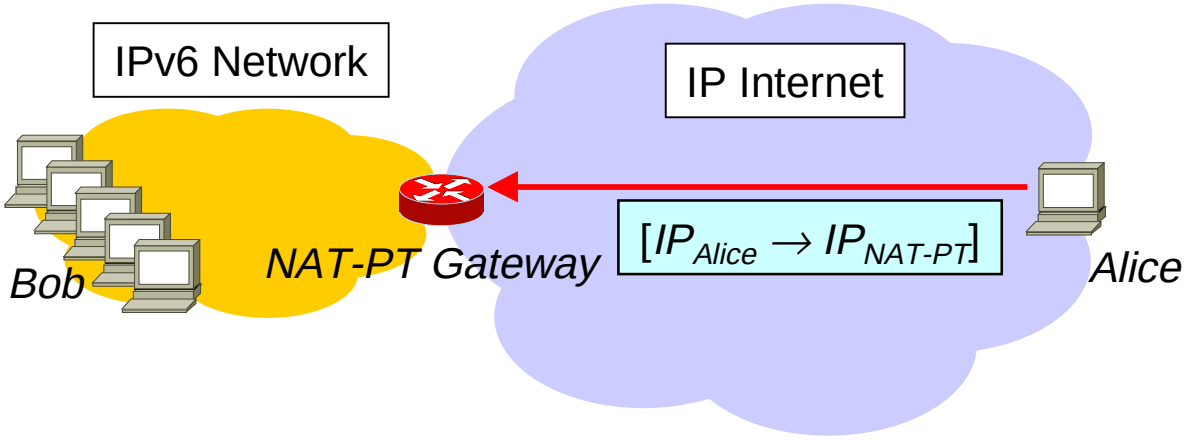
Even Worse



- No connectivity between Alice and Bob at all!

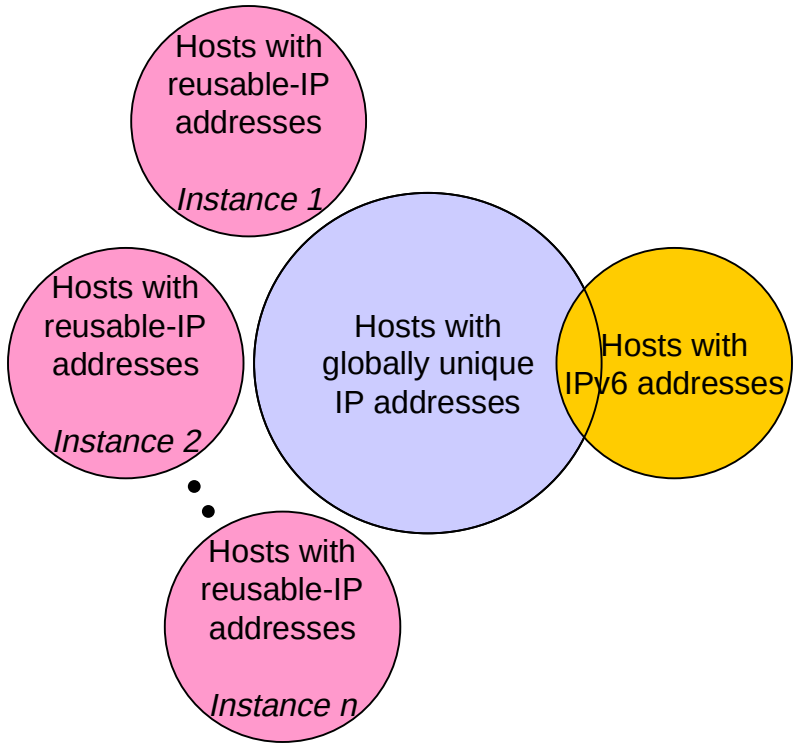
Using IPv6 Addresses

- IPv6 can be fully compatible with IP
- Key: every IPv6 host must consume a globally unique IPv4 address!
- In reality, many IPv6 networks will be IPv6-only
  - connect to IP Internet via NAT-Protocol Translation (PT) gateway
  - NAT-PT has the same problem as NAT



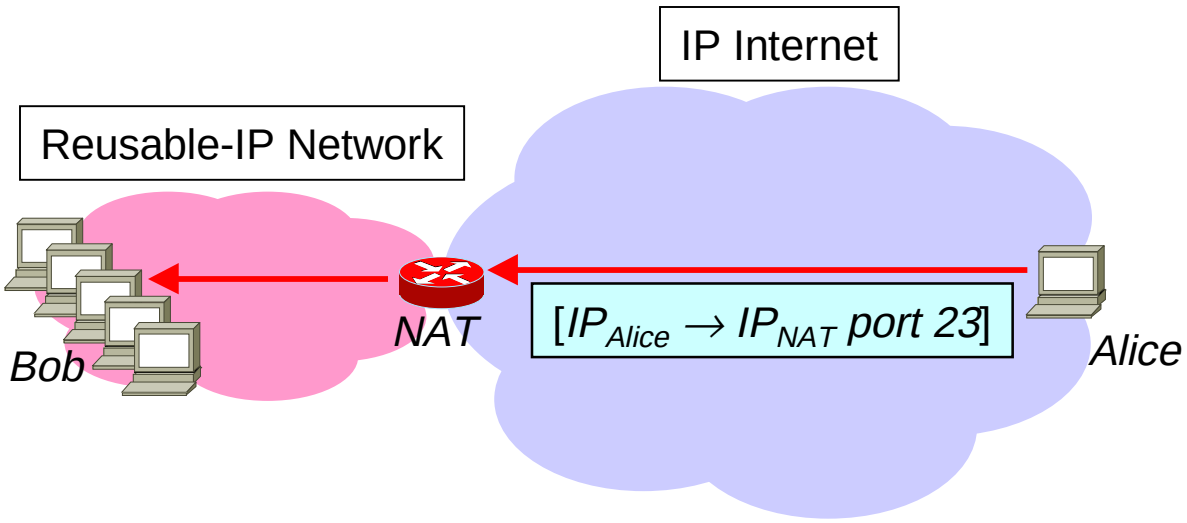
Heterogeneous Internet Address Spaces

- NAT and NAT-PT cannot provide bi-directional connectivity
- Key problems:
  IP to reusable-IP and
  IP to IPv6 connectivity
  - all other cases reduce to these
- For simplicity, only consider
  IP to reusable-IP connectivity



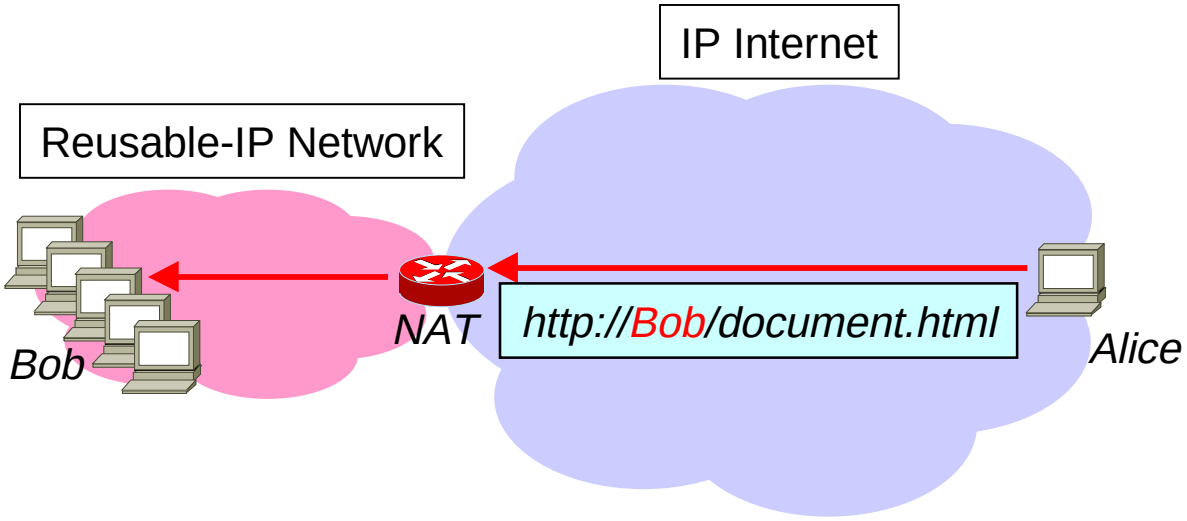
Solution Design Goals

- Should provide general connectivity



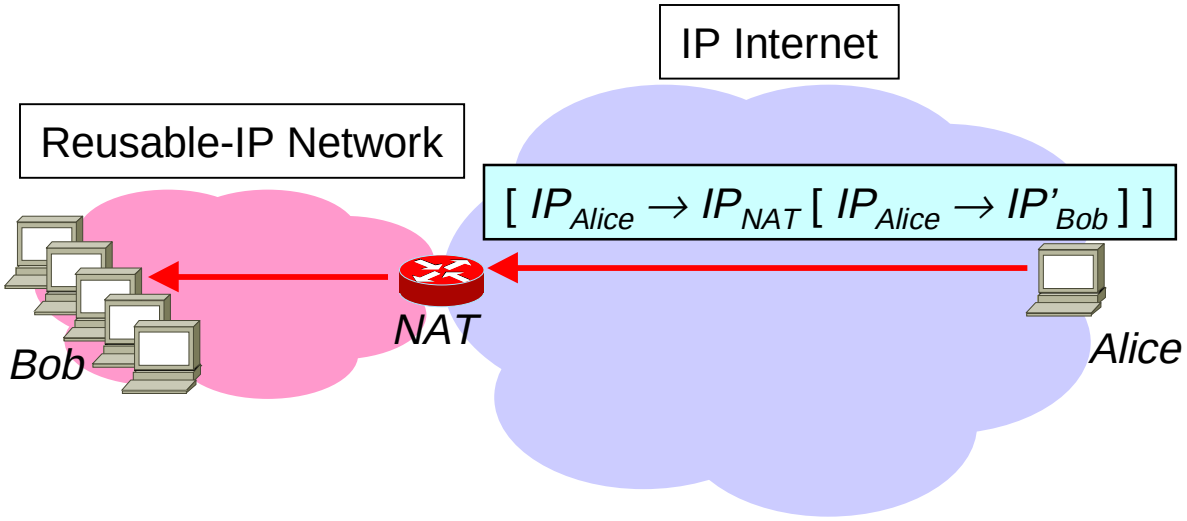
Solution Design Goals

- Should provide general connectivity
- Should be application independent



Solution Design Goals

- Should provide general connectivity
- Should be application independent
- Should not require changes to existing IP hosts and IP network routers
  - there is no incentive for them to make changes



Key Constraints

- Without IP addresses, reusable-IP hosts cannot be addressed by IP hosts
- IP addresses are a scarce resource
- Existing IP hosts and routers will not change

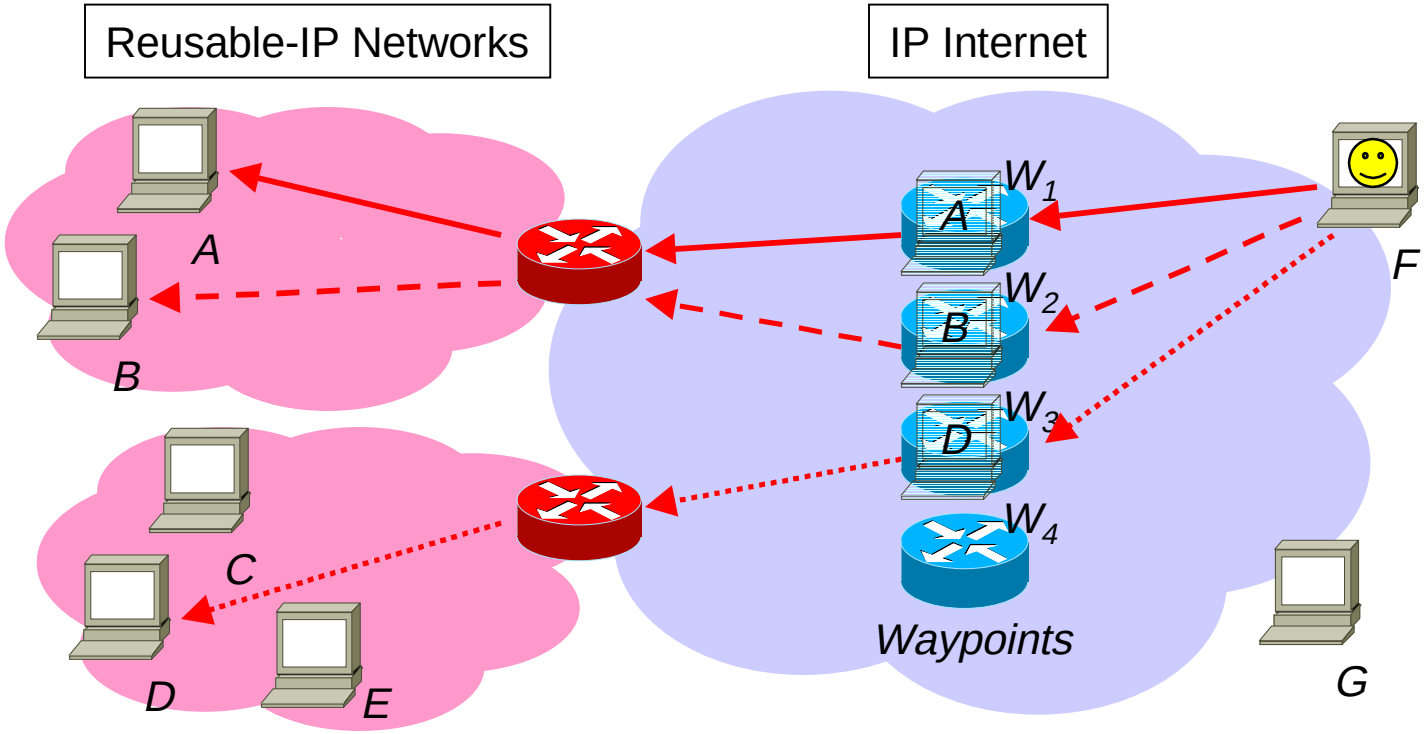
Must efficiently share IP addresses among many reusable-IP hosts simultaneously

Key Insight:
Design a 3rd-party service provider-based solution

Proposed Solution:
AVES: Address Virtualization Enabling Service

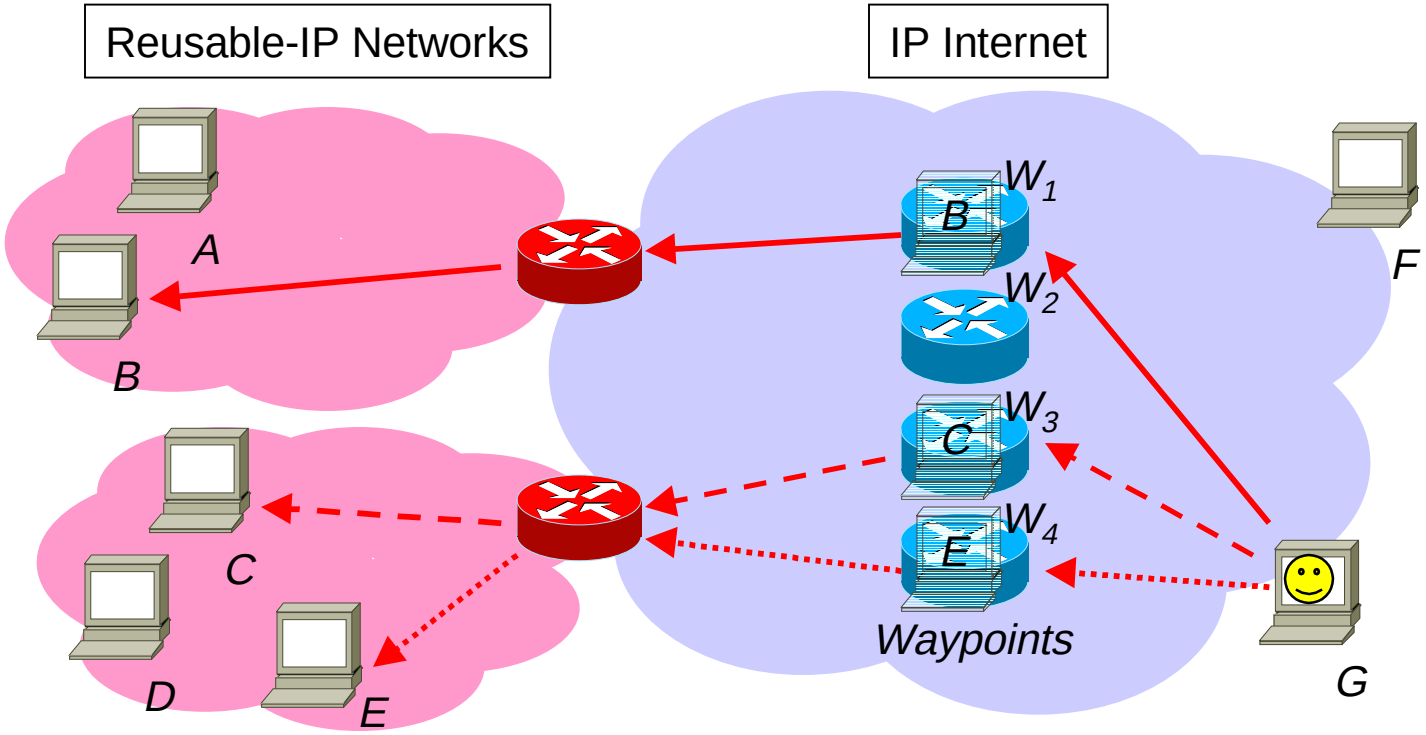
AVES Overview

- Service provider deploys IP agents called waypoints
- Virtualize reusable-IP hosts by the waypoints
- Update customer NAT gateways

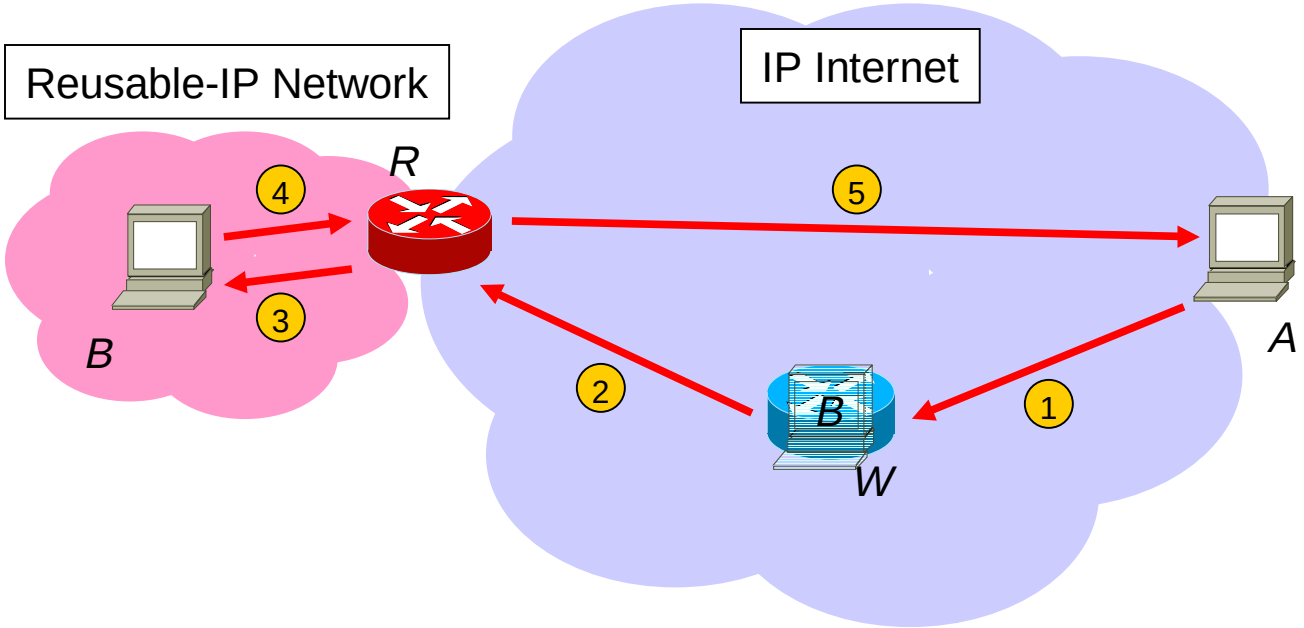


AVES Overview (Continued)

- Each initiator has its own unique virtual map
  - Each initiator can connect to 4 reusable-IP hosts simultaneously



Data Path Operations



Step	Packet sent
1	[$IP_A \rightarrow IP_W$]
2	[$IP_W \rightarrow IP_R$ [$IP_A \rightarrow IP'_B$]]
3	[$IP_A \rightarrow IP'_B$]
4	[$IP'_B \rightarrow IP_A$]
5	[$IP_W \rightarrow IP_A$]

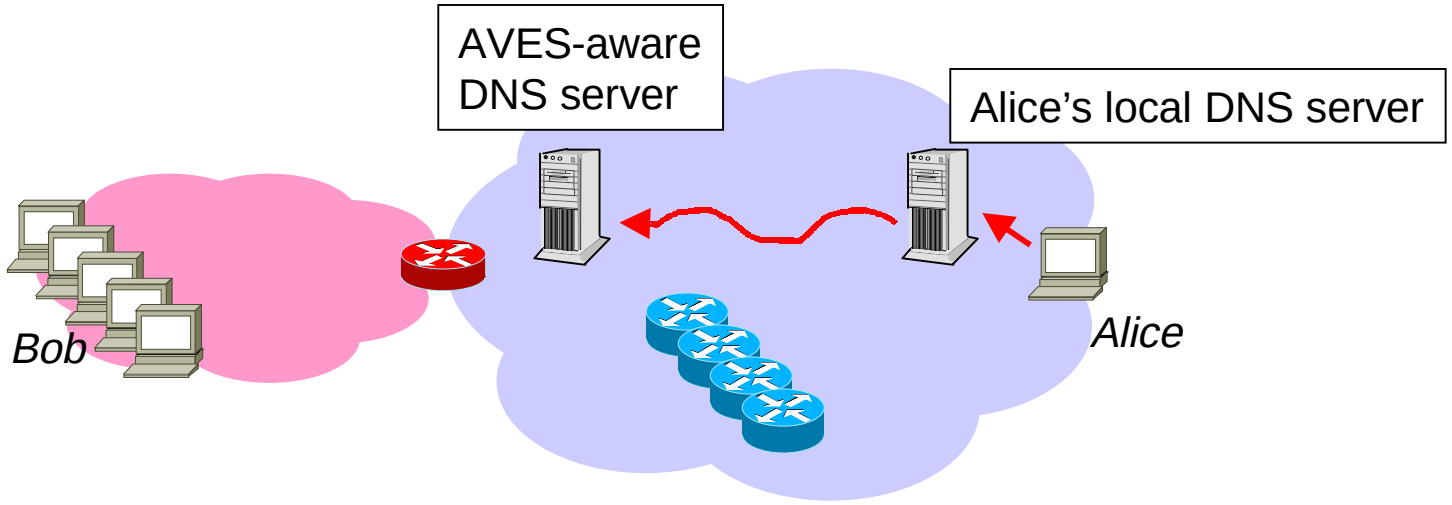
Control Path Operations

- How to dynamically create the reusable-IP host to waypoint virtual mapping for each initiator?

- Fundamentally a reusable-IP host still needs to be identified somehow before communications
- Use a name to uniquely identify a reusable-IP host
- Create waypoint mapping during name resolution

DNS Is Not the Perfect Answer

- Want the identity of the initiator during DNS name resolution
- Recursive DNS name lookup hides this identity



Idealistic Solutions

- Modify the DNS protocol to carry the initiator's IP address in a DNS query
  - also useful for DNS based load balancing
- Run local caching-only name servers on end hosts
  - has performance benefit
- Use an alternative naming system

What Can We Do Today?

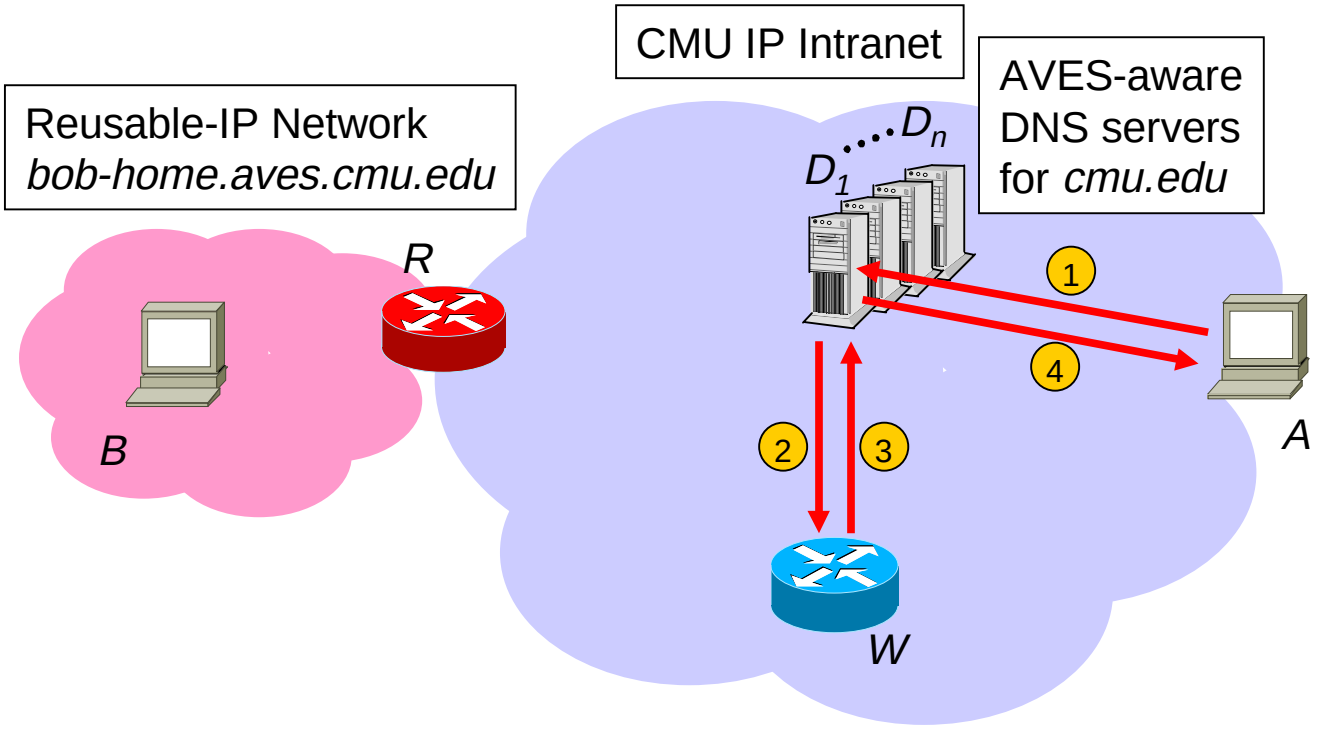
- In some specific deployment scenarios like Intranet deployment the right incentives exist to overcome the initiator identity problem
- When the incentives do not exist, trade performance for deployability

Scenario 1 -- Intranet Deployment

- CMU can deploy AVES so that people working at school can initiate connections back to their home computers behind NAT gateways

- Solution: CMU will upgrade local DNS servers to become AVES-aware
- Since local DNS servers interact directly with initiators, their identities can be known

Scenario 1 -- Intranet Deployment

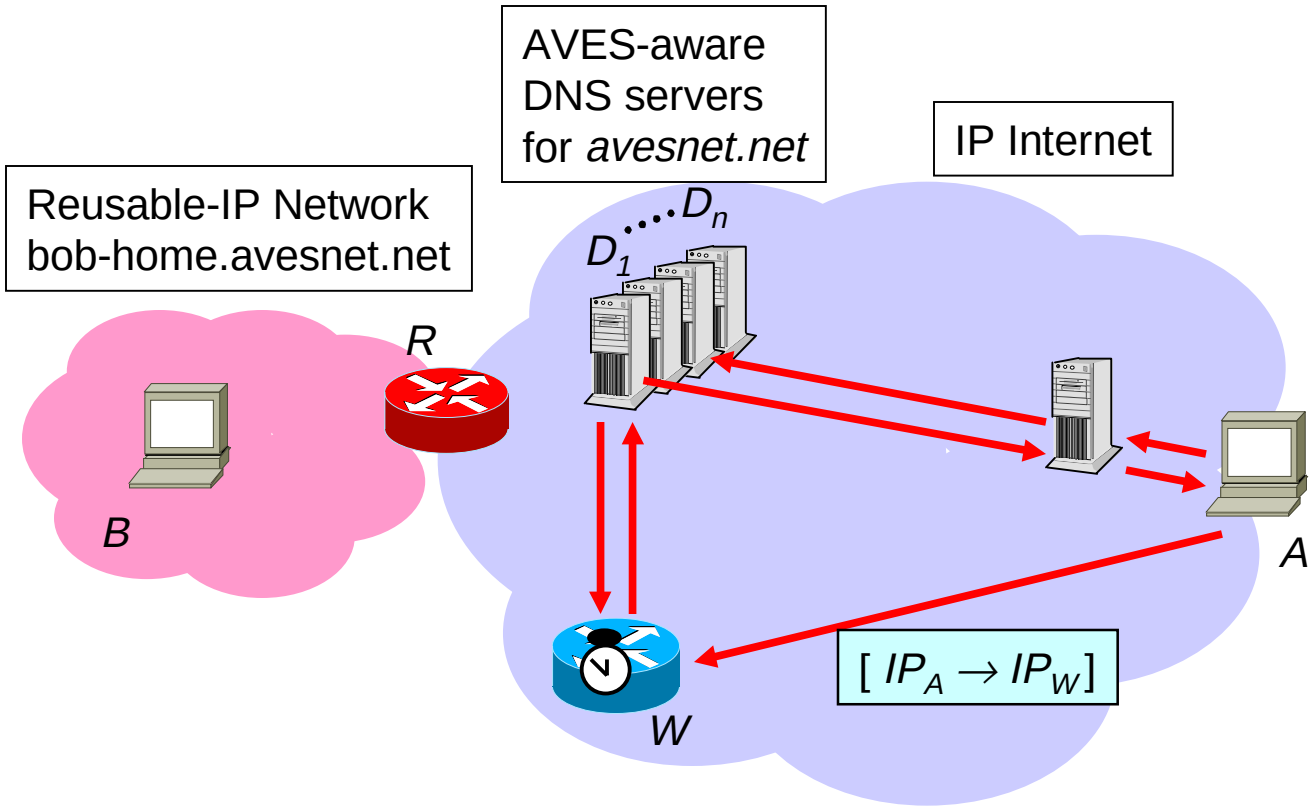


Step	Action
1	DNS query for *B*
2	SETUP message (IP_A, IP_R, IP'_B)
3	ACCEPT message
4	DNS reply for *B* (IP_W)

Scenario 2 -- General Deployment

- Cannot upgrade the local DNS servers used by initiators
- Solution: Delayed binding
  - serialize requests at waypoint
  - trades performance for deployability

Delayed Binding

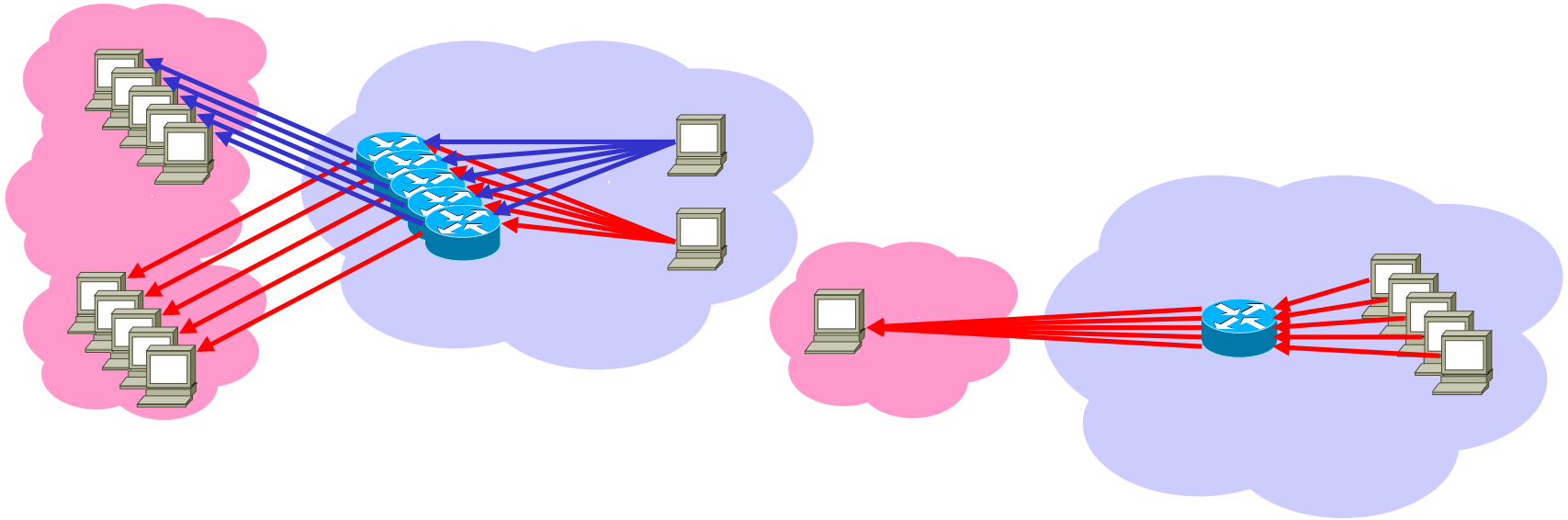


Delayed Binding Is Imperfect

- Significantly lowers the maximum rate at which names can be resolved
  - e.g. with 50 waypoint IP addresses and a wait period of 2 seconds, 25 host-to-host sessions can be created per second
- This is what we have implemented and deployed
  - quite usable so far
  - see our paper for full details

Connectivity Properties

- Using *N* IP addresses, every IP initiator can simultaneously reach up to *N* reusable-IP hosts
- Every reusable-IP host can be reached by an unlimited number of IP hosts

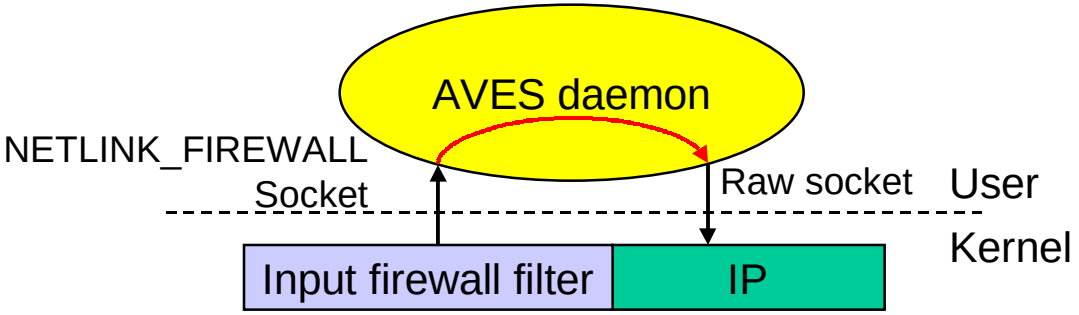


Deployability Properties

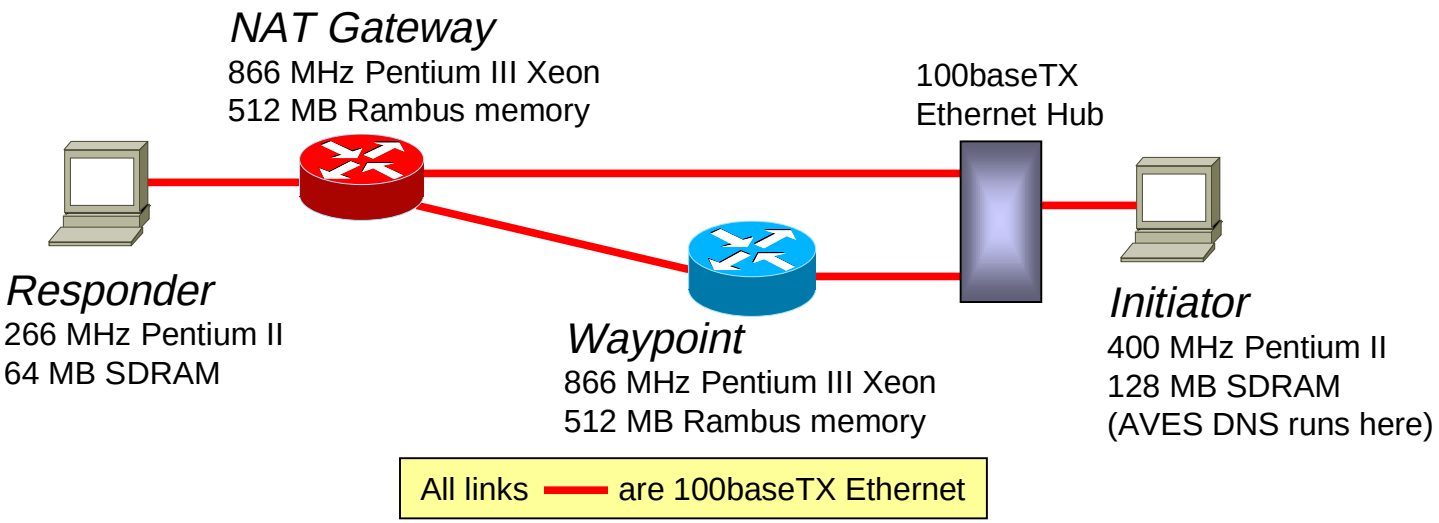
- Waypoints can be easily deployed
- NAT gateways need to be extended to process packets
  - necessary and the right incentive exists
- No change to existing IP hosts or IP network routers
- Intranet deployment
  - upgrading existing local DNS servers provide best performance
- General deployment
  - with delayed binding, no existing DNS server upgrade necessary, but performance is reduced significantly

Implementation

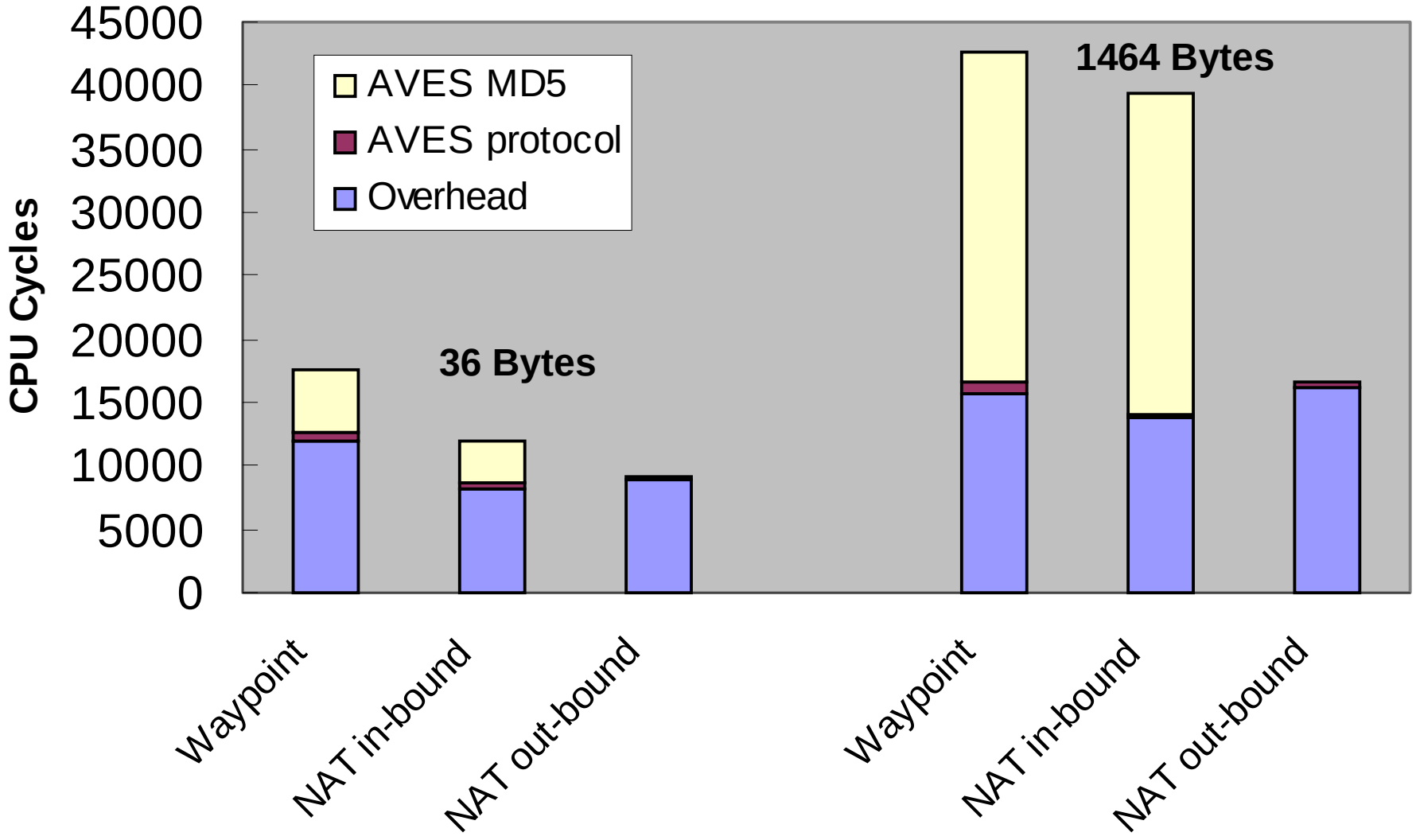
-  AVES DNS Server — Modified *named* running on Linux
-  AVES Waypoint — Linux user-level daemon (with delayed binding)
-  AVES NAT — Linux user-level daemon



Performance Measurement Testbed



Data Path Performance



Data Path Performance

- Theoretical maximum throughput 233Mbps with 1464 byte UDP packets
  - probably higher when overhead is amortized over a train of packets
- End-to-end throughput experiments
  - 96 Mbps with 1464 byte UDP packets
  - 80 Mbps with 1464 byte TCP packets
  - 41 Mbps with 48 byte TCP packets
  - could not get result for 48 byte UDP due to problem with Intel EtherExpress Pro driver

Prototype System

- Registered domain name avesnet.net
- 50 waypoint IP addresses assigned to two PCs
- One AVES-aware DNS server
- 10 trial customers
- Applications tested: telnet, ssh, ftp, scp, NFS, httpd, X windows, VNC, ping, traceroute

Summary

- AVES can provide high connectivity from IP hosts to reusable-IP or IPv6 hosts without
  - consuming many IP addresses
  - changing existing IP hosts or IP network routers
- Can provide connectivity even when both initiator and responder are behind NAT or NAT-PT
  - more sophisticated proposed solutions (IPNL, TRIAD) exist
- Optimized for deployability

Summary (Continued)

- Explore different ways of using 3rd-party agents to add functionality to the difficult to change Internet infrastructure
  - many previous application level services: web caches, CDN
  - AVES provides a fundamental addressing service
- http://www.avesnet.net
  - online demo
  - source code (really really soon)