Internet Addressing and Naming

CS 7260

Nick Feamster

January 10, 2007

Announcements

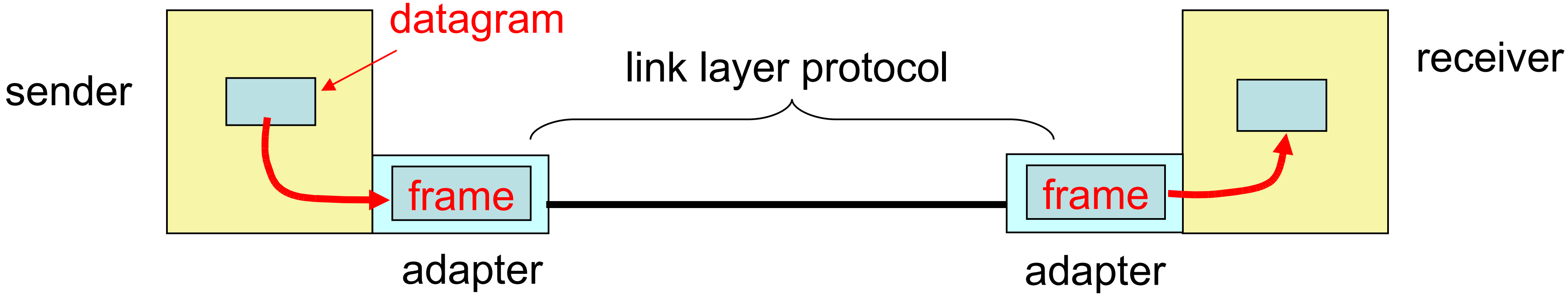
- Course mailing list
  - cs7260-course at mailman.cc.gatech.edu
  - https://mailman.cc.gatech.edu/mailman/listinfo/cs7260-course
- Wiki should be up soon (we hope)
- TA: Keshav Attrey (attrey@cc.gatech.edu)

Today: Addressing and Naming

- Internet **Addressing**
  - **Step 1:** Connecting a single network
  - **Step 2:** Connecting networks of networks
    - IPv4 Addressing
      - Structure
      - Scaling problems and CIDR (1994)
      - Allocation and ownership
      - Longest prefix match and Traffic Engineering
      - Issues and design questions
  - More scaling problems and solutions
- Internet **Naming**
  - Today: DNS and the naming hierarchy
  - Research: Flat names
- Paper discussion: Jung *et al.*

Bootstrapping: Networks of Interfaces

- LAN/Physical/MAC address
  - Unique to physical interface (no two alike)
  - Flat structure



- Frames can be sent to a specific MAC address or to the broadcast MAC address

What are the advantages to separating network layer from MAC layer?

ARP: IP Addresses to MAC addresses

- Query is IP address, response is MAC address
- Query is sent to LAN’s broadcast MAC address
- Each host or router has an ARP table
  - Checks IP address of query against its IP address
  - Replies with ARP address if there is a match

Potential problems with this approach?

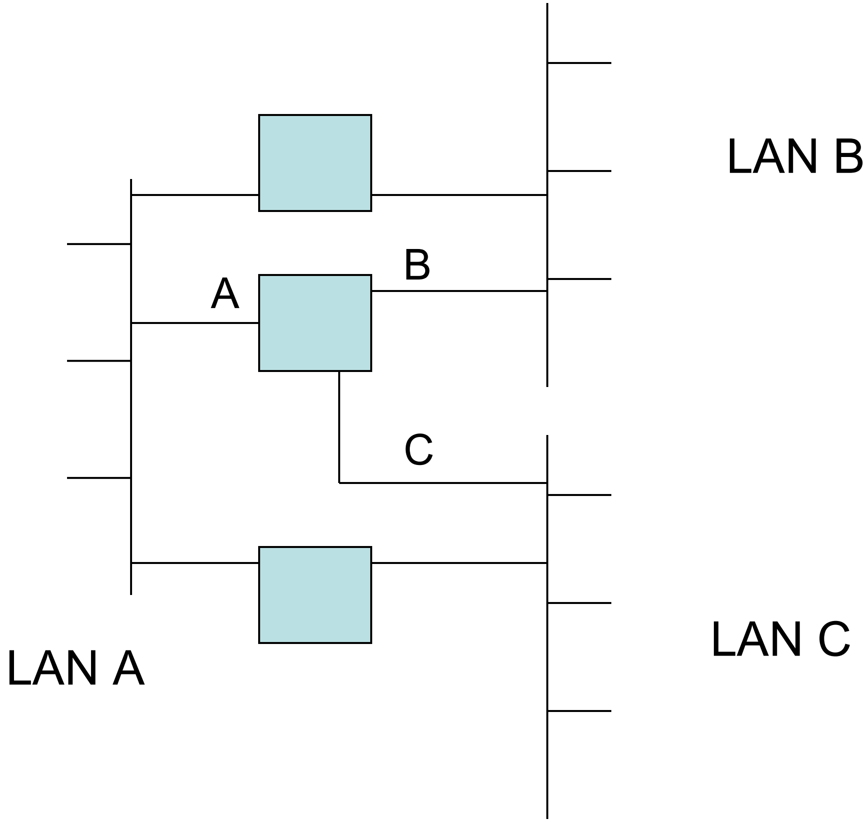
- Caching is key!
  - Try arp –a to see an ARP table

Interconnecting LANs: Bridging

- Receive & broadcast ("hub")
- Learning
- Spanning tree (RSTP, MSTP, etc.)

Learning Bridges

- Bridge builds mapping of which port to forward packets for a certain MAC address



- If has entry, forward on appropriate port
- If no entry, flood packet

Potential problems with this approach?

Virtual LANs (VLANs)

- A single switched LAN can be partitioned into multiple "colors"
- Each color behaves as a separate LAN
- Better scaling properties
  - Reduce the scope of broadcast storms
  - Spanning tree algorithms scale better
- Better security properties

IPv4 Addresses: Networks of Networks

Topological Addressing

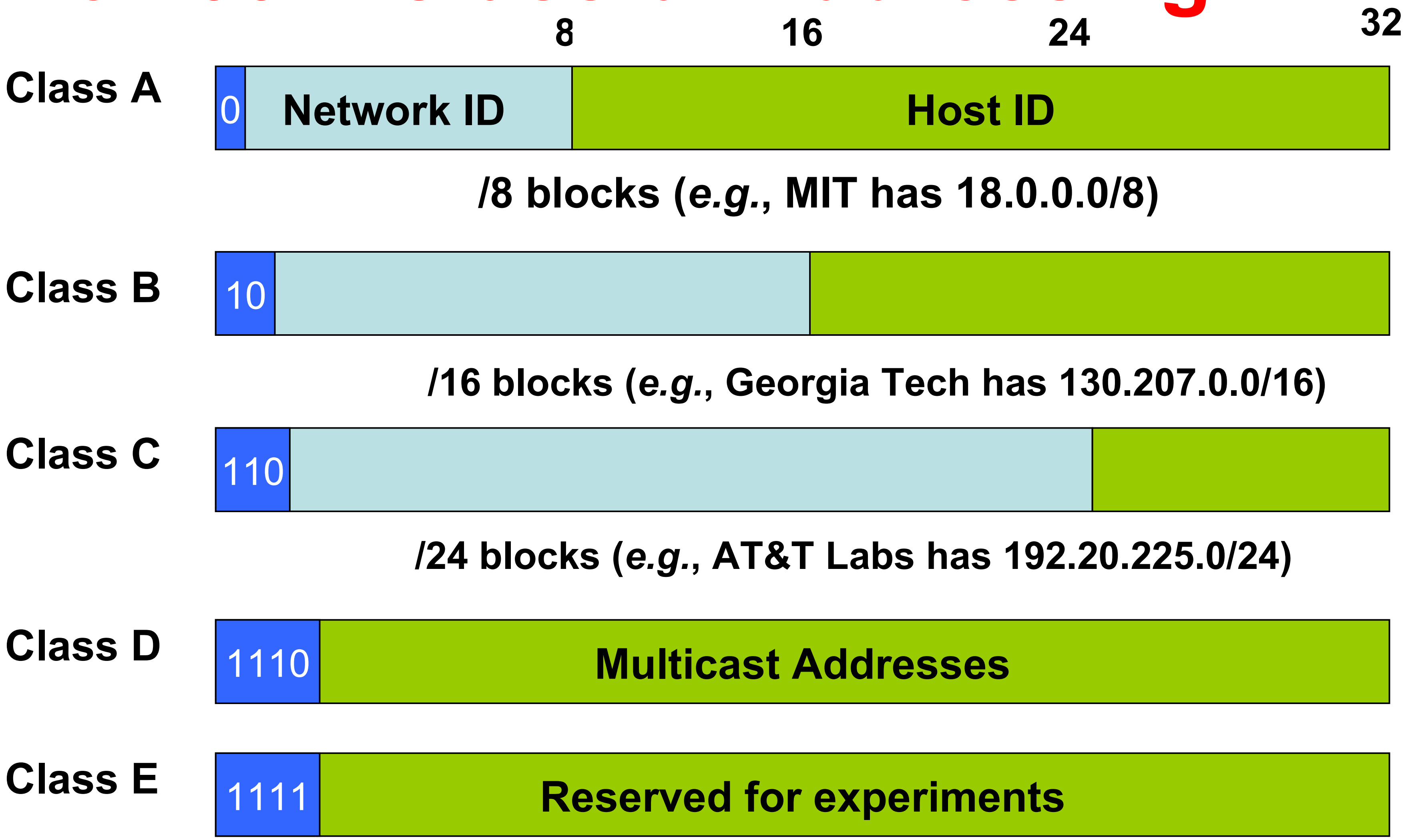
- 32-bit number in “dotted-quad” notation
  - www.cc.gatech.edu --- 130.207.7.36

130	207	7	36
10000010	11001111	00000111	00100100

Network (16 bits) **Host (16 bits)**

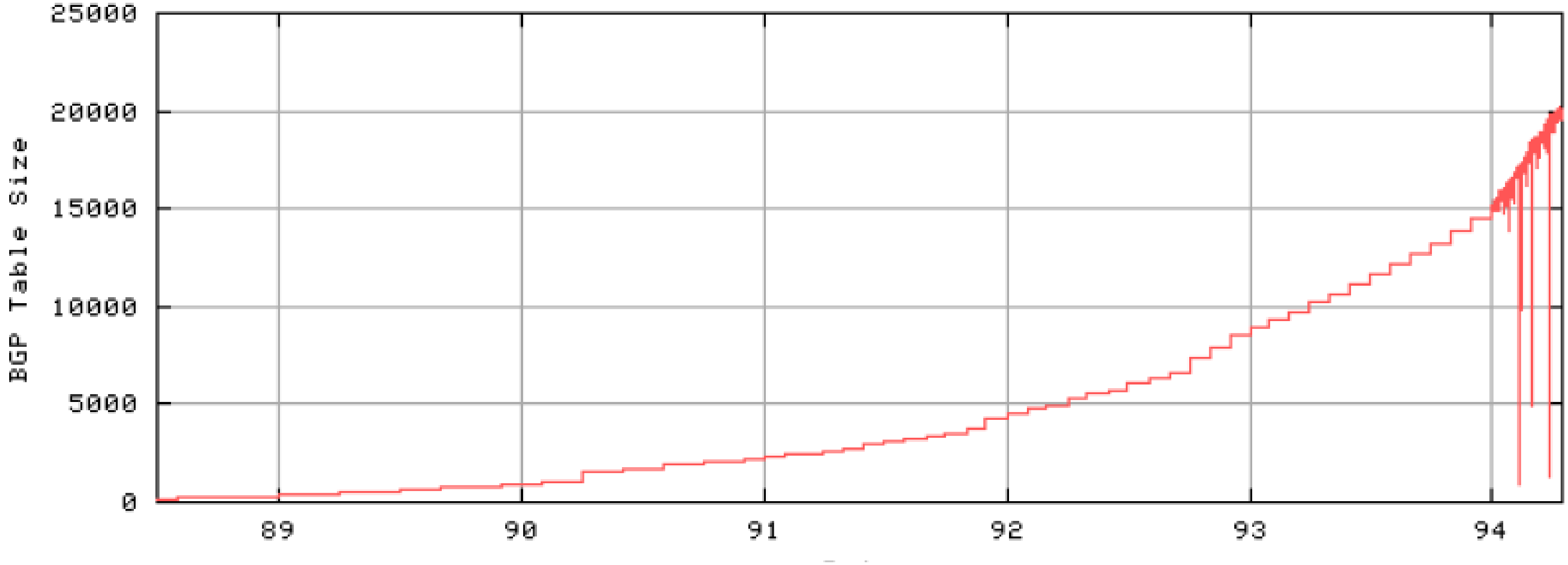
- **Problem:** 2^{32} addresses is a lot of table entries
- **Solution:** Routing based on network and host
  - 130.207.0.0/16 is a 16-bit *prefix* with 2^{16} IP addresses

Pre-1994: Classful Addressing



Simple Forwarding: Address range specifies network ID length

Problem: Routing Table Growth



Source: Geoff Huston

- Growth rates exceeding advances in hardware and software capabilities
- Primarily due to Class C space exhaustion
- Exhaustion of routing table space was on the horizon

Routing Table Growth: Who Cares?

- On pace to run out of allocations entirely
- Memory
  - Routing tables
  - Forwarding tables
- “Churn”: More prefixes, more updates

Possible Solutions

- Get rid of global addresses
  - NAT
- Get more addresses
  - IPv6
- Different aggregation strategy
  - Classless Interdomain routing

Classless Interdomain Routing (CIDR)

Use two 32-bit numbers to represent a network.
Network number = IP address + Mask

Example: BellSouth Prefix: 65.14.248.0/22

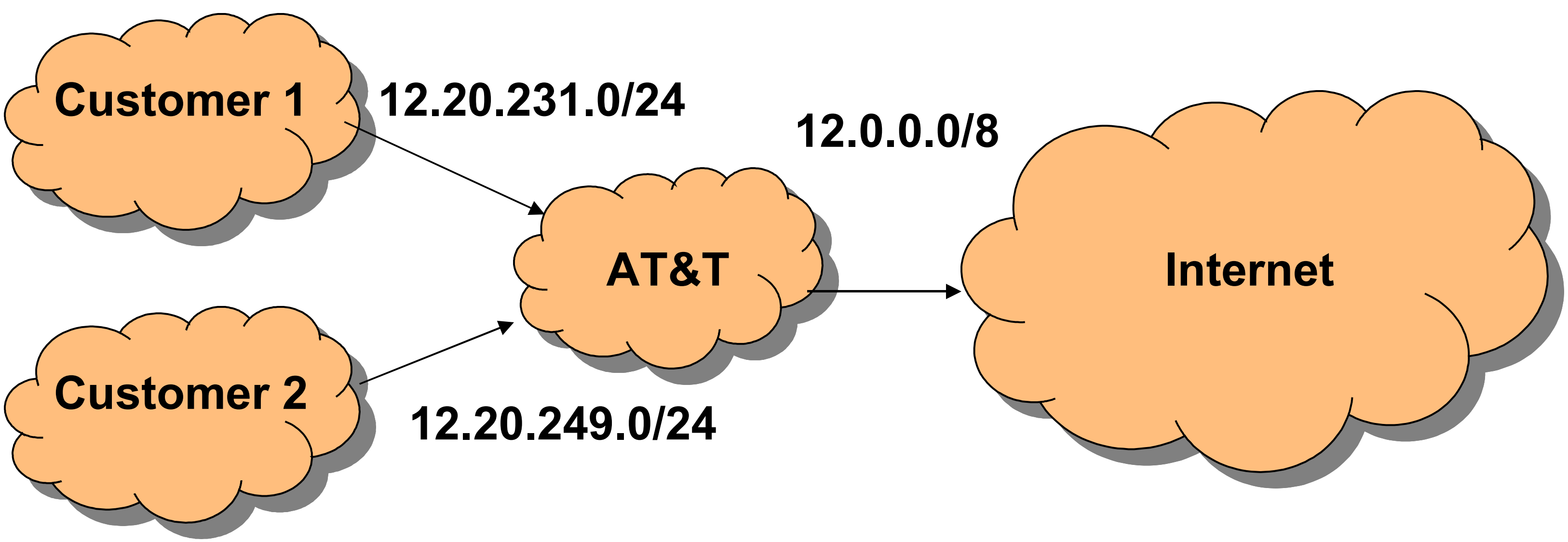
01000001	00001110	11111000	00000000
11111111	11111111	11111100	00000000

IP Address: 65.14.248.0 **"Mask": 255.255.252.0**

Address no longer specifies network ID range.
New forwarding trick: Longest Prefix Match

Benefits of CIDR

- **Efficiency:** Can allocate blocks of prefixes on a finer granularity
- **Hierarchy:** Prefixes can be *aggregated* into supernets. (Not always done. Typically not, in fact.)



Forwarding: Longest Prefix Match

- Forwarding tables in IP routers
  - Maps each IP prefix to next-hop link(s)
- *Destination-based* forwarding
  - Each packet has a destination address
  - Router identifies longest-matching prefix

forwarding table

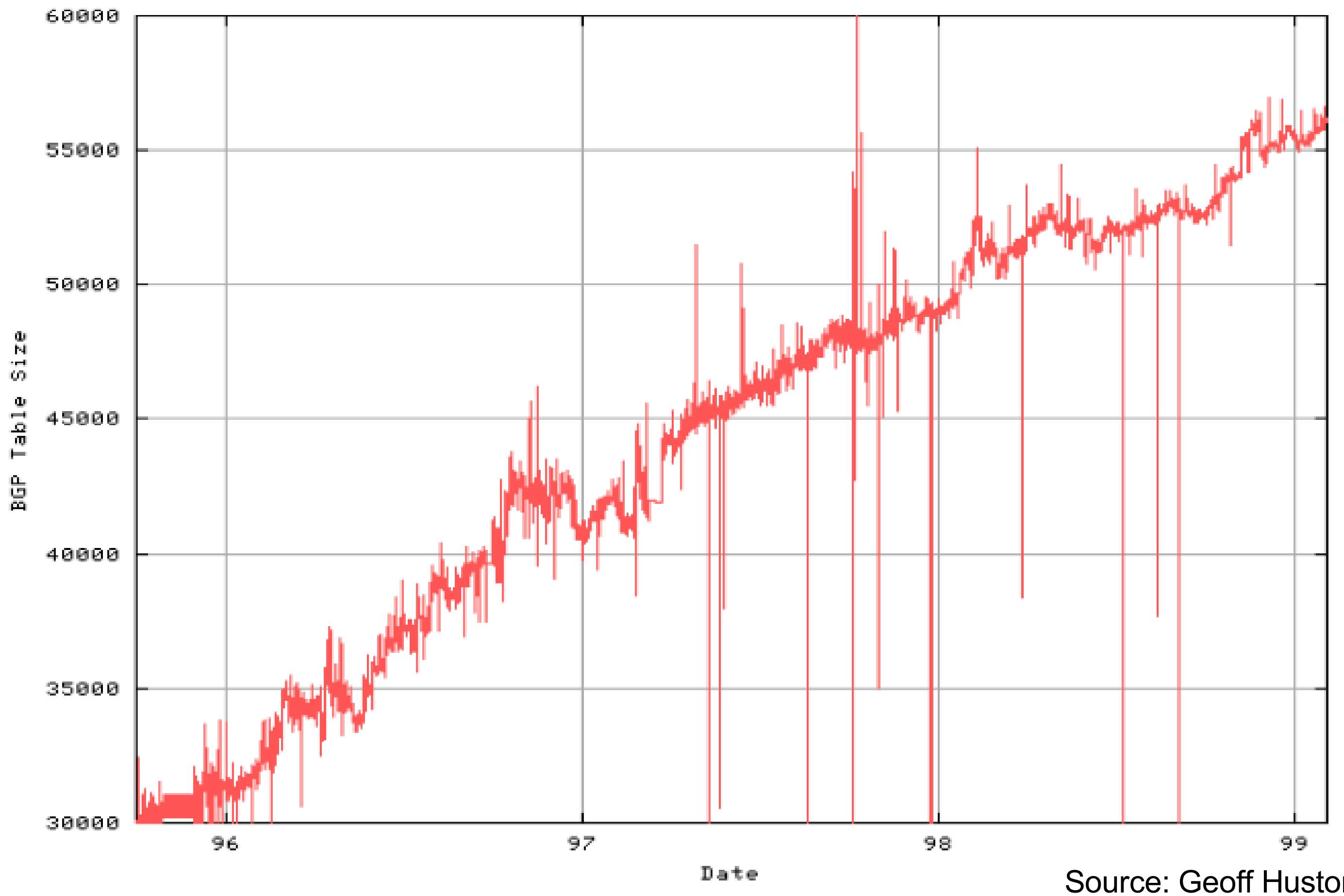
destination address

68.211.6.120 →

...
68.208.0.0/12
68.211.0.0/17
68.211.128.0/19
68.211.160.0/19
68.211.192.0/18
...

More on construction of forwarding tables in next lecture.

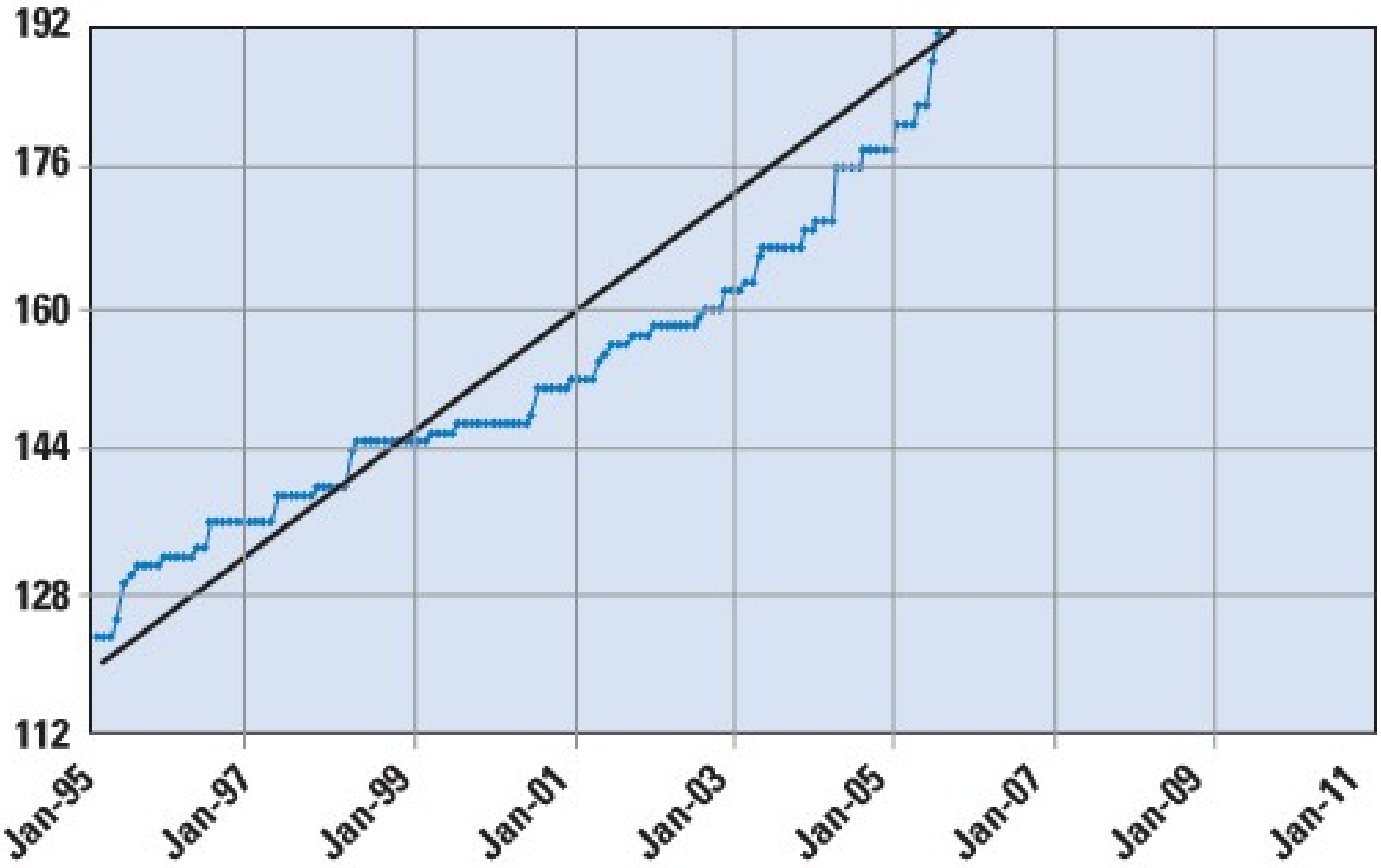
1994-1998: Linear Growth



Source: Geoff Huston

- About 10,000 new entries per year
- In theory, less instability at the edges *(why?)*

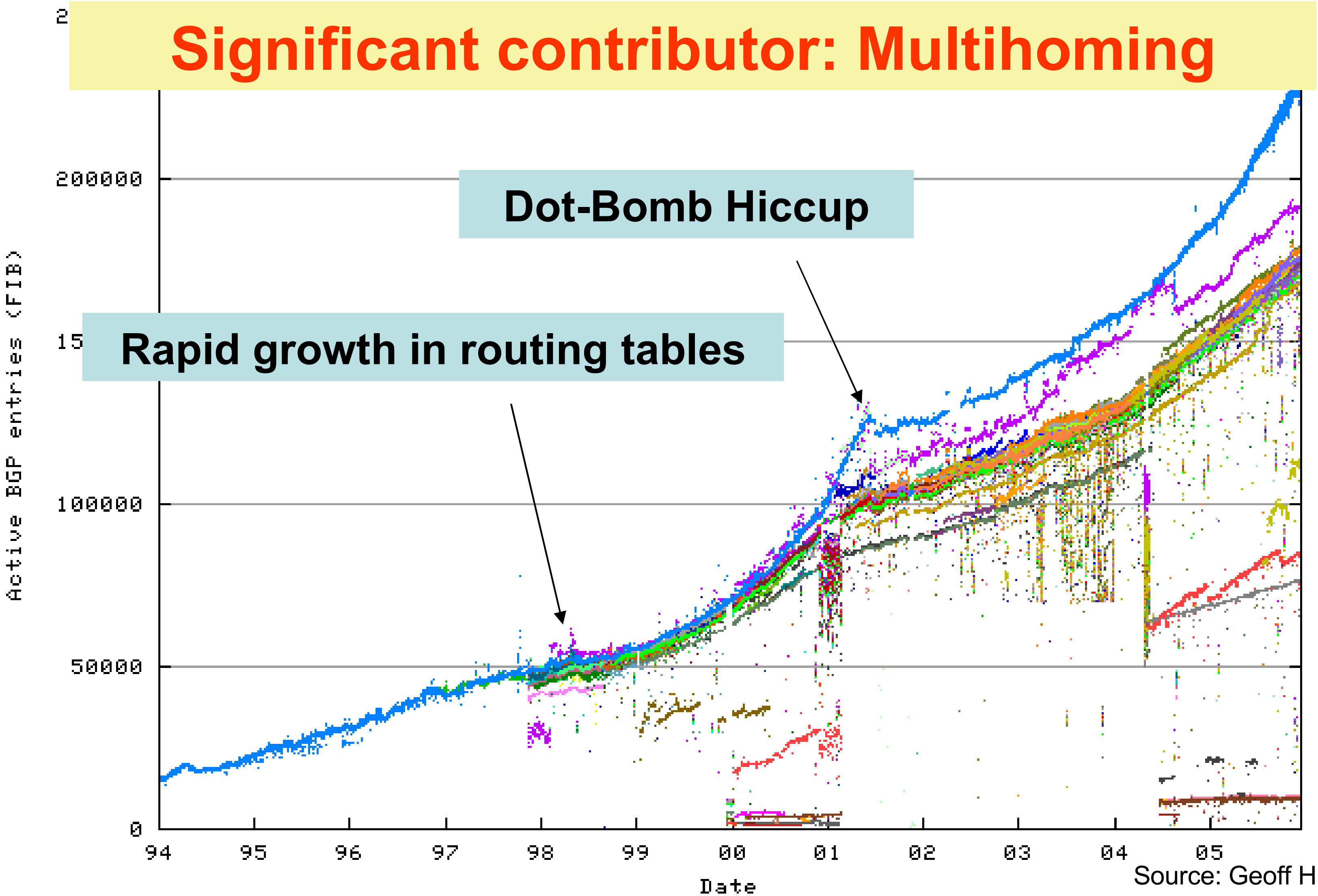
Around 2000: Fast Growth Resumes



T. Hain, "A Pragmatic Report on IPv4 Address Space Consumption", *Cisco IPJ*, September 2005

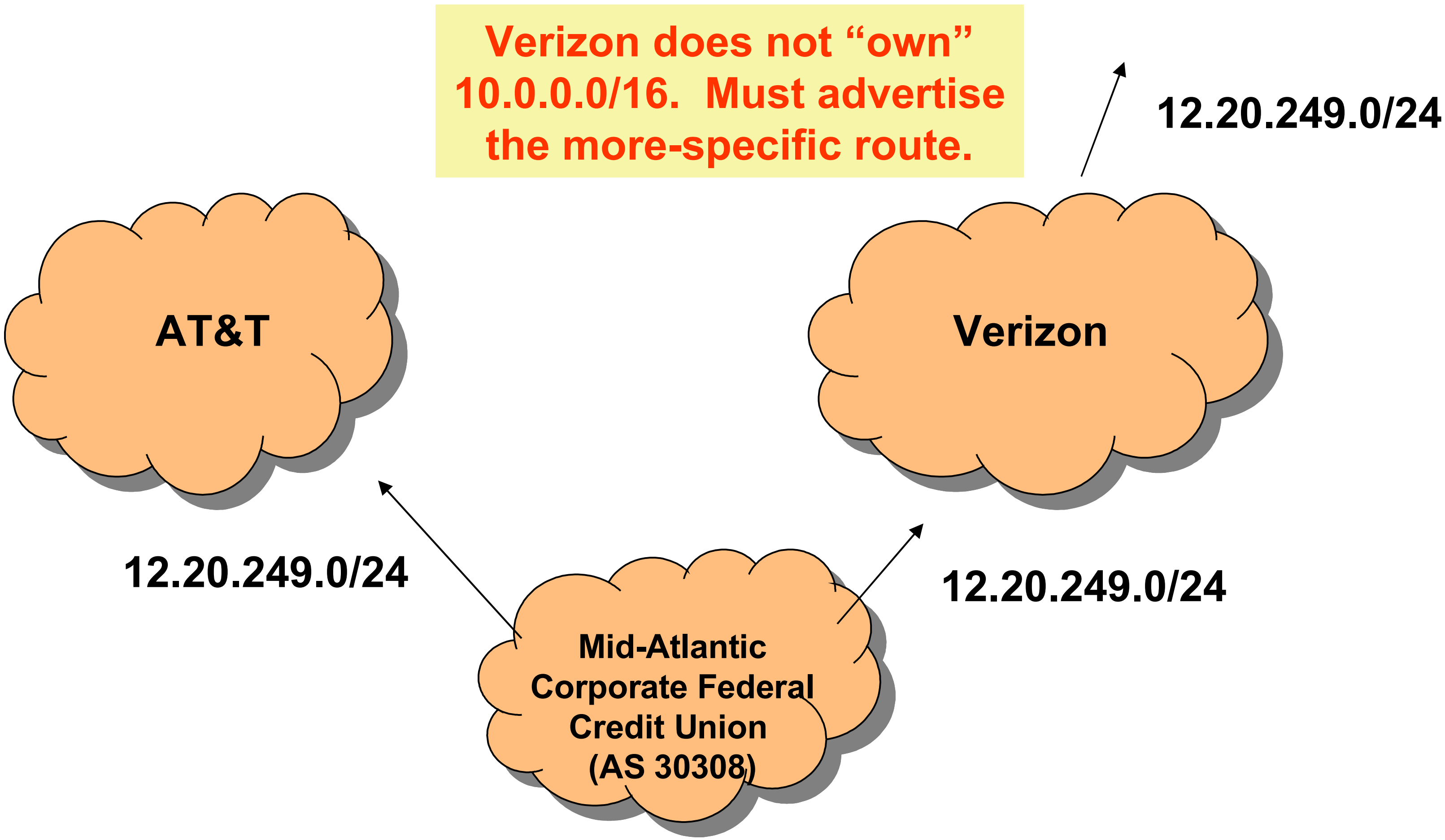
Claim: remaining /8s will be exhausted within the next 5-10 years.

Fast growth resumes



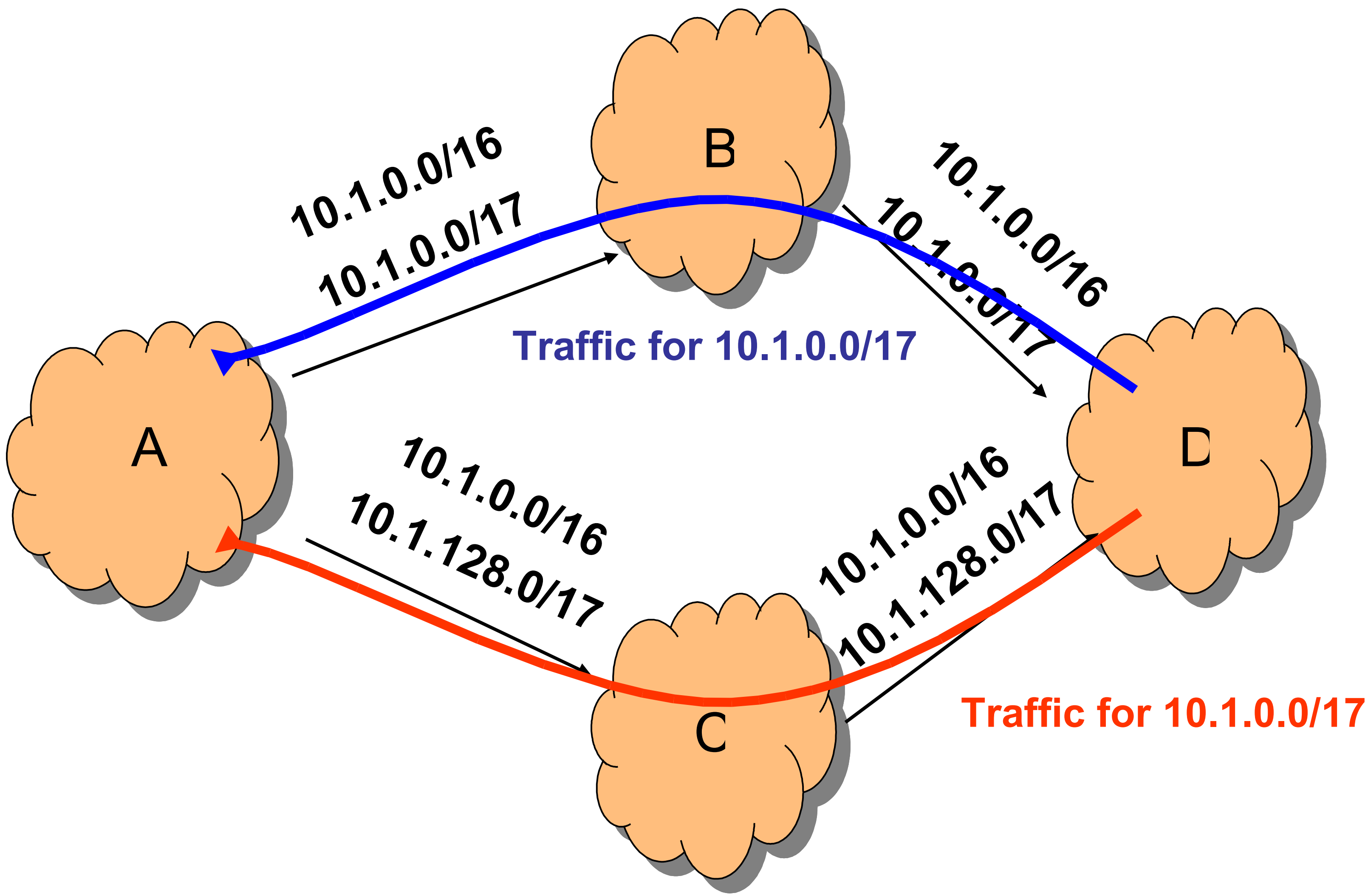
Source: Geoff Huston

Multihoming Can Stymie Aggregation

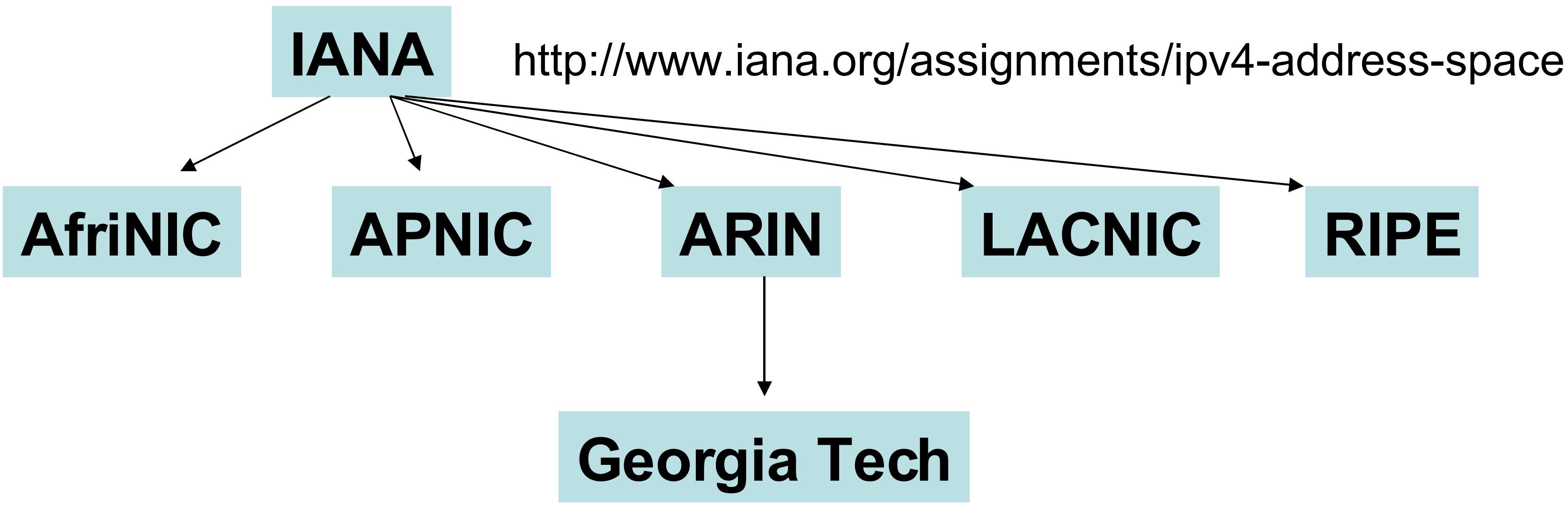


- "Stub AS" gets IP address space from one of its providers
- One (or both) providers cannot aggregate the prefix

Hacky Hack: LPM to Control Traffic

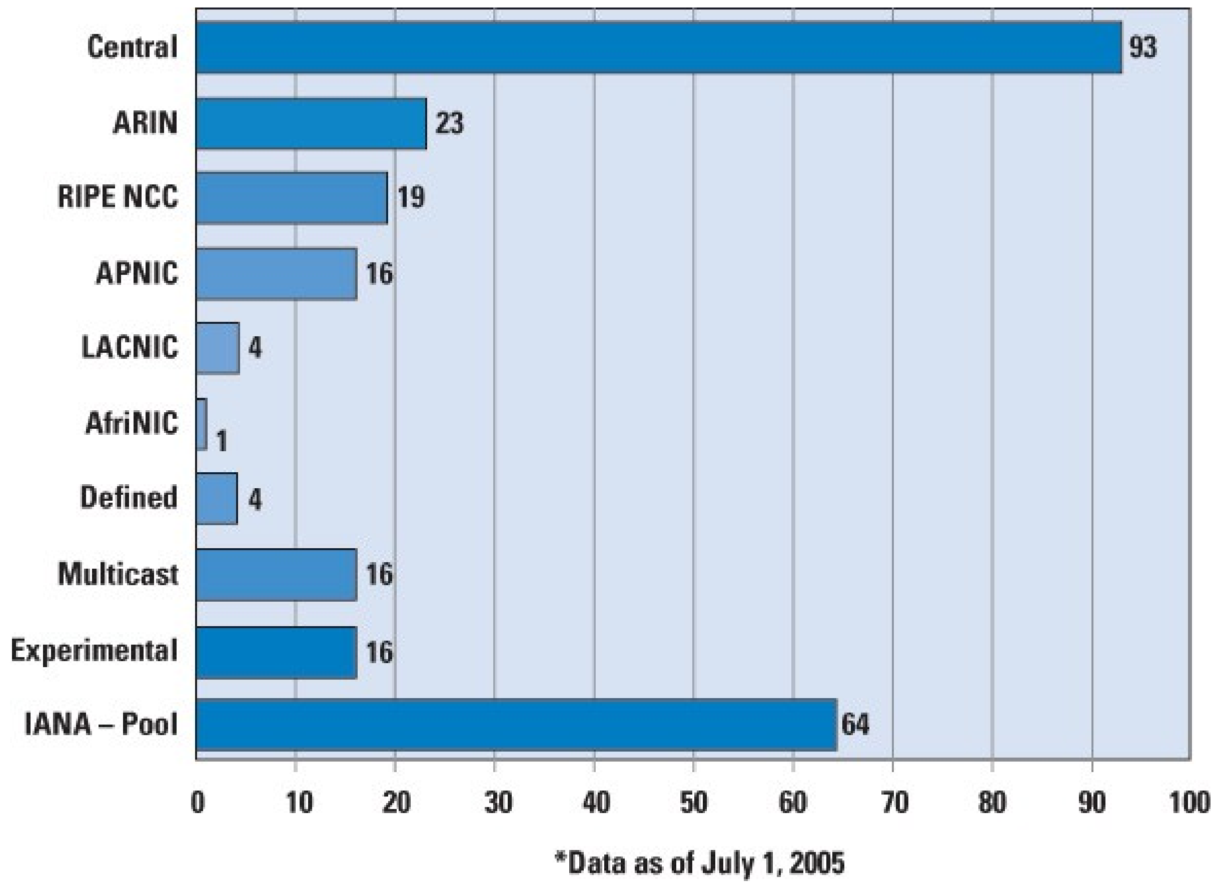


The Address Allocation Process



- Allocation policies of RIRs affect pressure on IPv4 address space

/8 Allocations from IANA



- MIT, Ford, Halliburton, Boeing, Merck
- Reclaiming space is difficult. A /8 is a bargaining chip!

Address Space Ownership

% whois -h whois.arin.net 130.207.7.36
[Querying whois.arin.net]
[whois.arin.net]

OrgName: Georgia Institute of Technology
OrgID: GIT
Address: 258 Fourth St NW
Address: Rich Building
City: Atlanta
StateProv: GA
PostalCode: 30332
Country: US

NetRange: 130.207.0.0 - 130.207.255.255
CIDR: 130.207.0.0/16
NetName: GIT
NetHandle: NET-130-207-0-0-1
Parent: NET-130-0-0-0-0
NetType: Direct Assignment
NameServer: TROLL-GW.GATECH.EDU
NameServer: GATECH.EDU
Comment:
RegDate: 1988-10-10
Updated: 2000-02-01

RTechHandle: ZG19-ARIN
RTechName: Georgia Institute of TechnologyNetwork Services
RTechPhone: +1-404-894-5508
RTechEmail: hostmaster@gatech.edu

OrgTechHandle: NETWO653-ARIN
OrgTechName: Network Operations
OrgTechPhone: +1-404-894-4669

- **Regional Internet Registries ("RIRs")**
  - **Public record of address allocations**
  - **ISPs should update when delegating address space**
  - **Often out-of-date**

Do Prefixes Reflect Topology?

Date: Sat, 11 May 2002 17:34:39 -0400 (EDT)

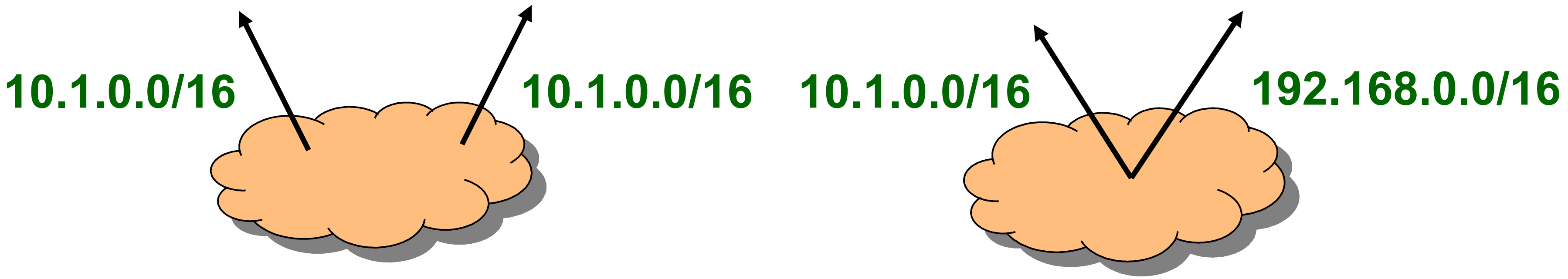
Subject: BGP and aggregation

To: nanog@merit.edu

I have transit in 2 cities...I've been using non-contiguous IPs, so there's been **no opportunity for aggregation**. Having just received my /20 from ARIN, I'm trying to plan my network. **Let's say I split the /20 into 2 /21's, one for each city…**

Missed opportunities for aggregation: non-contiguous prefixes
Multiple geographic locations within the same prefix

Two Problems



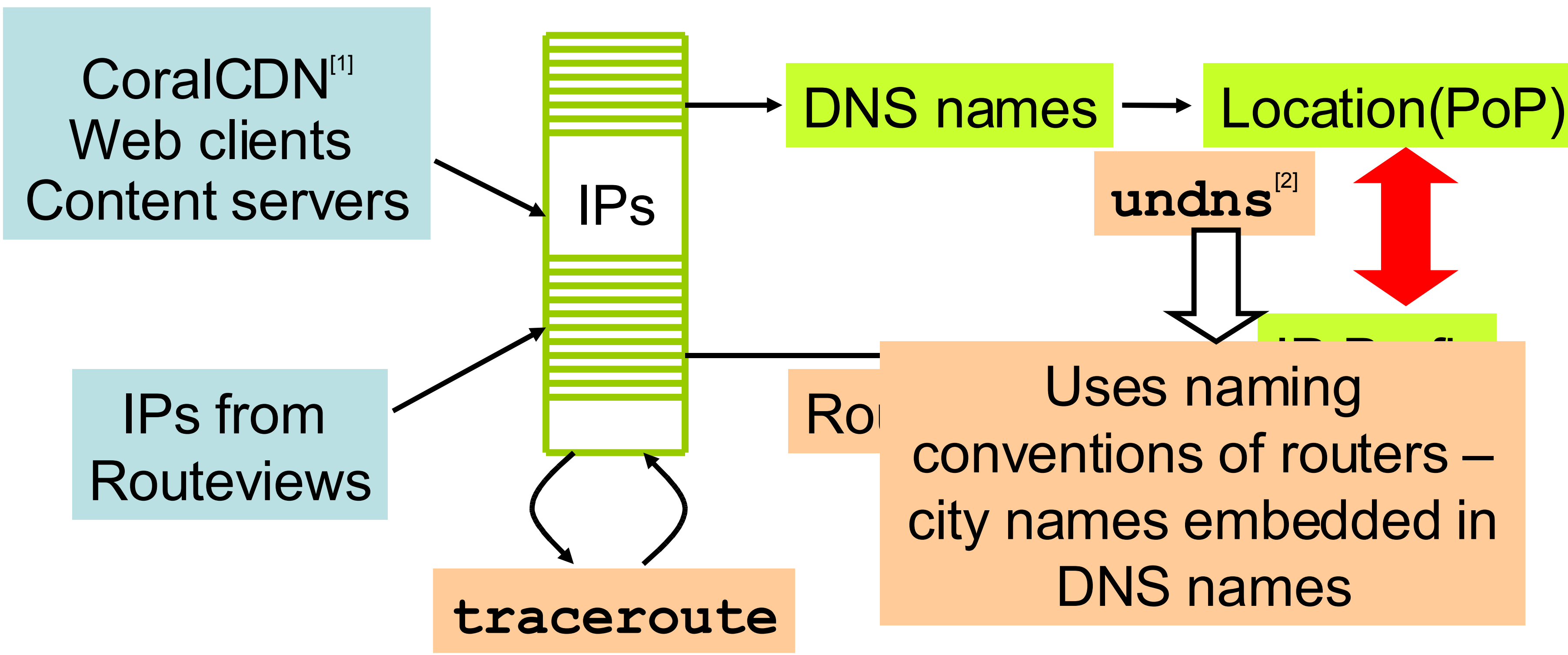
IP space	Geography	Problem
Close/Identical	Far	*Too Coarse-grained*
Far	Close/Identical	*Too Fine-grained*

Case #1 [coarse-grained]: single prefix, multiple locations
contiguous prefixes, multiple locations

Case #2 [fine-grained]: discontiguous prefixes, same location

Method

GOAL: Associate an IP prefix with a set of locations

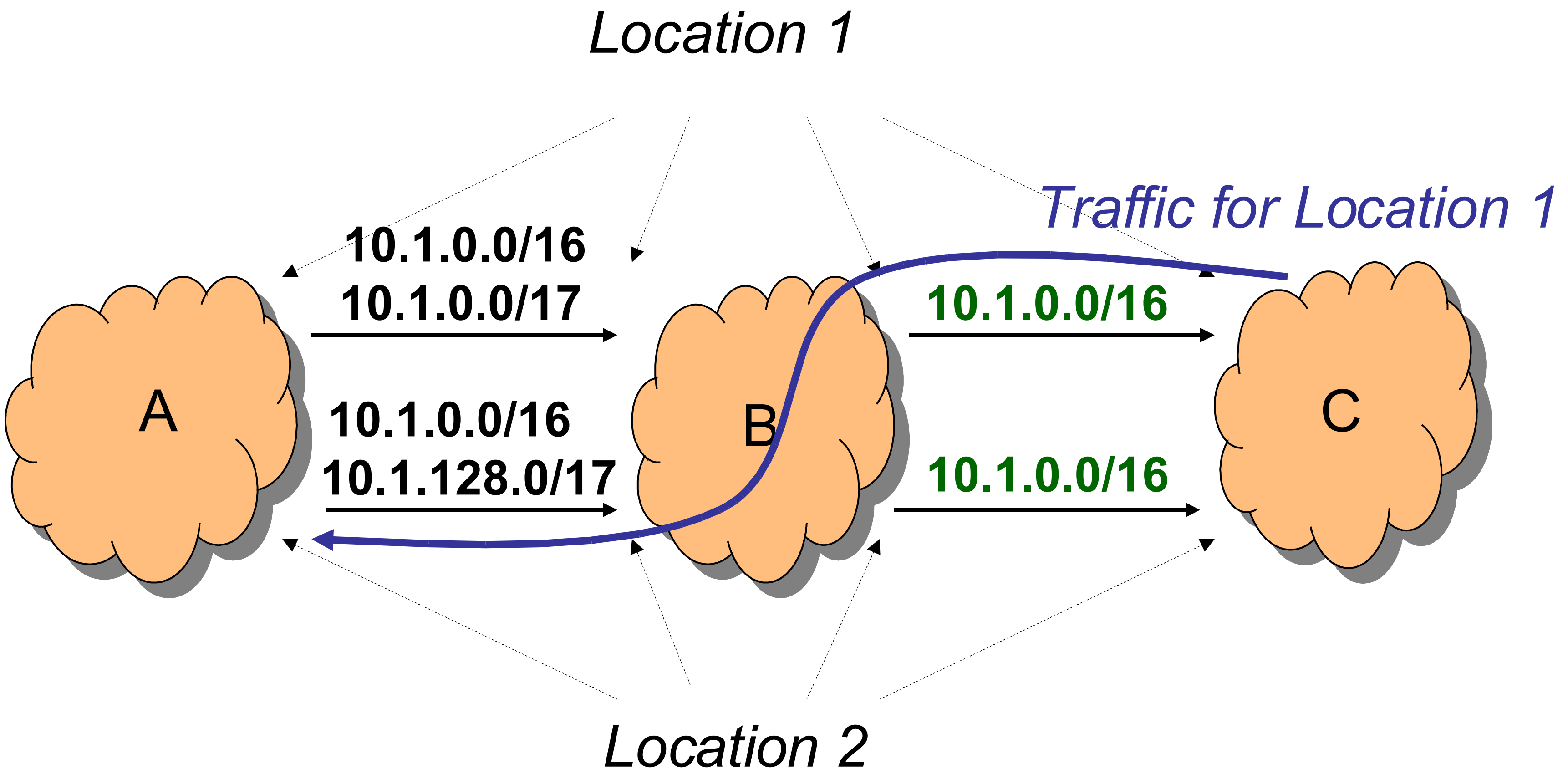


[1] http://www.coralcdn.org

[2] http://www.scriptroute.org

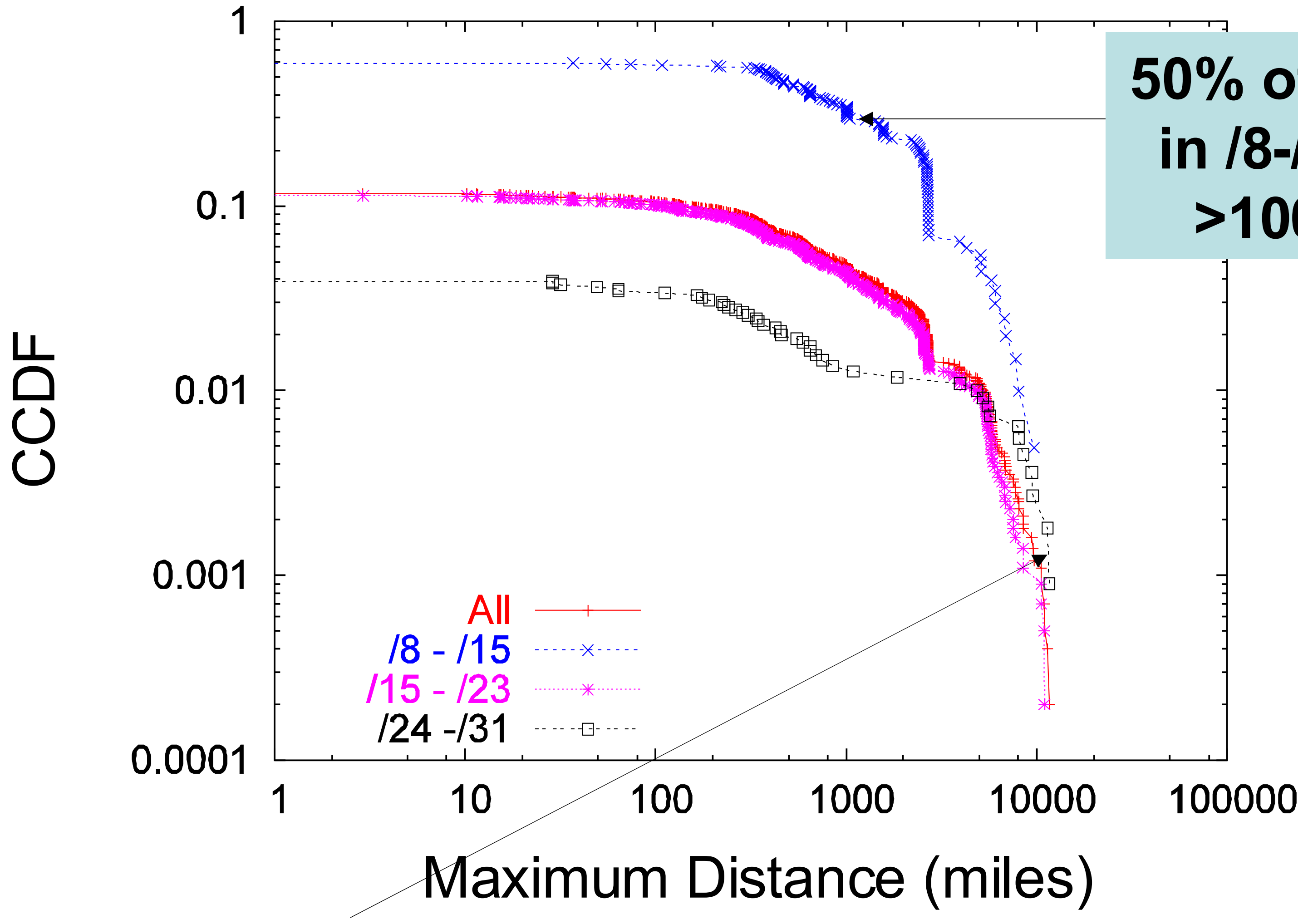
[3] http://www.routeviews.org

Case #1: Coarse-Grained Prefixes



Traffic does not enter AS as intended.
Routing table entries map poorly to reachability.

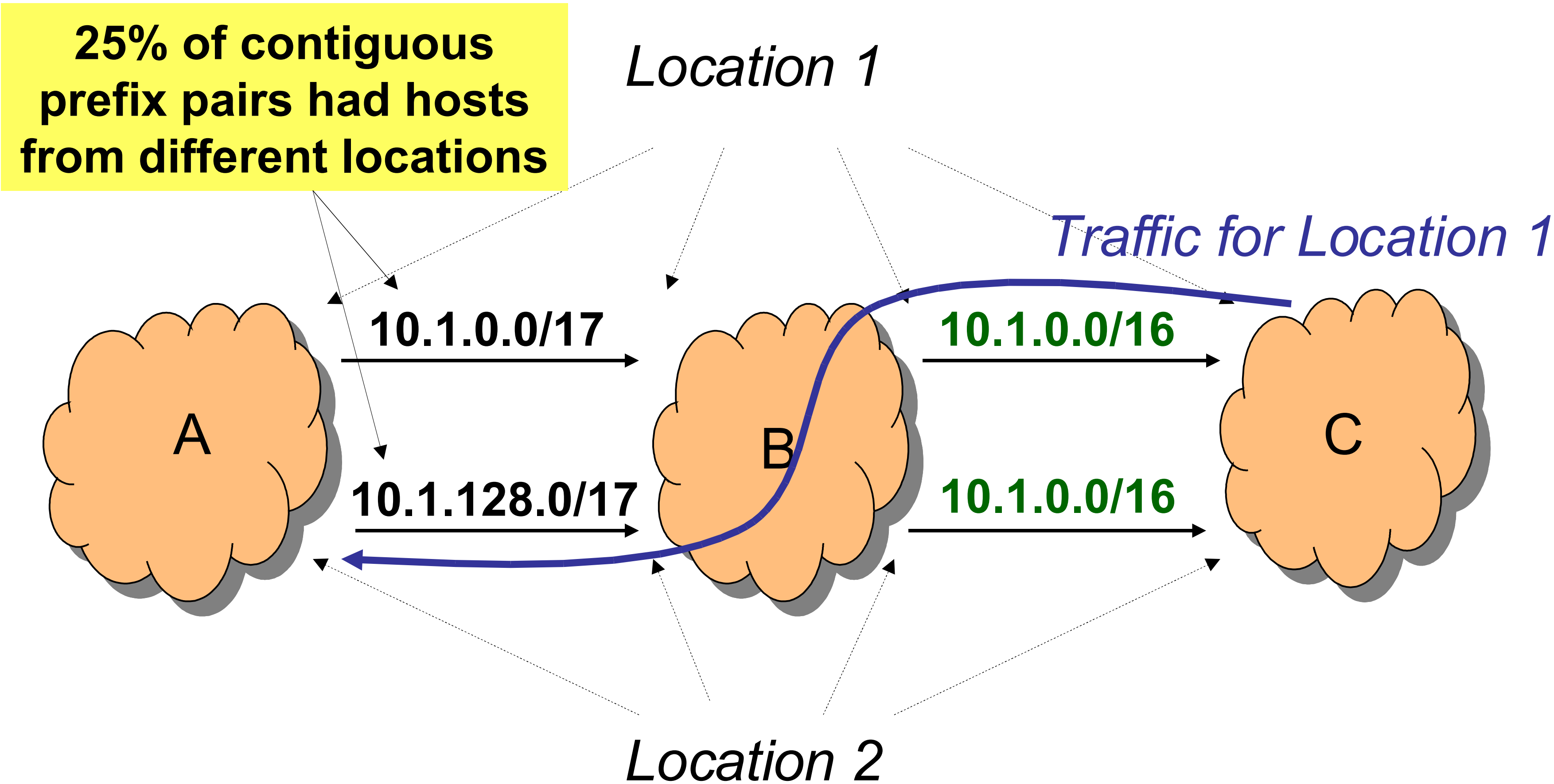
One Prefix May Span Large Distances



50% of prefixes in /8-/15 span >100 miles

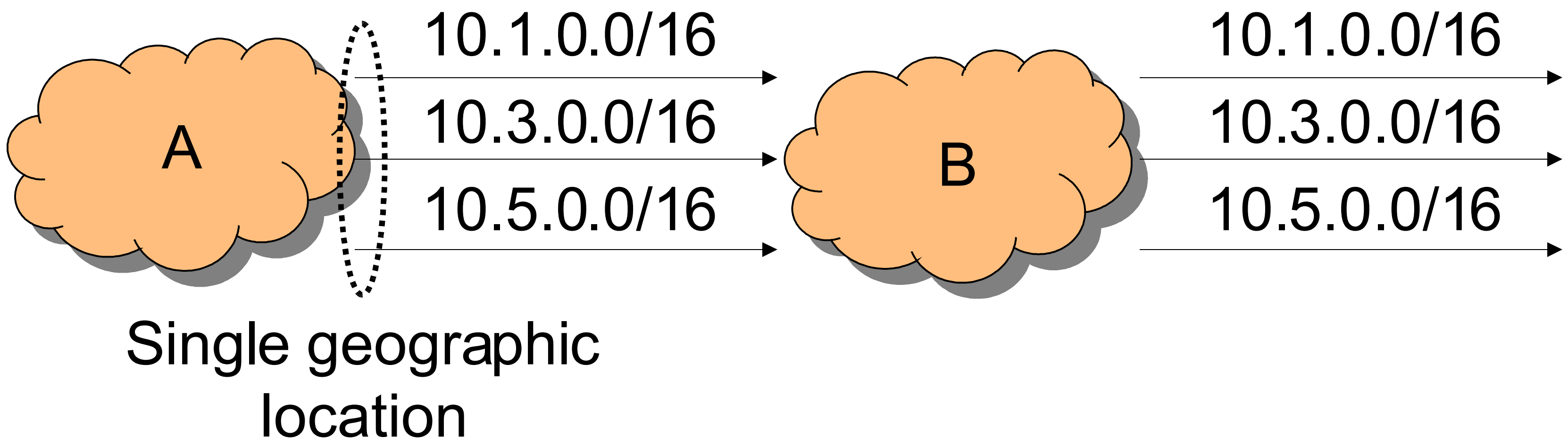
AS 4637: many /24s spanning more than 10,000 miles

Case #1: Coarse-Grained Prefixes



Traffic does not enter AS as intended.
Routing table entries map poorly to reachability.

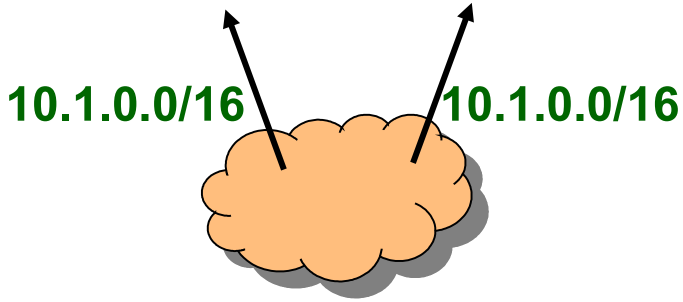
Case #2: Fine-Grained Prefixes



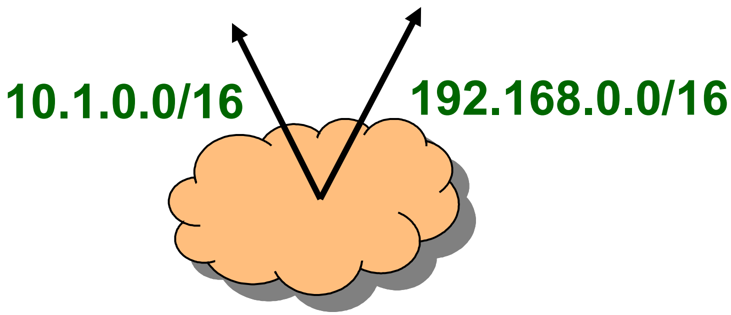
Inflation of routing table size.
Increased routing table churn.

Take-home lessons

- ***Case #1:* Coarse-grained prefixes**
  – Negative effects on traffic control
  – Poor correlation with actual reachability
  – **Finding:** Single prefixes and contiguous prefixes can span very large distances
  – Potential for aggregation overstated



- ***Case #2:* Fine-grained prefixes**
  – Causes many routing table updates
  – Inflates routing table size
  – **Finding:** 70% of discontiguous prefix pairs from common AS and location
  – Changes to routing granularity warranted



IPv6 and Address Space Scarcity

- 128-bit addresses
  - Top 48-bits: Public Routing Topology (PRT)
    - 3 bits for aggregation
    - 13 bits for TLA (like “tier-1 ISPs”)
    - 8 reserved bits
    - 24 bits for NLA
  - 16-bit Site Identifier: aggregation within an AS
  - 64-bit Interface ID: 48-bit Ethernet + 16 more bits

  - Pure provider-based addressing
    - Changing ISPs requires renumbering

Question: How else might you make use of these bits?

IPv6: Claimed Benefits

- Larger address space
- Simplified header
- Deeper hierarchy and policies for network architecture flexibility
- Support for route aggregation
- Easier renumbering and multihoming
- Security (*e.g.,* IPv6 Cryptographic Extensions)

IPv6: Deployment Options

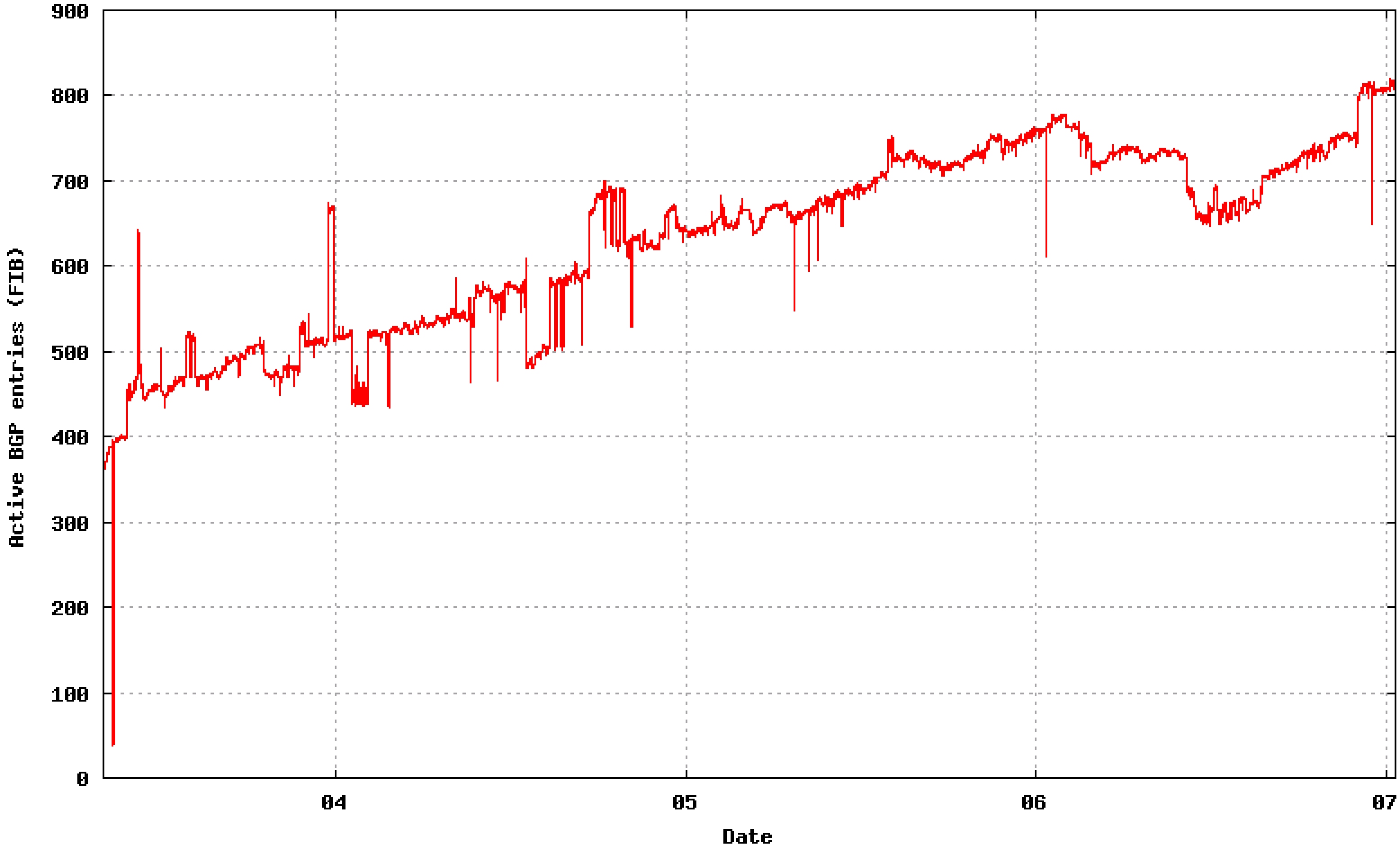
Routing Infrastructure

- IPv4 Tunnels
- Dual-stack
- Dedicated Links
- MPLS

Applications

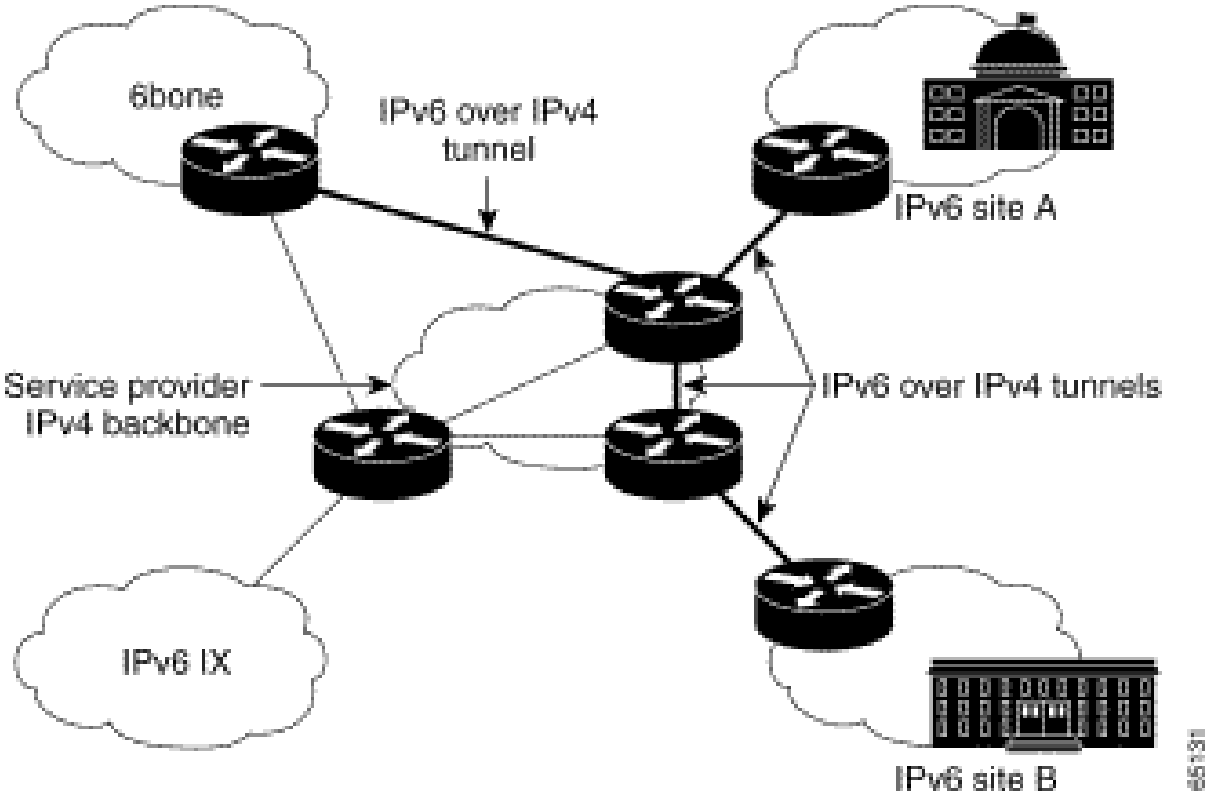
- IPv6-to-IPv4 NAPT
- Dual-stack servers

IPv6 Deployment Status



Big users: Germany (33%), EU (24%), Japan (16%), Australia (16%)

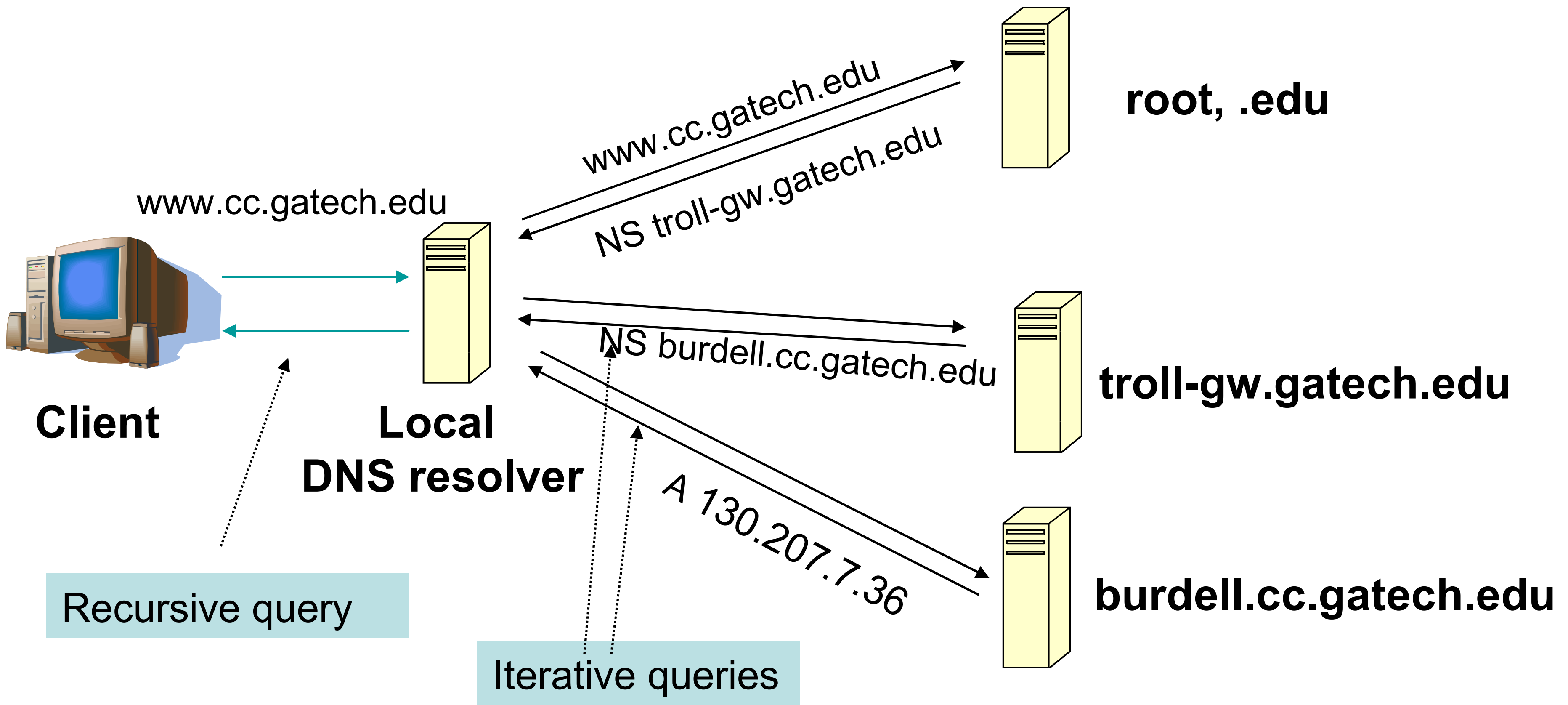
IPv6 over IPv4 Tunnels



One trick for mapping IPv6 addresses: embed the IPv4 address in low bits

http://www.cisco.com/en/US/tech/tk872/technologies_white_paper09186a00800c9907.shtml

DNS: Mapping Names to Addresses



Note the diversity of Georgia Tech's authoritative nameservers

Some Record Types

- A
- NS
- MX
- CNAME
- TXT
- PTR
- AAAA
- SRV

Caching

- Resolvers cache DNS responses
  - Quick response for repeated translations
  - Other queries may reuse some parts of lookup
    - NS records for domains typically cached for longer
  - *Negative responses* also cached
    - Typos, “localhost”, etc.

- Cached data periodically times out
  - Lifetime (TTL) of data controlled by owner of data
  - TTL passed with every record

- What if DNS entries get corrupted?

Root Zone

- Generic Top Level Domains (gTLD)
  - .com, .net, .org,
- Country Code Top Level Domain (ccTLD)
  - .us, .ca, .fi, .uk, etc…
- Root server ({a-m}.root-servers.net) also used to cover gTLD domains
  - Increased load on root servers
  - August 2000: .com, .net, .org moved off root servers onto gTLDs

Some Recent gTLDs

- .info → general info
- .biz → businesses
- .name → individuals
- .aero → air-transport industry
- .coop → business cooperatives
- .pro → accountants, lawyers, physicians
- .museum → museums

Do you trust the TLD operators?

- Wildcard DNS record for all .com and .net domain names not yet registered by others
  - September 15 – October 4, 2003
  - February 2004: Verisign sues ICANN
- Redirection for these domain names to Verisign web portal
- What services might this break?

Protecting the Root Nameservers

Attack On Internet Called Largest Ever

By David McGuire and Brian Krebs
washingtonpost.com Staff Writers
Tuesday, October 22, 2002; 5:40 PM

The heart of the Internet sustained its largest and most sophisticated attack ever, starting late Monday, according to officials at key online backbone organizations.

Around 5:00 p.m. EDT on Monday, a "distributed denial of service" (DDOS) attack struck the 13 "root servers" that provide the primary roadmap for almost all Internet communications. Despite the scale of the attack, which lasted about an hour, Internet users worldwide were largely unaffected, experts said.

Sophistocated?
Why did nobody notice?

gatech.edu. 13759 NS trollgw.gatech.edu.

Defense Mechanisms

- Redundancy: 13 root nameservers
- IP Anycast for root DNS servers {c,f,i,j,k}.root-servers.net
  - RFC 3258
  - Most *physical* nameservers lie outside of the US

Defense: Replication and Caching

Letter	Old name	Operator	Location
A	ns.internic.net	VeriSign	Dulles, Virginia, USA
B	ns1.isi.edu	ISI	Marina Del Rey, California, USA
C	c.psi.net	Cogent Communications	distributed using anycast
D	terp.umd.edu	University of Maryland	College Park, Maryland, USA
E	ns.nasa.gov	NASA	Mountain View, California, USA
F	ns.isc.org	ISC	distributed using anycast
G	ns.nic.ddn.mil	U.S. DoD NIC	Columbus, Ohio, USA
H	aos.arl.army.mil	U.S. Army Research Lab	Aberdeen Proving Ground, Maryland, USA
I	nic.nordu.net	Autonomica	distributed using anycast
J		VeriSign	distributed using anycast
K		RIPE NCC	distributed using anycast
L		ICANN	Los Angeles, California, USA
M		WIDE Project	distributed using anycast

source: wikipedia

DNS Hack #1: Reverse Lookup

- Method
  - Hierarchy based on IP addresses
  - 130.207.7.36
    - Query for PTR record of 36.7.207.130.in-addr.arpa.
- Managing
  - Authority manages IP addresses assigned to it

DNS Hack #2: Load Balance

- Server sends out multiple A records
- Order of these records changes per-client

DNS Hack #3: Blackhole Lists

- *First:* Mail Abuse Prevention System (MAPS)
  - Paul Vixie, 1997
- *Today:* Spamhaus, spamcop, dnsrbl.org, etc.

Different addresses refer to different reasons for blocking

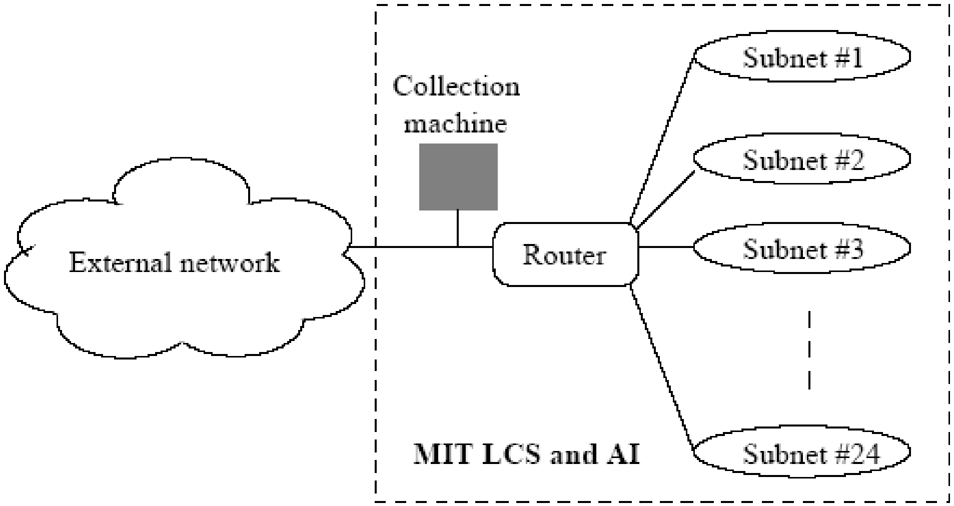
% dig 91.53.195.211.bl.spamcop.net

;; ANSWER SECTION:
91.53.195.211.bl.spamcop.net. 2100 IN A 127.0.0.2

;; ANSWER SECTION:
91.53.195.211.bl.spamcop.net. 1799 IN TXT "Blocked - see http://www.spamcop.net/bl.shtml?211.195.53.91"

Highlights from Today's Paper

- Jung *et al., DNS Performance and the Effectiveness of Caching, ACM IMC,* 2001
- Three different traces: One from MIT, Two from KAIST
  - Joint analysis of DNS and TCP



What types of queries will this miss?

Highlights and Thought Questions

- Load-balancing with A-records does not incur penalty
  - Lower TTLs for A records do not affect performance
  - Wide-area traffic not greatly affected by short TTLs on A records
  - DNS performance relies more on NS-record caching
  - Sharing of caches among clients not effective. Why?
- Referrals responsible for client-perceived latency
- 50% of Lookups not associated with any TCP connection
  - 10% follow from a TCP connection. Why?
- Negative response caching doesn't appear to be effective
  - What effect do DNSBLs have on this?
- Lots of junk DNS traffic
  - 23% of all DNS queries received no answer
  - Half of DNS traffic is for these unanswered queries
  - 15%-27% of traffic at the root is bogus