Network Layer (2)

Networked Systems 3
Lecture 9

Lecture Outline

- **Addressing**
  - Concepts
  - Addressing in the Internet
    - IPv4
    - IPv6
    - The shortage of IPv4 addresses

Addressing

- How to name hosts in a network?

  - Is the address an identity or a location?

    - Does it name the host, or the location at which it attaches to the network

  - How should addresses be allocated?

    - Hierarchical or flat?

  - What is the address format?

    - Human or machine readable?

    - Textual or binary? Structured or unstructured?

    - Fixed or variable length? How large?

Identity and Location

- Addresses can denote host identity

  - Give hosts a consistent address, irrespective of where or when they attach to the network

  - Simple upper-layer protocols

    - Transport layer and applications unaware of multi-homing or host mobility

  - Puts complexity in network layer

    - Network must determine location of host before it can route data

    - Often requires in-network database to map host identity to routable address

    - E.g. non-geographic and mobile phone numbers: 0870 154 154 → 01604 230 230

Identity and Location

- Alternatively, an address can indicate the *location* at which a host attaches to the network

    - Address structure matches the network structure

        - Network can directly route data given an address

        - E.g. geographic phone numbers: +44 141 330 4256

    - Simplifies network layer, by pushing complexity to the higher layers

        - Multi-homing and mobility must be handled by transport layer or applications

        - E.g. transport layer connections break when host moves

Address Allocation

- Are addresses allocated hierarchically?

  - Allows routing on aggregate addresses

    - E.g. phone call to +1 703 243 9422

      Route to US without looking at rest of number

  - Forces address structure to match network topology

  - Requires rigid control of allocations

- Or is there a flat namespace?

  - Flexible allocations, no aggregation → not scalable

Address Formats

- Textual or binary? Fixed or variable length?

  - Fixed length binary easier (faster) for machines to process

  - Variable length textual easier for humans to read

  - Which are you optimising for?

IP Addresses

- IP addresses have the following characteristics:

  - They specify location of a network interface

  - They are allocated hierarchically

  - They are fixed length binary values

    - IPv4: 32 bits

    - IPv6: 128 bits

- Domain names are a separate *application level* namespace

IP Addresses

- Both IPv4 and IPv6 addresses encode location

  - Addresses are split into a *network part* and a *host part*

    - A *netmask* describes the number of bits in the network part

    - The network itself has the address with the host part equal to zero

    - The broadcast address for a network has all bits of host part equal to one(allows messages to be sent to all hosts on a network)

  - A host with several network interfaces will have one IP addresses per interface

    - E.g. a laptop with Ethernet and Wi-Fi interfaces will have two IP addresses

IPv4 Addresses

32 bit binary addresses

IP address: 130.209.247.112 = 10000010 11010001 11110111 01110000

Netmask: 255.255.240.0 = 11111111 11111111 11110000 00000000

20 bits → network = 130.209.240.0/20

Broadcast address:

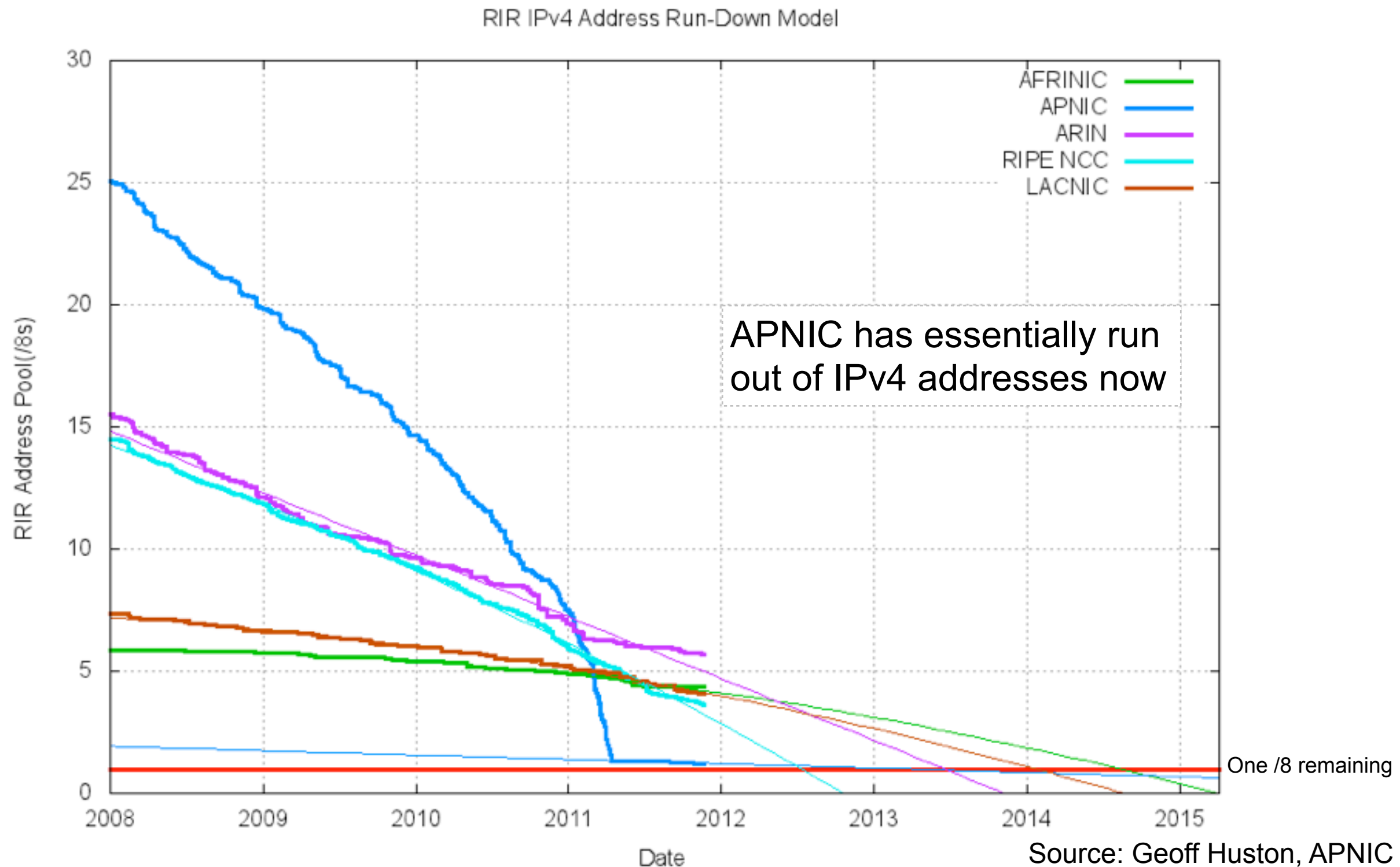130.209.255.255 = 10000010 11010001 11111111 11111111

Aside: Classes of IP address

- IP addresses used to be allocated so the netmask was a multiple of 8 bits

    - Class A → a /8 network (~16 million addresses)
    - Class B → a /16 network (65536 addresses)
    - Class C → a /24 network (256 addresses)

  - Inflexible, and wasted addresses

Old terminology still used sometimes...

- Arbitrary length netmask allowed since 1993:
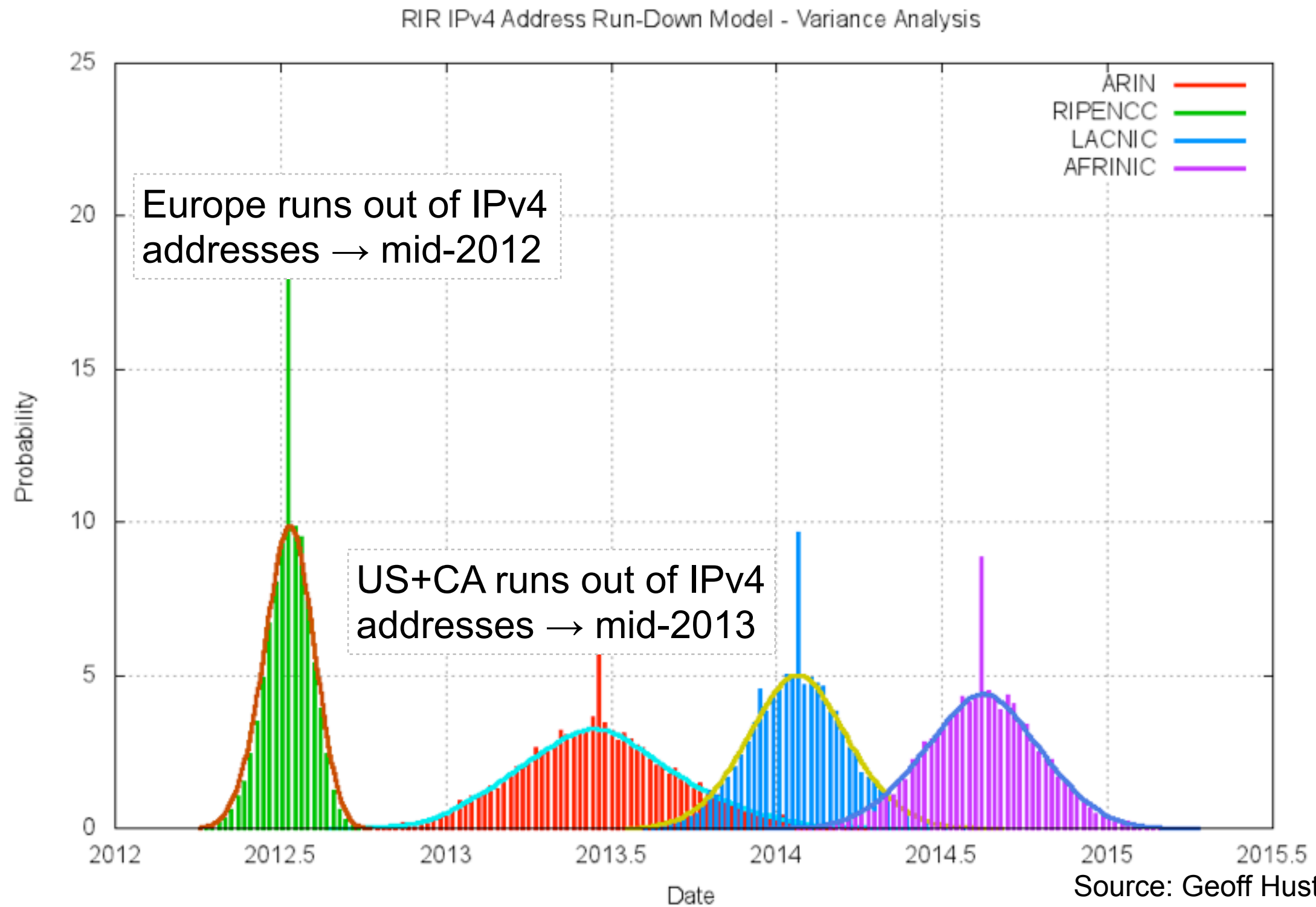
    - The DCS network is a /20

IP Address Management

- IPv4 has 2^{32} = 4,294,967,296 addresses

  - IANA administers the pool of unallocated addresses

    - Historically would assign addresses directly to ISPs, large enterprises, etc.

    - Now, addresses assigned to regional Internet registries (RIRs) as needed:

      - AfriNIC (Africa), APNIC (Asia-Pacific), ARIN (North America), LACNIC (Latin America and Caribbean), and RIPE (Europe, Middle East, Central Asia)
      - Allocations made one /8 (224 = 16,777,216 addresses) at a time

  - RIRs allocate addresses to ISPs and large enterprises within their region; ISPs allocate to their customers

- IANA has allocated all available addresses to RIRs

  - Last allocation made on 3 February 2011

RIR Exhaustion Dates (1)

RIR IPv4 Address Run-Down Model

Legend: AFRINIC, APNIC, ARIN, RIPE NCC, LACNIC

APNIC has essentially run out of IPv4 addresses now

One /8 remaining

Source: Geoff Huston, APNIC
http://ipv4.potaroo.net/

RIR Exhaustion Dates (2)

RIR IPv4 Address Run-Down Model - Variance Analysis

Europe runs out of IPv4 addresses → mid-2012

US+CA runs out of IPv4 addresses → mid-2013

Legend:
ARIN
RIPENCC
LACNIC
AFRINIC

Source: Geoff Huston, APNIC
http://ipv4.potaroo.net/

What Happens Next?

- IPv6 is widely deployed
- Widespread use of NAT
- Or…?

IPv6

- IPv6 provides 128 bit addresses – if deployed it will solve address shortage for a *long* time

  - 2^{128} = 340,282,366,920,938,463,463,374,607,431,768,211,456 addresses

  - i.e., 665,570,793,348,866,943,898,599 addresses for every square metre of the Earth's surface
  - i.e., the same number of IP addresses per person as the number of atoms in a metric ton of carbon

IPv6 Addresses

128 bit binary addresses, written as : separated hexadecimal

> 2001:0db8:85a3:08d3:1319:8a2e:0370:7334

A *single* run of consecutive zeros can be compressed to a ::

> 2001:200::8002:203:47ff:fea5:3085

IPv6 Addresses

Local identifier part of IPv6 address is 64 bits:

2001:0db8:85a3:08d3:1319:8a2e:0370:7334

Local identifier part

Can be derived from Ethernet/Wi-Fi MAC address:

48 bit IEEE MAC: 0014:5104:25ea

Expand to 64 bits: 0014:51ff:fe04:25ea

Invert bit 6:   0214:51ff:fe04:25ea

Or randomly chosen, with bit 6 set
to zero, to give illusion of privacy

IPv6 Addresses

Routers advertise network part, hosts auto-configure address:

2001:0db8:85a3:08d3:1319:8a2e:0370:7334

← Network part →

Network part is split into a *global routing prefix* (a.k.a. "routing goop" of up to 48 bits) and a *subnet identifier*

Formalises the distinction present in IPv4:

130.209.247.112 = 10000010 11010001 11110111 01110000

Global routing prefix   Subnet   Host

Netmask

# IPv6 Deployment Issues

- IPv6 requires changes to *every* single host, router, firewall, and application...

  - Significant deployment challenge!

  - Host changes done: MacOS X, Windows XP & Vista, Linux, FreeBSD, Symbian, etc.

  - Backbone routers generally support IPv6, home routers and firewalls (mostly) not yet

  - Some applications have been updated

Network Address Translation
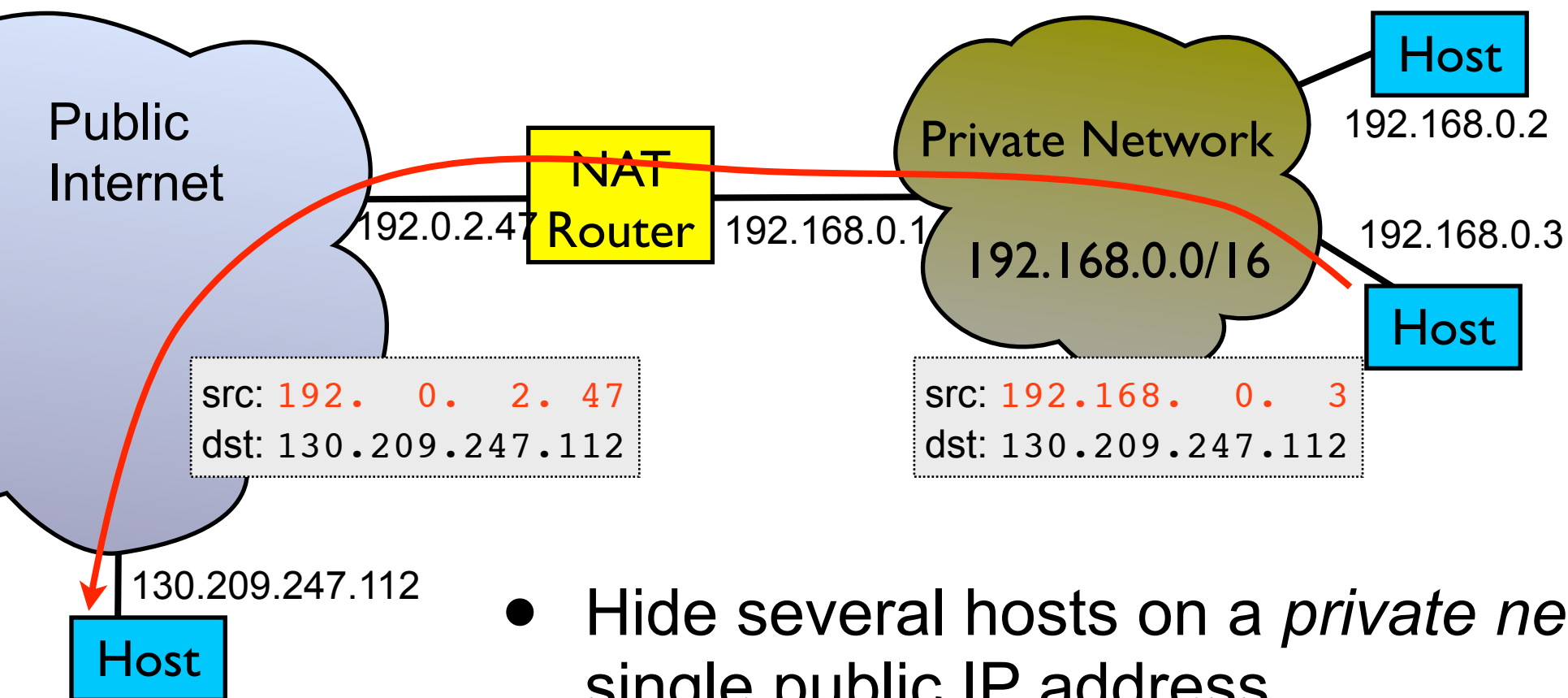
- Hide several hosts on a *private network* behind a single public IP address
  - Private IPv4 addresses are 10.0.0.0/8, 192.168.0.0/16, 176.16.0.0/12
- Rewrite packet headers at network boundary
  - Doesn't require changes to hosts or routers (other than the NAT)
- Tries to give the illusion of more address space

Network Address Translation
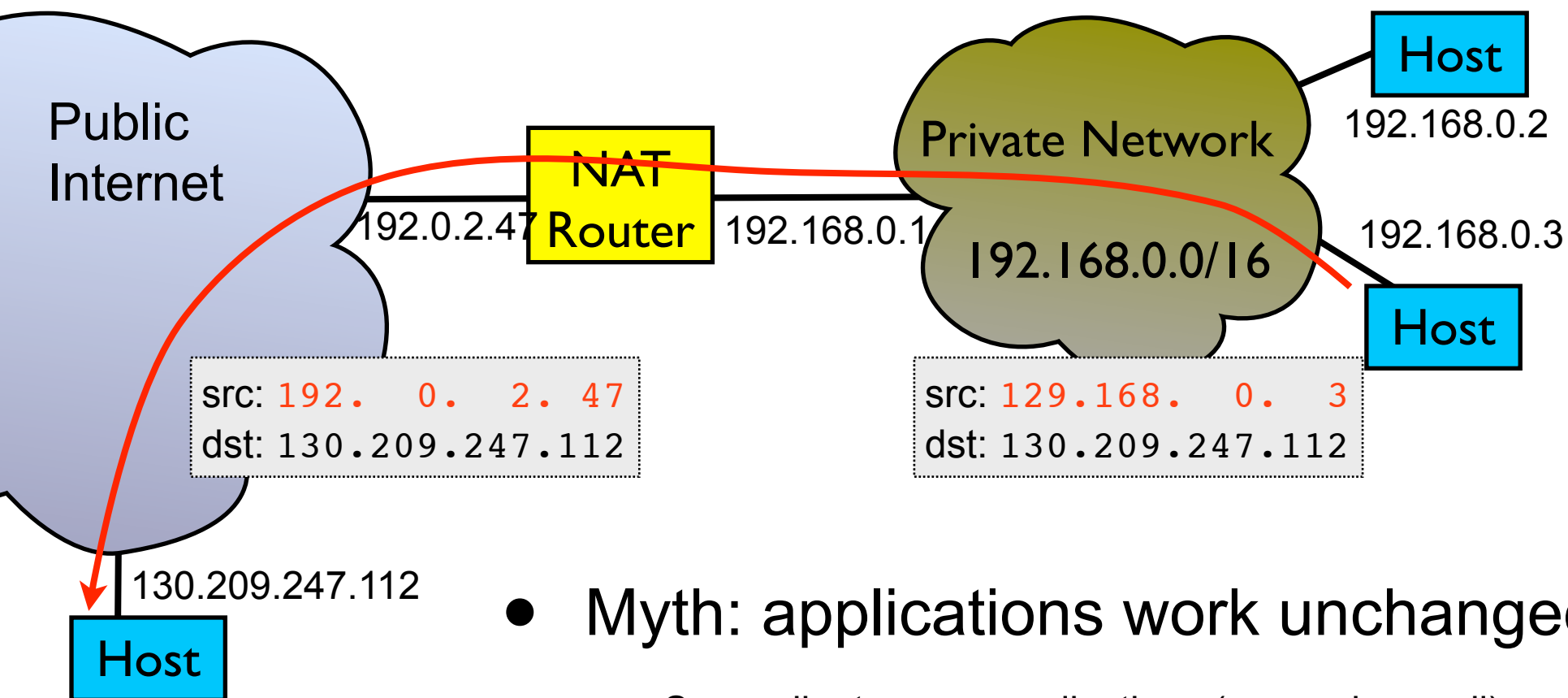
- **Myth: applications work unchanged**
  - *Some* client-server applications (e.g. web, email) work without changes
  - But peer-to-peer applications (e.g. VoIP) need extensive changes before they work through a NAT (~200 pages spec to describe algorithm!)

- **Myth: provides security**
  - Most NATs include a firewall to provide security, the NAT function gives no security benefit

NAT *vs.* IPv6

- NAT widely deployed now for IPv4

    - Initially *appears* simple to end users (no need to change hosts, web browsing & email still work)

    - *Hugely* complex for peer-to-peer applications

    - Very difficult to debug network problems and deploy new classes of application

- The *hope* is that IPv6 will be deployed as IPv4 addresses run out

Questions?