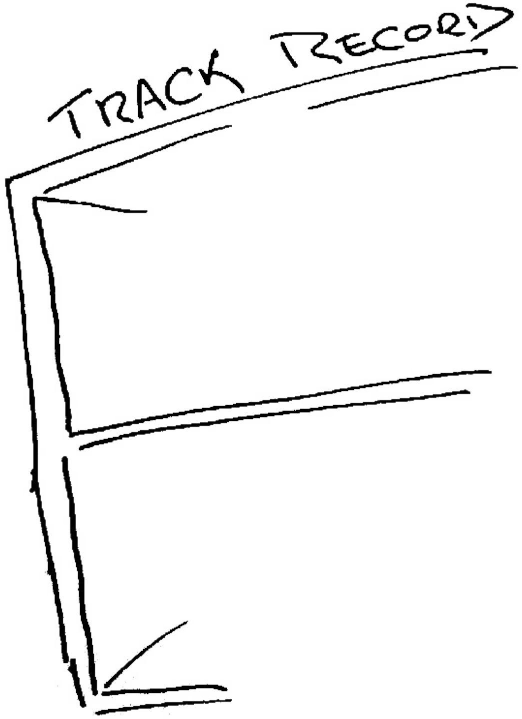
CGNs in IP
What are you going to do about it?

Mark Kosters, ARIN

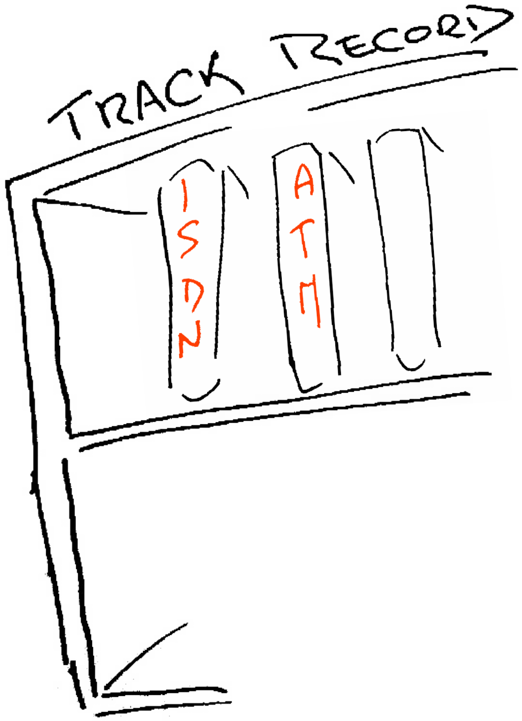
Geoff Huston, APNIC

The mainstream telecommunications industry has a rich history



The mainstream telecommunications industry has a rich history

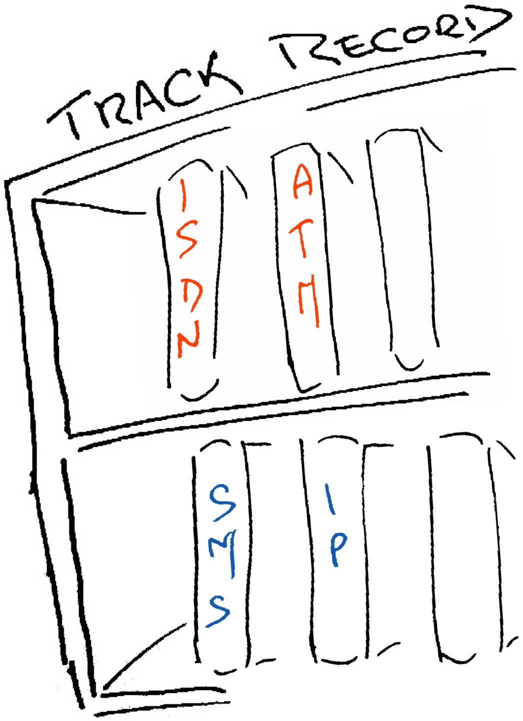
…of making very poor technology choices



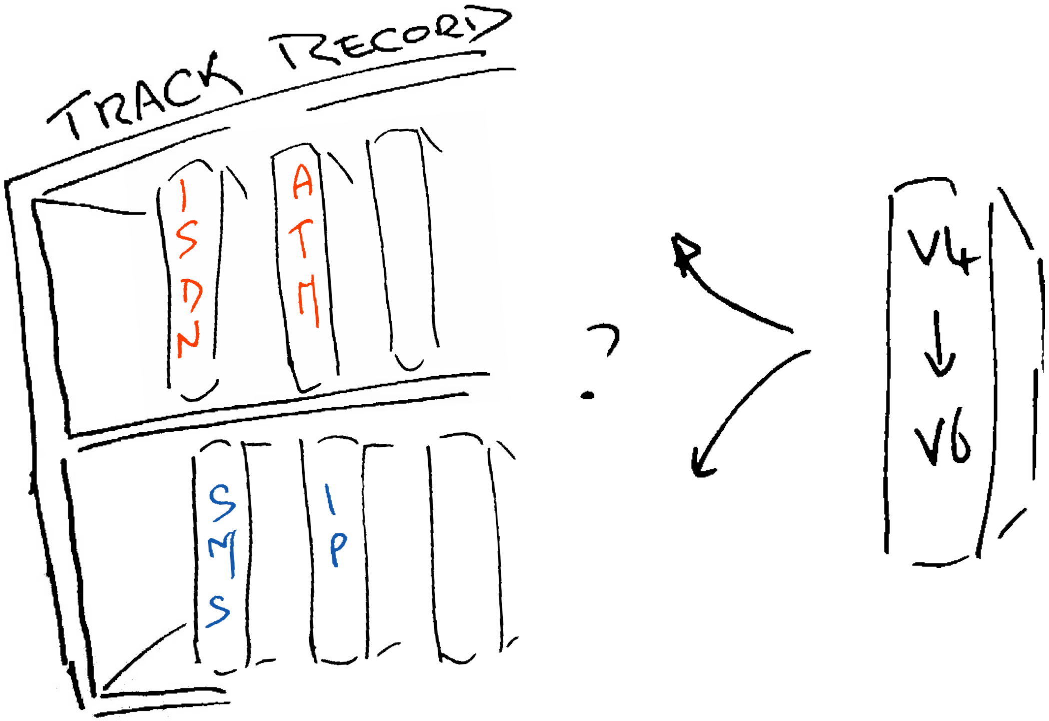
The mainstream telecommunications industry has a rich history

…of making very poor technology guesses

and regularly being taken by surprise!



So, how are we going with the IPv4 to IPv6 transition?

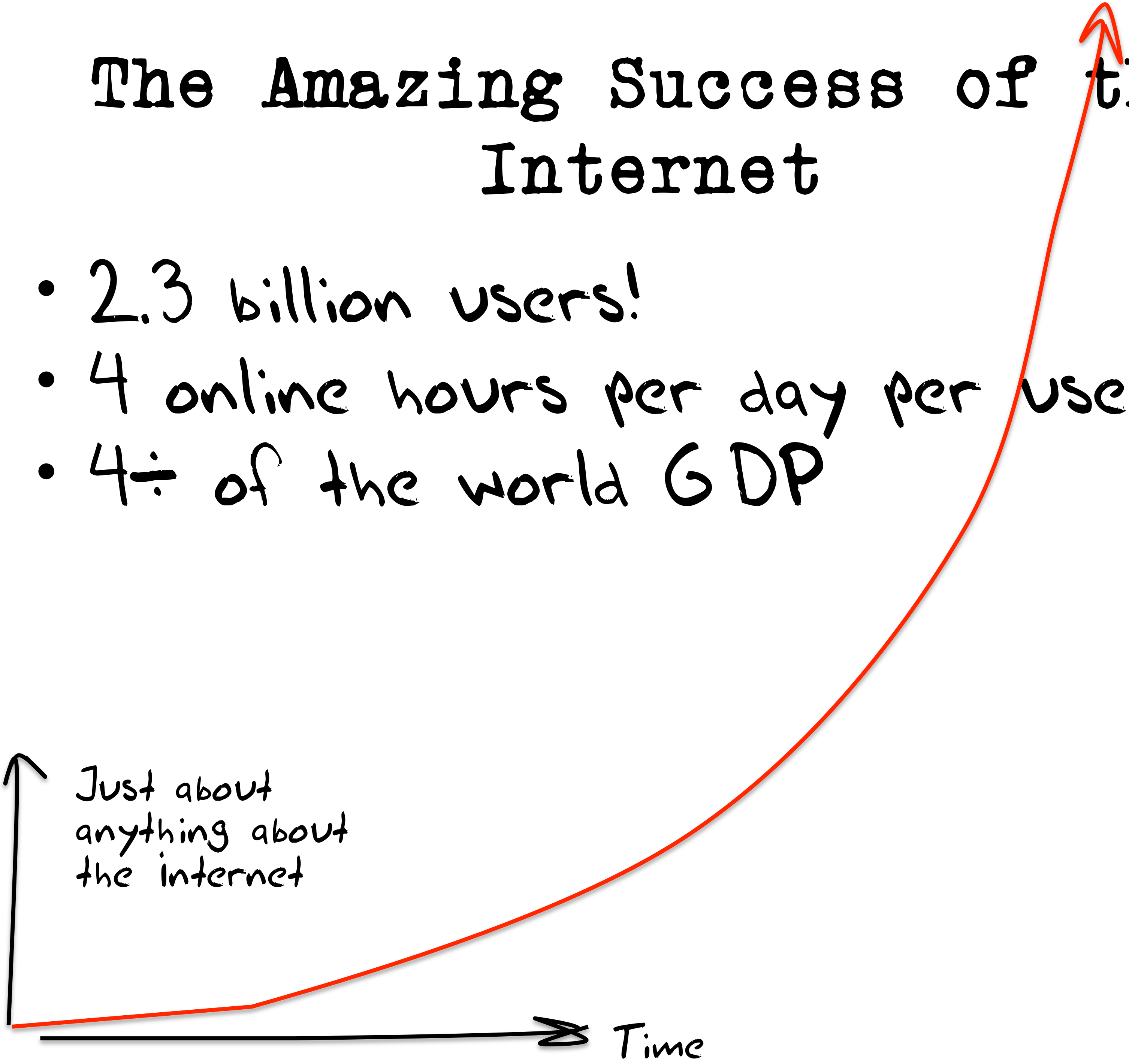


The Amazing Success of the Internet

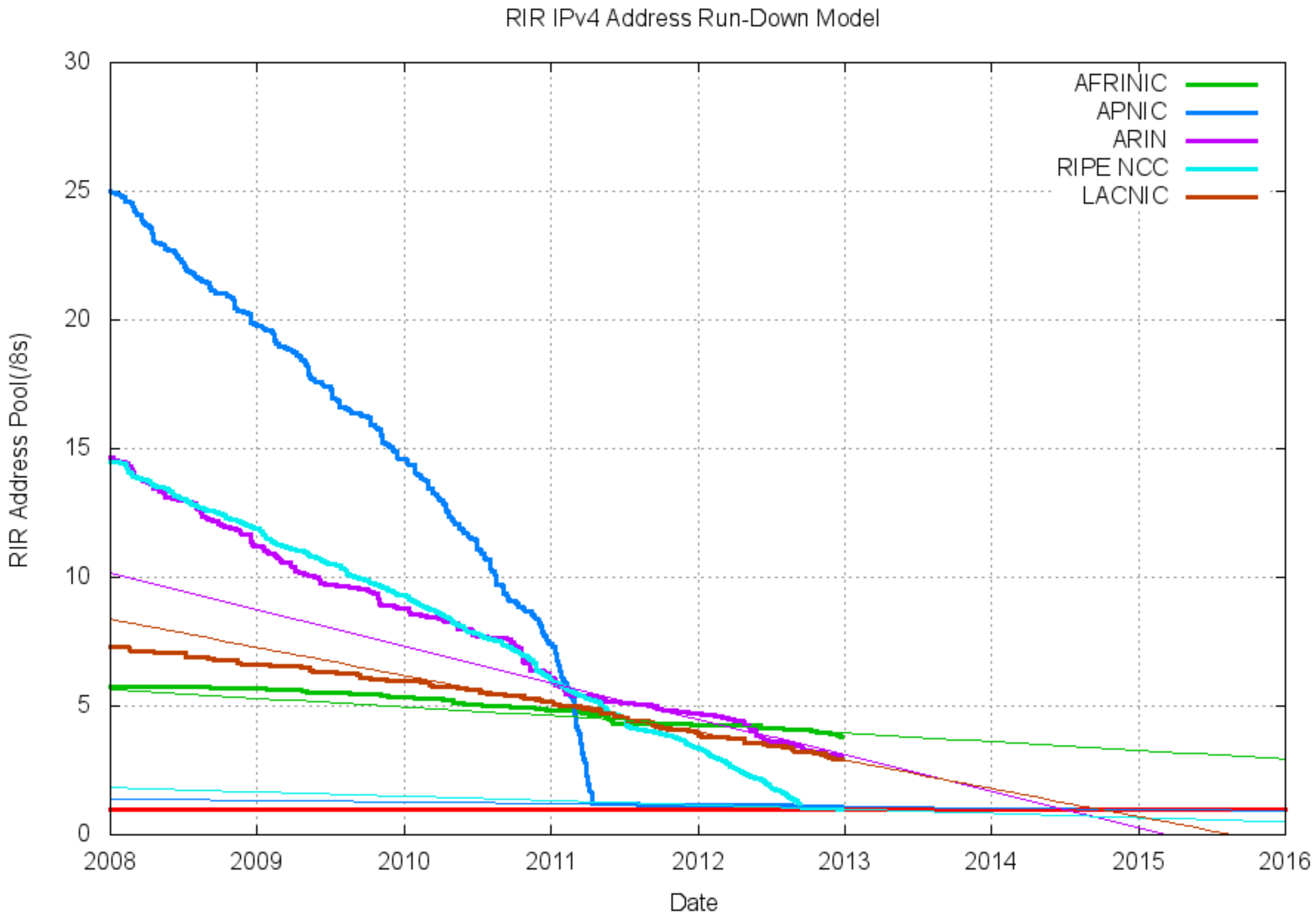
- 2.3 billion users!
- 4 online hours per day per user!
- 4% of the world GDP

The Amazing Success of the Internet

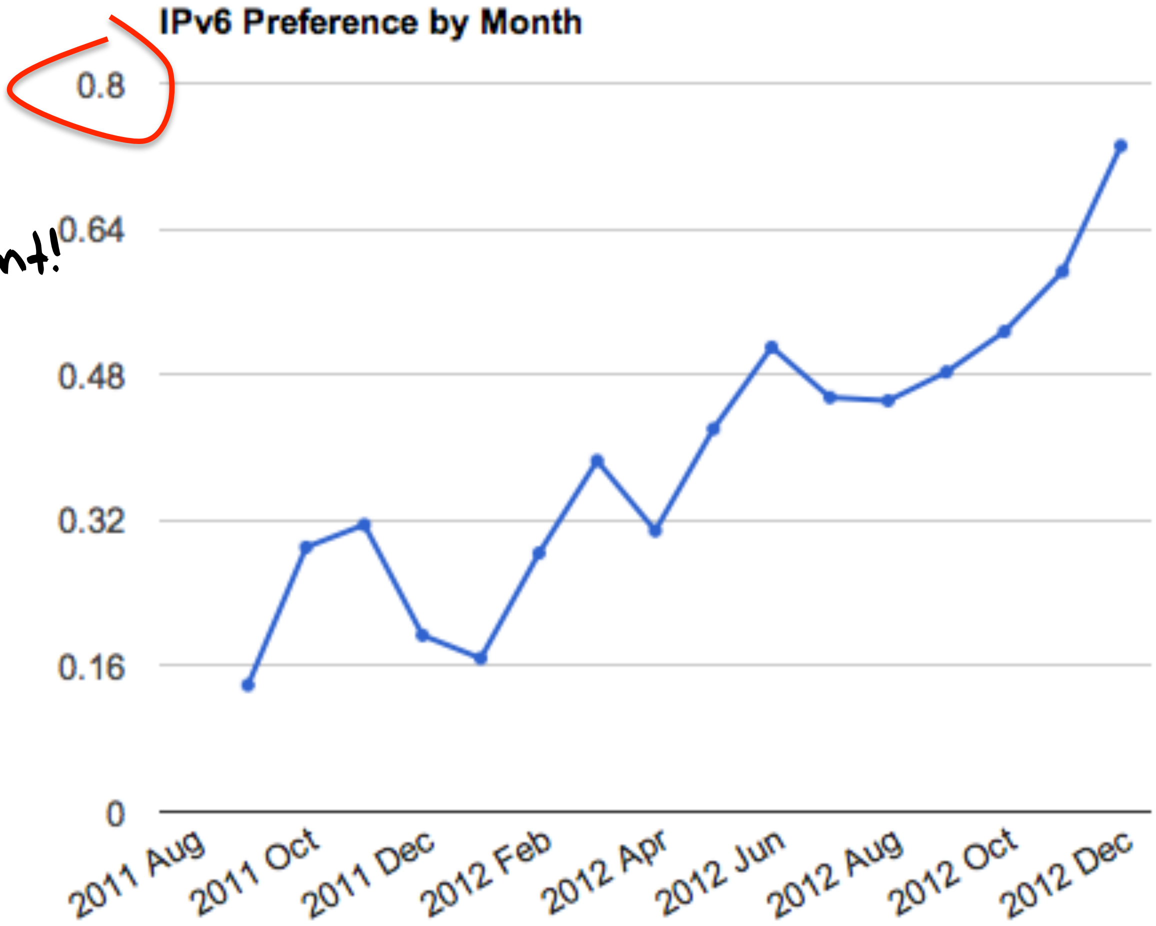
- 2.3 billion users!
- 4 online hours per day per user!
- 4÷ of the world GDP



Success-Disaster

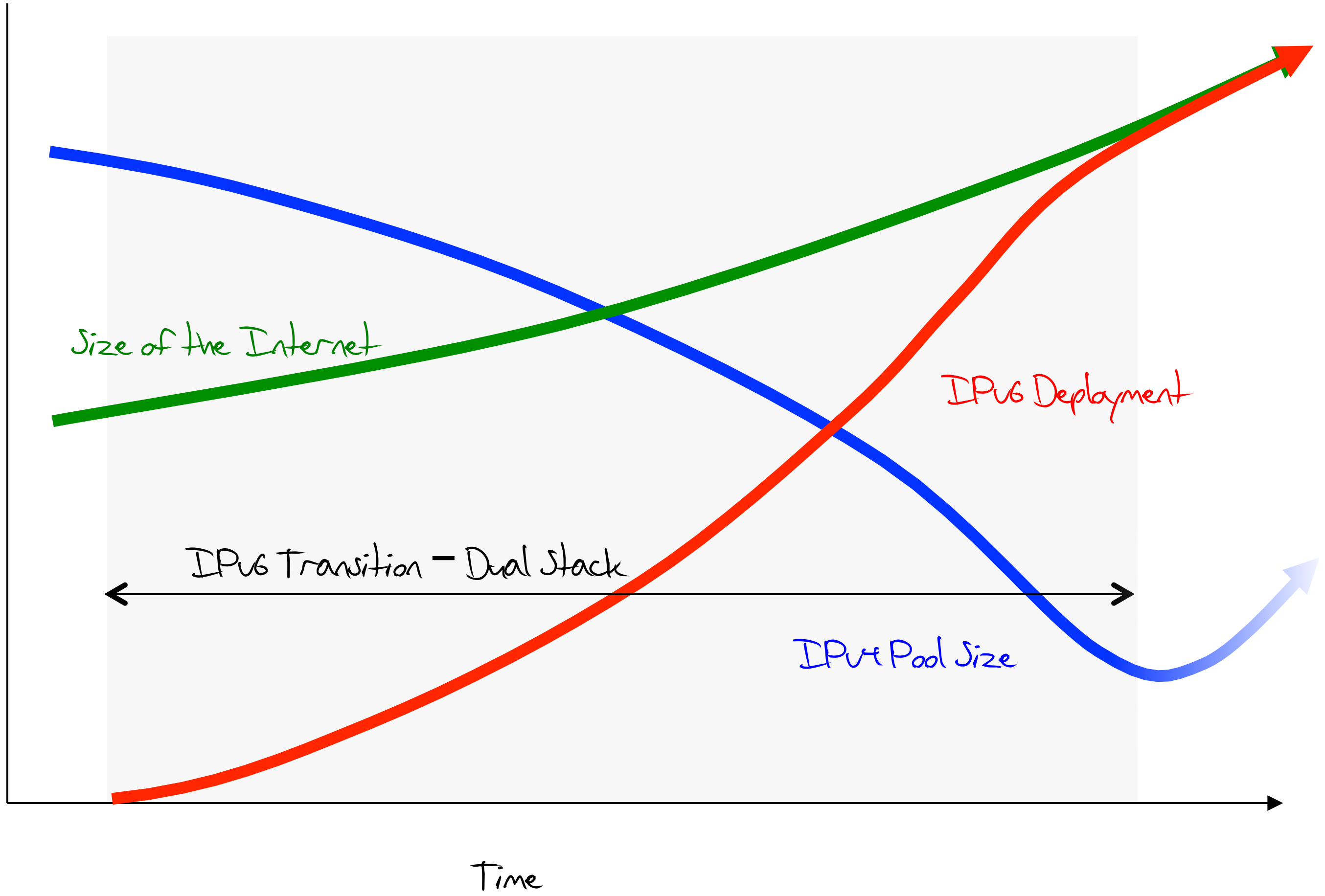


Success-Disaster

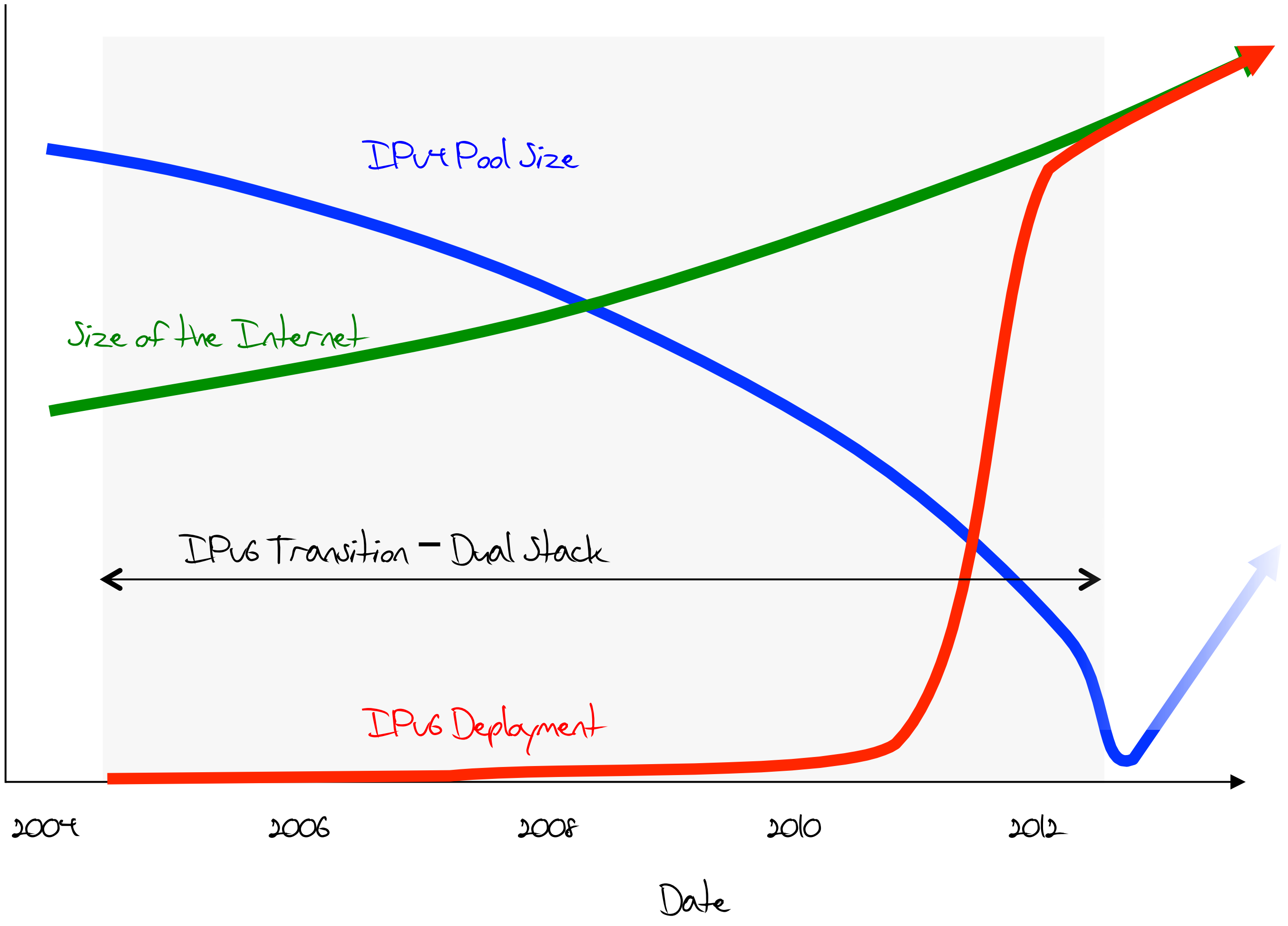


Fractions of a percent!

The Original IPv6 Plan c. 1995



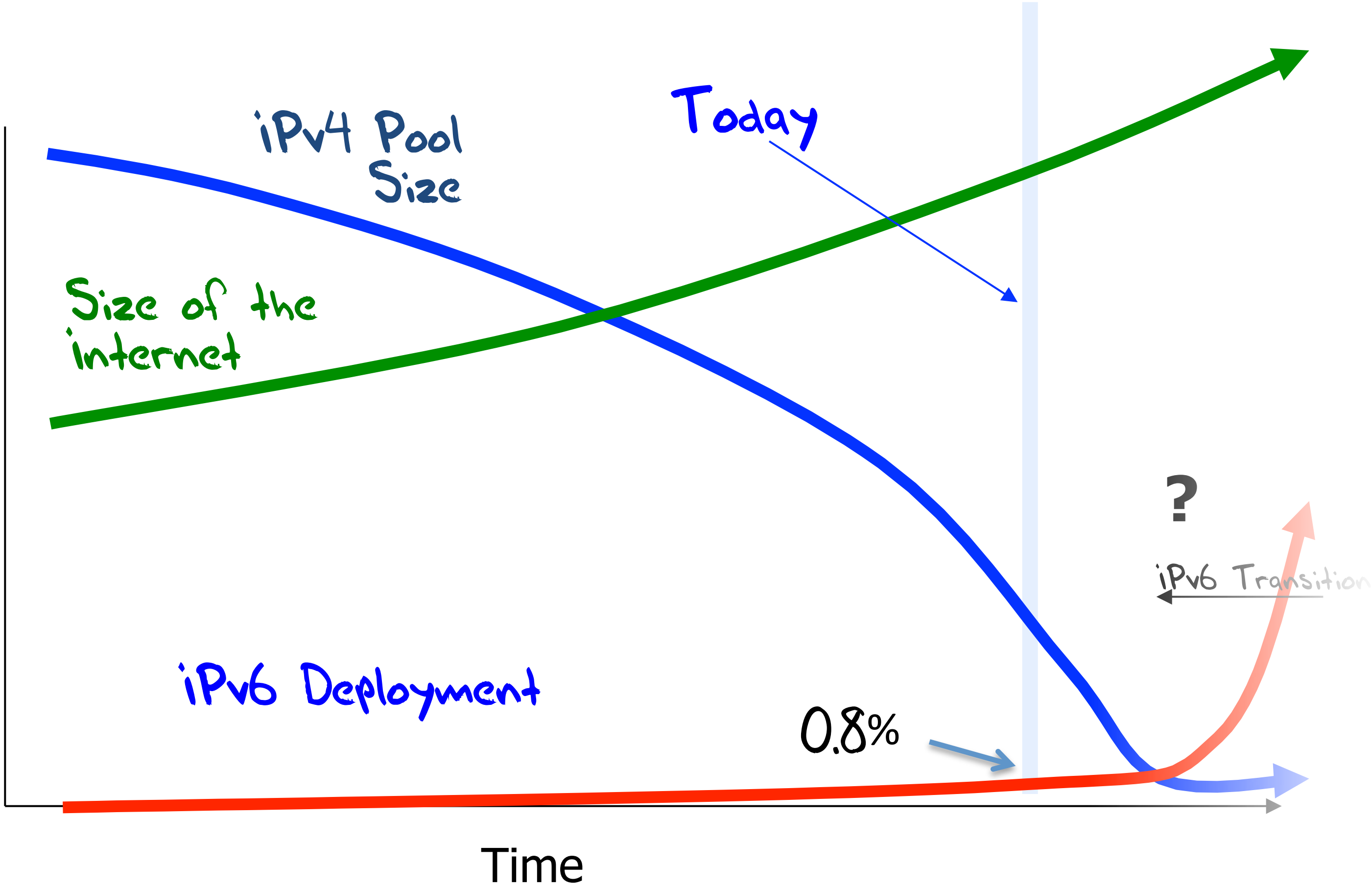
The Revised IPv6 Plan c. 2005



Oops!

We were meant to have completed the transition to IPv6 BEFORE we completely exhausted the supply channels of IPv4 addresses!

Today's Plan


iPv4 Pool Size
Today
Size of the internet
?
iPv6 Transition
iPv6 Deployment
0.8%
Time

Transition ...

The downside of an end-to-end architecture:

Transition ...

The downside of an end-to-end architecture:

- There is no backwards compatibility across protocol families
- A V6-only host cannot communicate with a V4-only host

Transition ...

We have been forced to undertake a "Dual Stack" transition:

Transition ...

We have been forced to undertake a "Dual Stack" transition:

Provision the entire network with both IPv4 AND IPv6

Transition ...

We have been forced to undertake a "Dual Stack" transition:

Provision the entire network with both IPv4 AND IPv6

in Dual Stack hosts configure the hosts' applications to prefer IPv6 to ipv4

Transition ...

We have been forced to undertake a "Dual Stack" transition:

Provision the entire network with both iPv4 AND iPv6

in Dual Stack hosts configure the hosts' applications to prefer iPv6 to ipv4

When the traffic volumes of iPv4 dwindle to insignificant levels, then its possible to shut down support for iPv4

Dual Stack Transition ...

Dual Stack Transition ...

We did not appreciate the operational problems with this dual stack plan while it was just a paper exercise

Dual Stack Transition ...

We did not appreciate the operational problems with this dual stack plan while it was just a paper exercise

The combination of an end host preference for IPv6 and a disconnected set of IPv6 "islands" created operational problems

- Protocol "failover" from IPv6 to IPv4 takes between 19 and 108 seconds (depending on the operating system configuration)
- This is unacceptably slow

Attempting to "bridge" the islands with IPv6-in-IPv4 tunnels created a new collection of IPv6 path MTU Discovery operational problems

- There are too many deployed network paths contain firewall filters that block all forms of IMCP, including ICMP6 Packet Too Big

Attempts to use end-host IPv6 tunneling also presents operational problems

- Widespread use of protocol 41 (IP-in-IP) firewall filters
- Path MTU problems

Dual Stack Transition

Signal to the ISPs:

- Deploy IPv6 and expose your users to operational problems in IPv6 connectivity

Or

- Delay IPv6 deployment and wait for these operational issues to be solved by someone else

So we wait...

And while we wait...

The Internet continues its growth

And without an abundant supply of IPv4 addresses to support this level of growth then the industry is increasingly reliant on NATs:

- Edge NATs are now the defacto choice for residential broadband services at the CPE

- ISP NATs are now the defacto choice for 3G and 4G mobile IP services

NATTing the Net

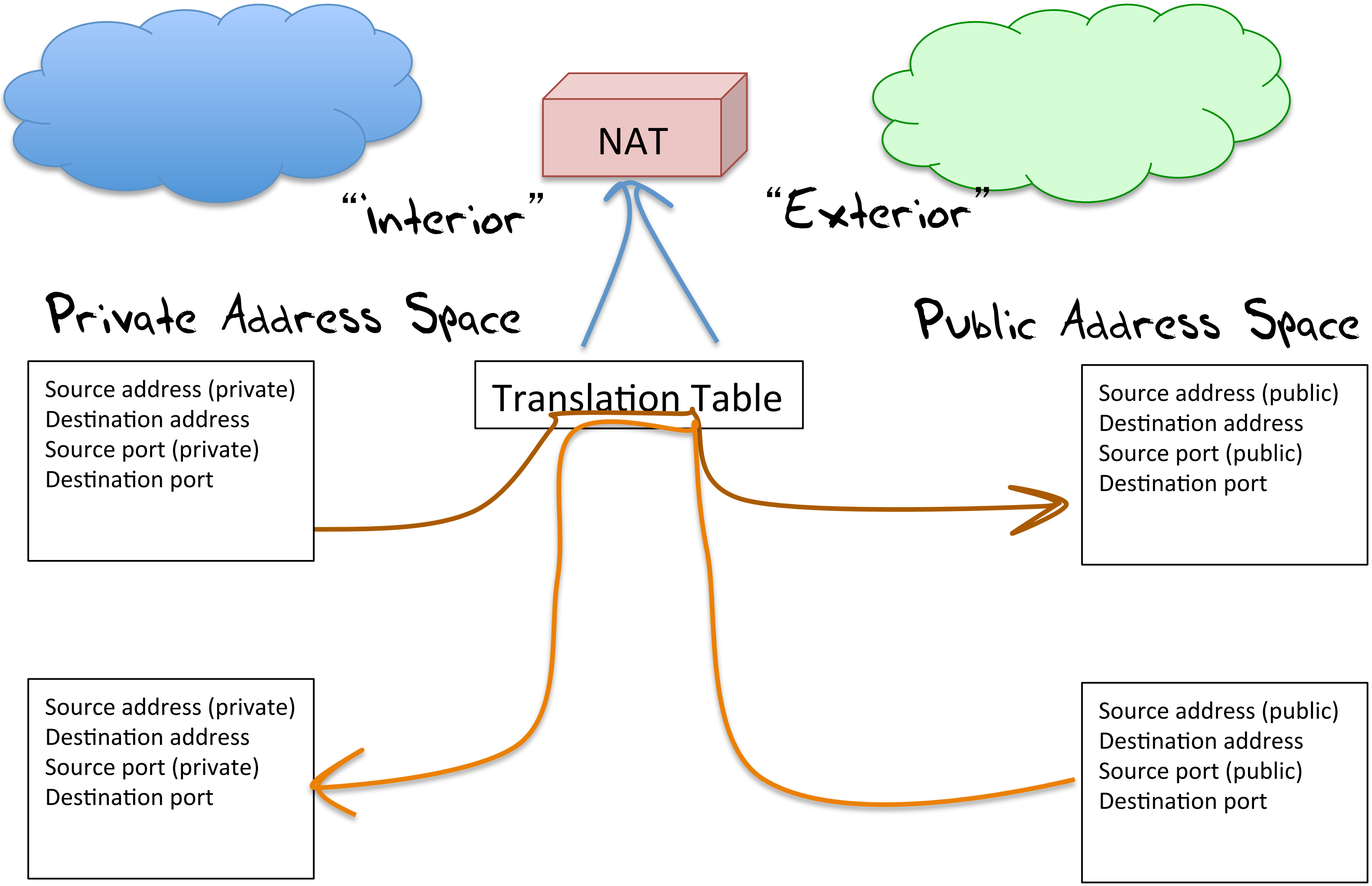
In 2012:

- The RIRs made 8,547 allocations to LIRs, allocating a total of 114M IPv4 addresses
- The routing table grew by 120M addresses
- The ISC host survey* indicates a growth of ~60M visible hosts
- BUT
  - in 2012 Apple sold ~110M iPhones and ~60M iPads and they have ~30% market share globally
  - This implies that some ~560M mobile devices were sold in the last 12 months

- It appears that the NATTed Internet grew by ~550M devices in the last 12 months!

* *http://www.isc.org/solutions/survey*

The Anatomy of NATs



NAT

Translation Table

Source address (private)
Destination address
Source port (private)
Destination port

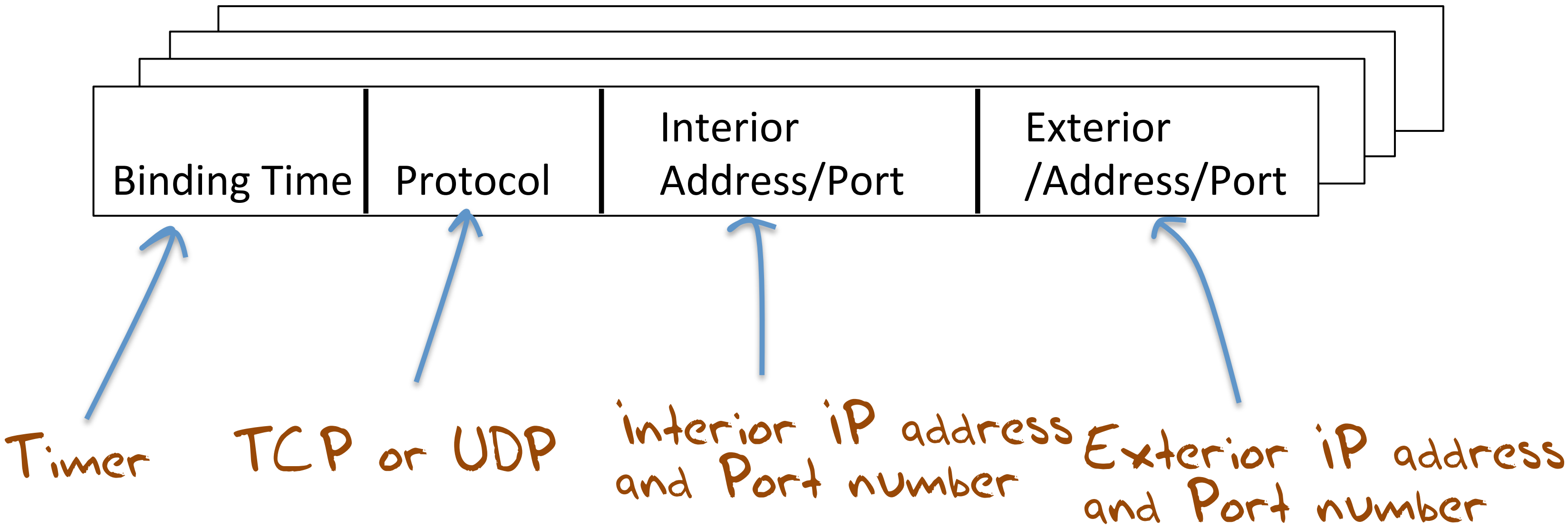
Source address (public)
Destination address
Source port (public)
Destination port

Source address (private)
Destination address
Source port (private)
Destination port

Source address (public)
Destination address
Source port (public)
Destination port

The Anatomy of NATs

Translation Table:



Binding Time	Protocol	Interior Address/Port	Exterior /Address/Port
Timer	TCP or UDP	interior iP address and Port number	Exterior iP address and Port number

Design Parameters

TCP

- Open NAT Binding:
  - interior SYN
- Access NAT Binding
  - Symmetric (same exterior IP address, same exterior port)
- Release NAT Binding:
  - interior RST
  - interior FIN?
  - Exterior FIN?
  - Exterior RST?
  - Timer?

UDP

- Open NAT Binding
  - interior packet
- Access NAT Binding
  - Symmetric (same exterior IP address, same exterior port)
  - Full cone (any exterior IP address, any exterior port)?
  - Restricted cone (same exterior IP address, any exterior port)?
  - Port-restricted code (any exterior IP address, same exterior port)?
- Release NAT Binding:
  - Timer?

Port Control Protocols

- STUN/TURN
- PCP relay of UPnP and NAT-PMP

Design Parameters

- Different NATs make different choices in these design parameters
- Applications then have to "discover" the particular behavioral type in order to perform non-trivial operations
- This adds delay, complexity and fragility to the service model of the network

TCP

- Open NAT Binding:
  - Interior SYN
- Access NAT Binding
  - Symmetric (same exterior IP address, same exterior port)
- Release NAT Binding:
  - Interior RST
  - Interior FIN?
  - Exterior FIN?
  - Exterior RST?
  - Timer?

UDP

- Open NAT Binding
  - Interior packet
- Access NAT Binding
  - Symmetric (same exterior IP address, same exterior port)
  - Full cone (any exterior IP address, any exterior port)?
  - Restricted cone (same exterior IP address, any exterior port)?
  - Port-restricted code (any exterior IP address, same exterior port)?
- Release NAT Binding:
  - Timer?

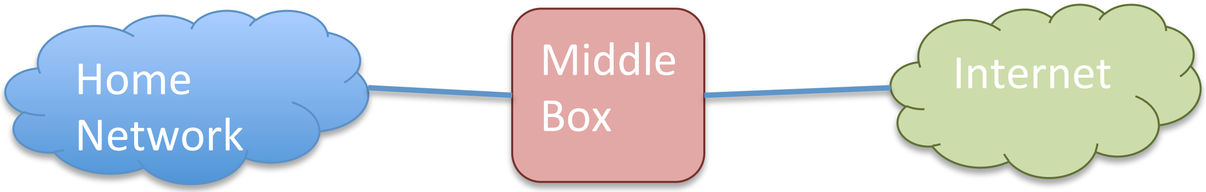
Port Control Protocols

- STUN/TURN
- PCP relay of UPnP and NAT-PMP

2 Party NATs
AKA Subscriber-Based NATs

Relieved pressure for iPv4 space

is nearly everywhere



Home Network

Middle Box

Internet

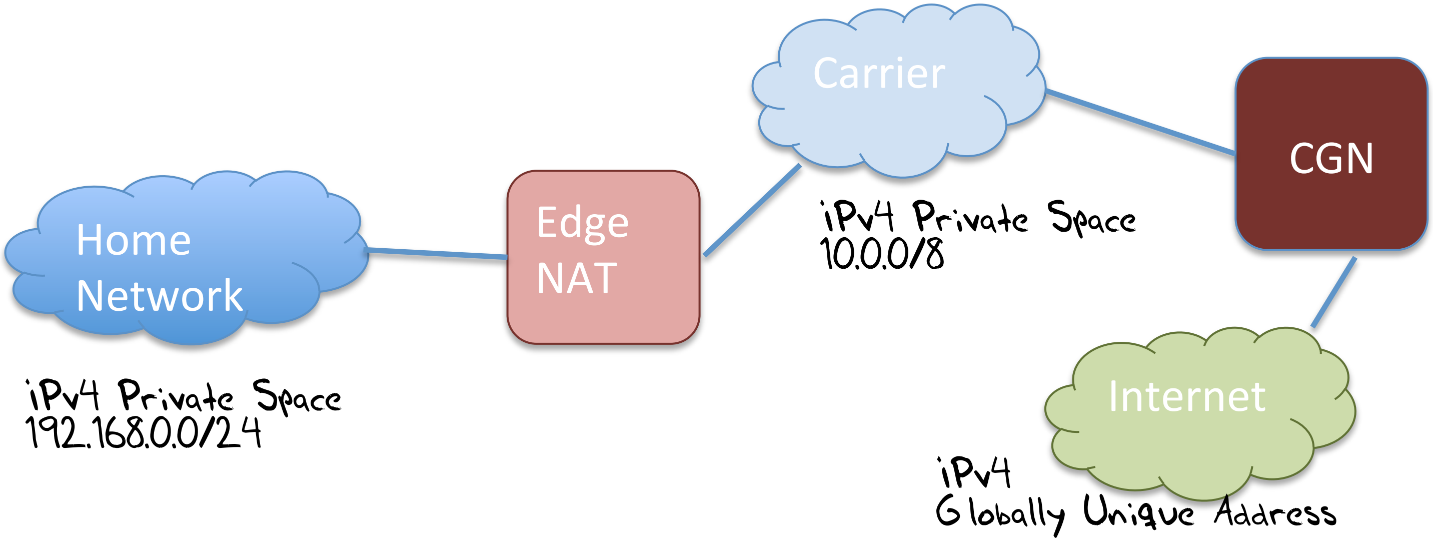
iPv4
192.168.0.0/24

iPv4
Globally Unique Address

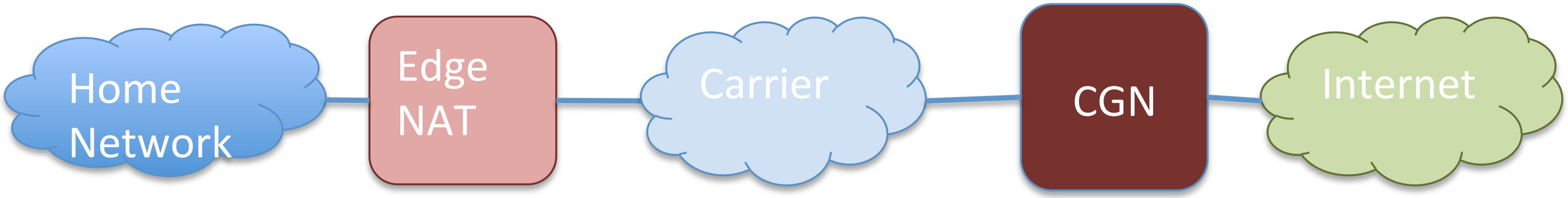
3 Party NATs
AKA Carrier Grade NAT

Adds a new non-unique realm in the Carrier

Adds more complexity but "slows" runout



Some Multi-NAT Issues



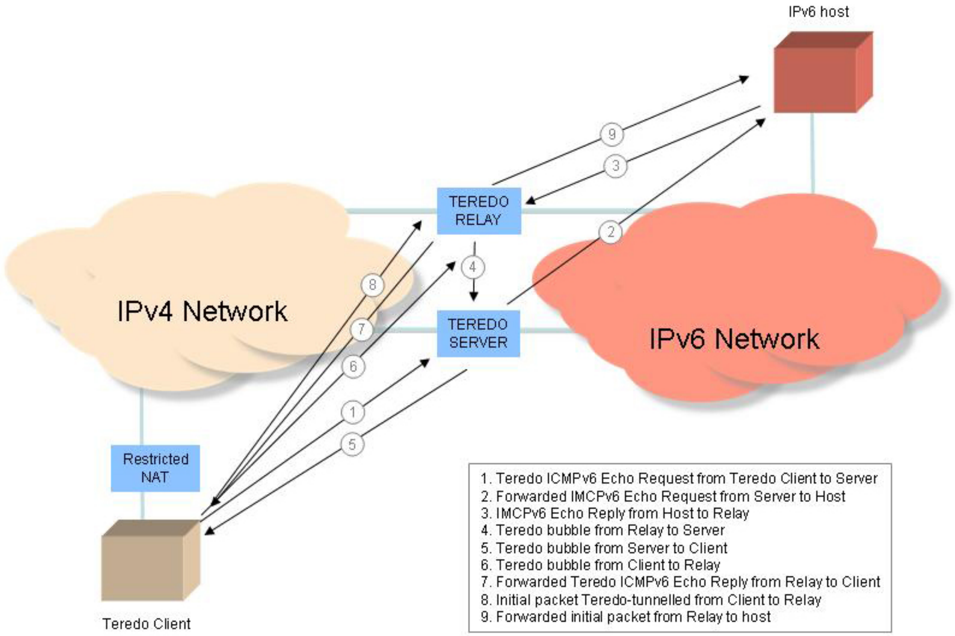
- What is the aggregate NAT binding behavior as seen by an application?
- How can an application "discover" this aggregate binding behaviour?
- Can an application determine how many NATs (and of what type) are in its data path?
- Does the carrier need a new private address space that is distinct from RFC1918 address space?
- How does home-to-home work in this model?
- Does this model become more complex with 3 NATs in series?

How Good Are NATs?

3-party rendezvous:

- A knows about B and C
- A tells B to contact C

Teredo is a good example here:



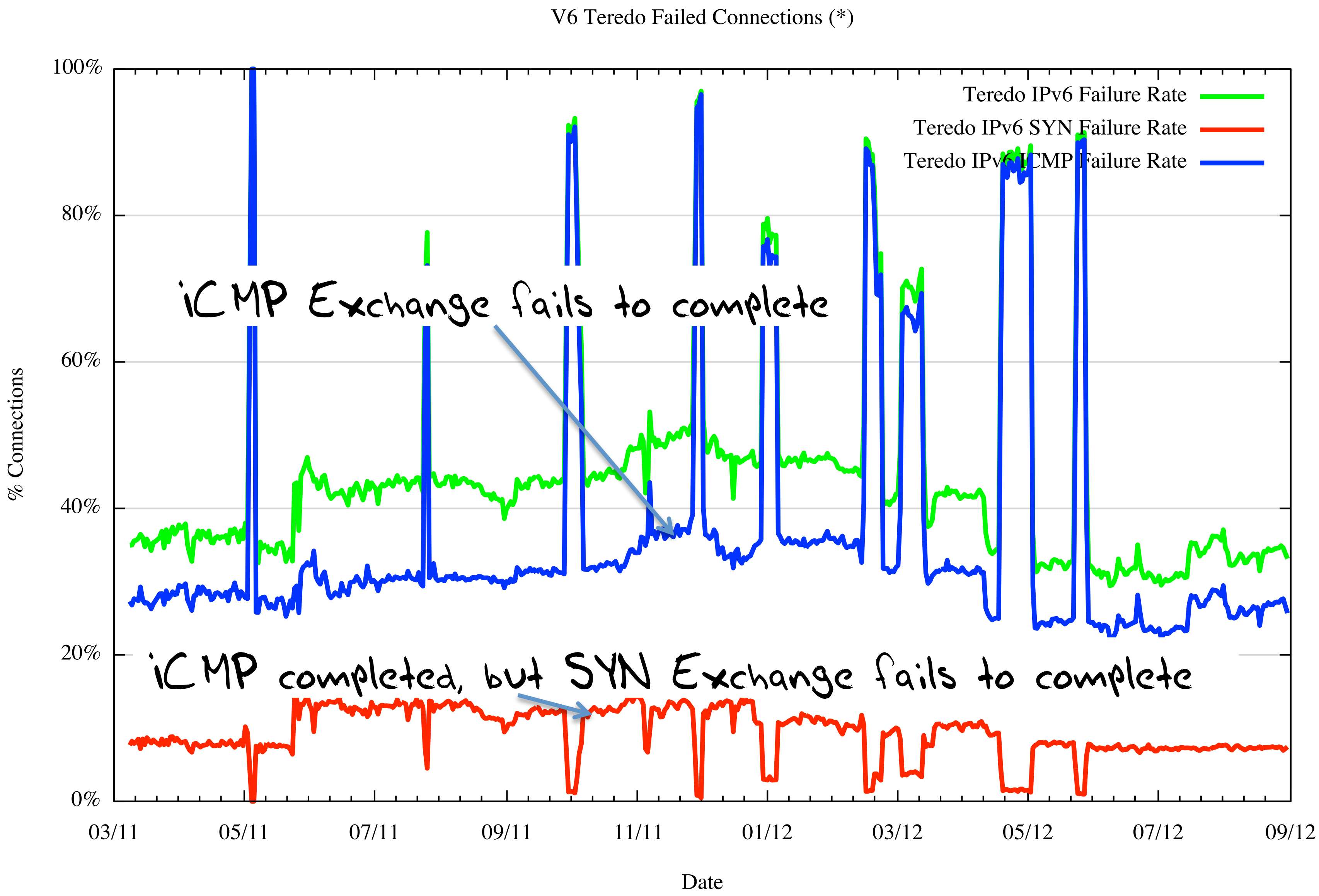
1. Teredo ICMPv6 Echo Request from Teredo Client to Server
2. Forwarded IMCPv6 Echo Request from Server to Host
3. IMCPv6 Echo Reply from Host to Relay
4. Teredo bubble from Relay to Server
5. Teredo bubble from Server to Client
6. Teredo bubble from Client to Relay
7. Forwarded Teredo ICMPv6 Echo Reply from Relay to Client
8. Initial packet Teredo-tunnelled from Client to Relay
9. Forwarded initial packet from Relay to host

NAT Failure

How well do NATs perform in supporting an application performing a 3-party rendezvous?

- One way to measure this is to test a common 3-party rendezvous application across a large number of clients
- So we measured it
- And we were pretty surprised

Teredo Failure Rate



ICMP Exchange fails to complete

ICMP completed, but SYN Exchange fails to complete

It's NAT Traversal Failure

Teredo failure is around 35% of all connection attempts

- Obviously, this is unacceptably high!
- This is unlikely to be local filtering effects given that Teredo presents to the local NAT as conventional IPv4 UDP packets
- More likely is the failure of the Teredo protocol to correctly identify the behaviour mode of the local NAT device
- The ICMP failure rate comes from the limited number of UDP NAT traversal models used by the Teredo handshake protocol vs the variance of UDP NAT traversal models used in networks
- The SYN failure rate is a result of the Teredo protocol making incorrect assumptions about the NAT's behaviour

Working with Failure

A 35% connection failure is unworkable is *almost* all circumstances

But one particular application can thrive in this environment – Bit Torrent:

- The massive redundancy of the data set across multiple sources reduces the sensitivity of individual session failures

All other protocols fail under such adverse conditions

CGN Deployment

What's the likely outcome of widespread CGN deployment on today's Internet?

- it's TCP, UDP or failure!
- it's simple client-server 2-party rendezvous or failure!
- it's network path symmetry, or failure!

Really simple transactions in a restricted application environment will still function, but not much else can be assumed to work

What's the New New Plan?

- If NATs make the network complex and fragile,
- And the IPv6 deployment program continued to proceed at a geological pace,
- Then what are we going to do to make the Internet work for the next 5 years of growth?

And don't say "SDN"

Or "OpenFlow"

What's the New New Plan?

How can we pull the Internet though a middleware dense environment for the next 5 years?

- What application models are robust in a CGN-dense world
- How do CGNs break?
- How variable are CGNs?
- What will applications need to cope with?

What would help

- Can we perform wide scale measurements of NAT robustness?
- is there improvements that can be learned from testing?
- How?

And what would not

inaction