IPv6 Transition - A Broader Perspective

Geoff Huston AM
APNIC

Why ?

Because we ran out of IP addresses

Again.

We've been here before ...

The original ARPAnet design from 1969 used the NCP protocol, which used 8 bit addresses in the NCP packet header

- Maximum network span of 256 nodes
- Enough?

- Well yes, because at the time computers were the size of entire rooms, cost many millions of dollars and there were only a few thousand in the entire world.
- At the time, the entire concept of shrinking the computer to something you could hold in one hand and trying to connect billions of them together was just too far into the future to worry about

Transition V1.0

- Turns out that 8 bits of addresses was not enough for the next generation of mini-computers
- ARPAnet undertook a transition from NCP to a new protocol: TCP/IP:
  - Expansion from 8-bit to 32-bit addresses
  - Flag Day: 1 January 1983
  - Shutdown and reboot every node into the new protocol!

Vint Cerf, APRICOT, Feb 2011

“This time, for sure!” *

* Actually Vint Cerf did not say that!

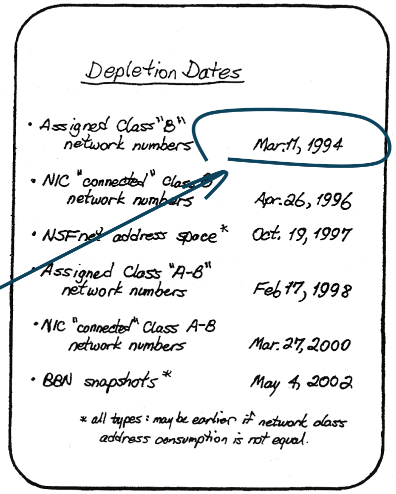
* Actually Vint Cerf didn’t say this!

IP Version 4

- 32-bit address field
  - That’s 4,294,967,296 addresses!
- In 1983 that looked like enough for a HUGE number of computers

Digital Pressures

- Through the 1980's computers changed from large expensive pieces of ironware to desktop consumer products
  - The size of the computer market changed from thousands to tens of millions
- And the prospect was even smaller, cheaper and more
  - By 1990 that 4 billion address space was looking pretty small
  - And it was not going to last much longer!

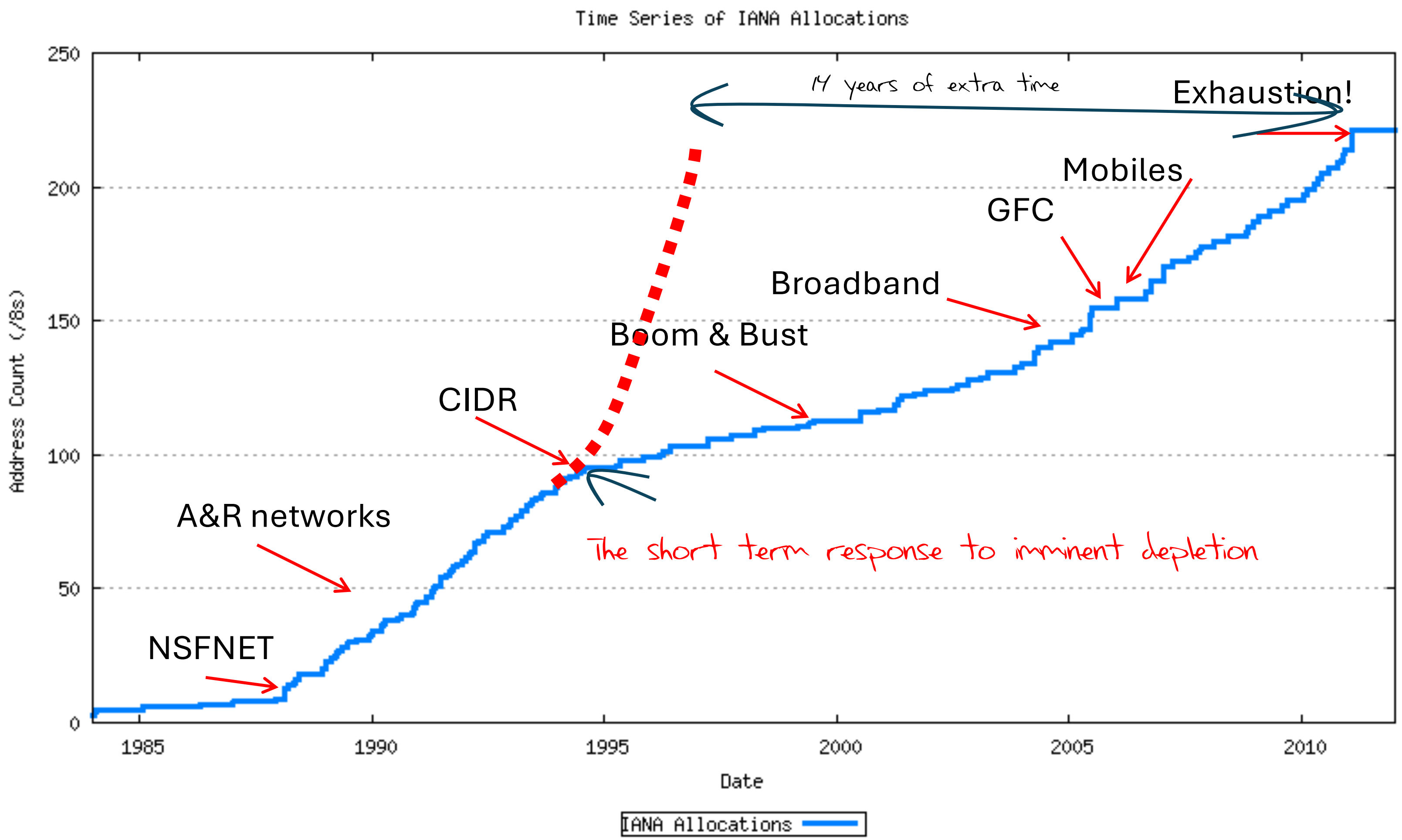


Frank Solensky – IETF 1990
Presentation on IPv4 Address Depletion Predictions

What did we do? Panic!

- This was a brutal wakeup call
  - We had hardly started with the Internet and its demise was just 4 years away!
  - We rapidly worked on short term responses to push this exhaustion date out
  - To give us more time to work on the longer term solutions
- So:
  - **We dropped the classful address plan**
  - **We introduced NATs to allow address sharing**
  - **We worked on a new protocol**

The short-term band aid worked!



That longer term plan

- Was to develop a new IP protocol
- And then transition the Internet over to use this new protocol

What is this new protocol?

IPv6!

IPv6 is…

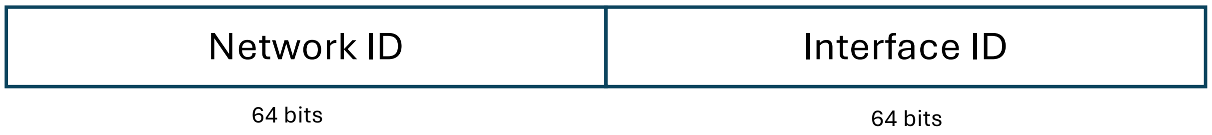
- IPv4 with larger address fields
- Not much else changed
  - It's still an address-based stateless datagram forwarding protocol
  - It still has decoupled forwarding, routing, naming, transport
  - The interfaces to the underlying media protocols are largely unchanged
  - The APIs to interface to the upper layers are largely unchanged
- The transition to IPv6 should be easy!

Aside: "Not much else changed"

What changed with IPv6:

- 128 bit address fields
- Semi-Fixed host/network boundary
- Replace Broadcast and ARP with Multicast and SLAAC
- Removed on-the-fly fragmentation with source fragmentation
- Make Fragmentation Controls an optional Extension Header
- No NATS!
- Flow Label
- Change “Options” to “Extension Headers”
- Multi-Addressing
- Scoped Addresses

128 bit addresses



- We experimented with a constant interface ID for a while, and realised that this was an unacceptable privacy leak if the value was held constant
- So clients use a temporary random interface identifier for the low order 64 bits
- Which means that the constant length of an IPv6 address is back to 64 bits

The Flow ID

- Huh?
- It probably sounded like a reasonable thing at the time
- But no one remembers what its good for!
- So that’s 20 waste bits in the header!

Extension Headers

- **Don’t work** on today’s silicon!
- High speed switching gear that wants to preserve TCP and UDP packet flows need to see the TCP and UDP headers at a fixed offset in the packet header
- Extension headers are a chained header structure where each header points to the next header, with the transport header at the end of the chain
- Unravelling that chain in very high speed silicon is a challenging task
- Easier to drop all IPv6 packets with extension headers!

Fragmentation Control

- Packet Fragmentation was a big feature of the IPv4 design
  - It allowed new media types to the integrated into the Internet environment seamlessly
  - But it had a performance cost
- The IPv6 design disallowed fragmentation on the fly
  - If a packet is too big for the next hop you need to send a signal to the source and discard the too-big packets
  - This is slow and unreliable
  - The signalling is also uinreliable
  - So the best response is to avoid packet fragmentation completely
  - Which means that packets in IPv6 tend be 1,280 octets
  - Which has a performance cost

No NATs!

- Really?
- We've grown addicted to NATs and IPv6 NATs are already a common feature of IPv6

Aside: "Not much else changed"

What’s giving us grief with IPv6:

- 128 bit address fields
- Semi-Fixed host/network boundary
- Replace Broadcast and ARP with Multicast and SLAAC
- Removed on-the-fly fragmentation with source fragmentation
- Make Fragmentation Controls an optional Extension Header
- No NATS!
- Flow Label
- Change “Options” to “Extension Headers”
- Multi-Addressing
- Scoped Addresses

Transition should be easy?

Right?

Transition should be easy?

Well, no, not really!

Why

Is this IPv6 transition taking us so long?

The answer to this question probably lies in the deregulated nature of the Internet and the economics of markets

Let’s take a look…

Organisational Models

A. Centralised Command and Control

Use a single body to:

- Define a common network architecture
- Devise and promulgate standards for suppliers to allow interoperation
- Centralized purchasing of equipment and services
- Enforce compliance from vendors through purchase prerequisites

This can ensure coherence and consistency across the network, but such models are prone to inefficiency and inflexibility

Organisational Models

B. Distributed Market-based Control

Operate the network as a loosely coordinated collection of markets for the supply of goods and services

It’s left to suppliers to formulate their own common specifications to promote interoperability across suppliers.

Consumers drive the evolution of the service platform through collective preferences for functionality and cost efficiency. Coherence and consistency is expressed as a consumer preference

Diverse market-driven systems can be quick to pick up on innovative new technologies in some cases, and can also be remarkably conservative and slow to innovate in other cases.

Organisational Models

We’ve had experience with both models:

- Public Telephone Service Networks were organized along highly centralized models, with the telephone operator acting as the point of command and control for the entire network
- The Internet was an outcome of deregulation of this PSTN environment, replacing the single command and control operator with a deregulated collection of suppliers and operators, coordinated through competitive market actions rather than by explicit constraints through regulation

How Transition works within the Internet

Think of transition along the same lines as any other market adoption:

- Individual actors will undertake their part of a transition if they see opportunities that directly accrue to them, or if they fear being left behind if they do not act
- Where there is a clear benefit, or clear cost saving, then the transition is more likely to proceed, otherwise the transition will stall
- But there is more to it than this:
  - Individual actors need to see individual benefits as a result of performing their part of any transition – irrespective of any common benefit, if this is not aligned to individual benefit then the transition stalls
  - Forced synchronization across large distributed systems is effectively impossible – individual actors need to perform their own transition in their own time
  - But no individual actor can stray too far away from the common pack – early and late adopters both run the risk of isolation and lost market share if they stray too far!

What Factors Assist a Transition

- **Backward compatibility**
  - Allows early adopters to proceed at their own timing, while still being able to interoperate with the entire network base
  - Lack of backward compatibility is a negative impediment to transition
- **Realizable individual benefit**
  - Allows early adopters to improve their competitive position through increased efficiencies or a greater value proposition of the product
- **Alignment of cost and benefit**
  - Those adopters who are gaining the benefit are exposed to the marginal cost. As long as the incremental cost is less than the benefit then there is a motivation to proceed with the transition
  - If cost and benefit are not aligned then some actors see the transition as a cost without benefit and will not embark on the transition

Negative Factors

- **Poor Market Signaling**
  - Each market actor needs to be aware of what others are doing in order to maintain rough coherence in the distributed space – if these signals are occluded or distorted the network is more prone to fragmentation
- **Network Fragmentation**
  - If the network starts to lose coherence then various actors will be subject to different perceptions of benefits and cost and the outcome is fragmentary
  - Such differences can arise from many factors, not the least of which is differing public regulatory frameworks in various economies

A Transition Scorecard

- ❑Backward compatibility
- ❑Realizable individual benefit
- ❑Alignment of cost and benefit
- ❑Market Signaling
- ❑Network Fragmentation

Why is this taking so long?

IPv6 was a minimal change to IPv4

- So early adopters found little in the way of an early adoption benefit

IPv6 was not backward compatible with IPv4

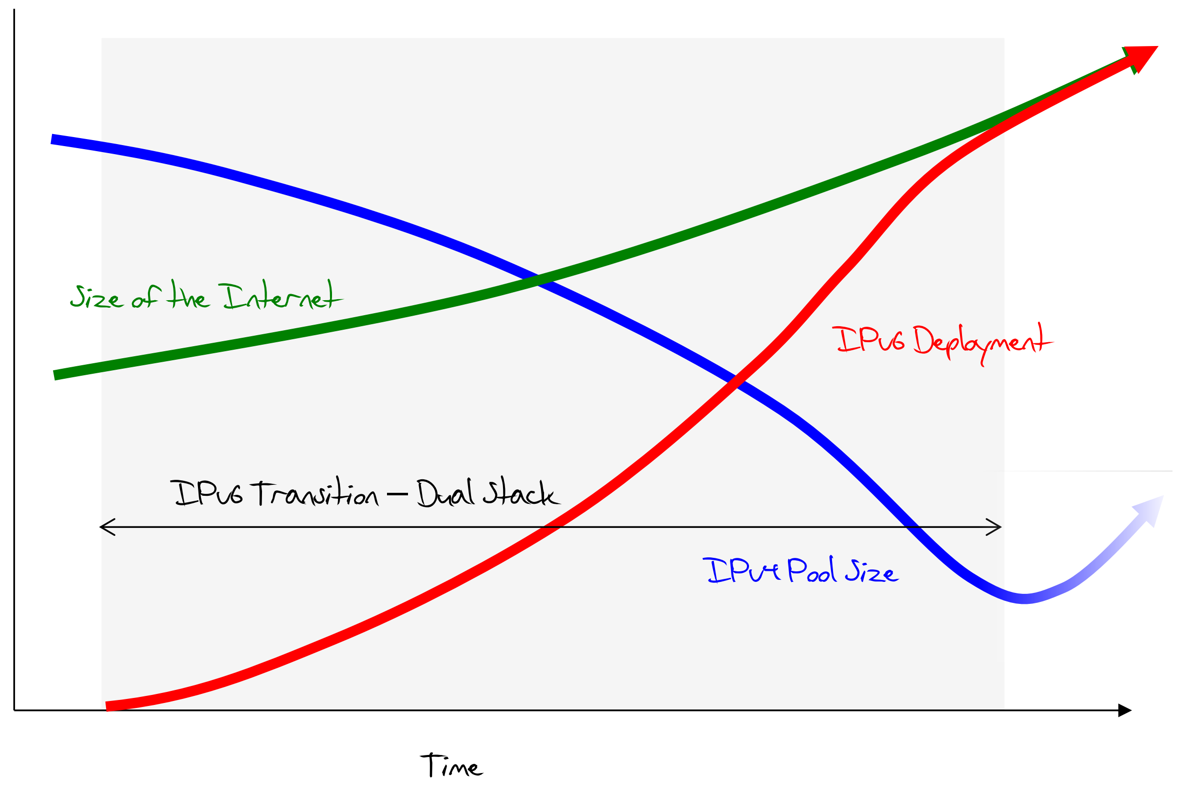
- This is a key "feature" of protocols that use fixed length addresses in their packet headers – you can't jam all 128 bits into a 32 bit field.
- This meant that IPv6 hosts could not directly communicate with IPv4 hosts
- IPv6 adopters still had to support IPv4
- **This meant that EVERYONE has to transition to dual stack BEFORE we can drop IPv4**

Transition for everyone

That last point is important – let me repeat it for you:

- **This meant that EVERYONE has to transition to dual stack BEFORE we can drop IPv4**
  - A “Flag Day” switchover is impossible
  - Piecemeal replacement won’t work either, as IPv6 is not backward compatible with IPv4
  - So, we need to run both protocols in tandem “for a while”
  - But bear in mind that one protocol has already run out of addresses
  - And network growth continues at record levels

What we thought was the IPv6 Transition Plan

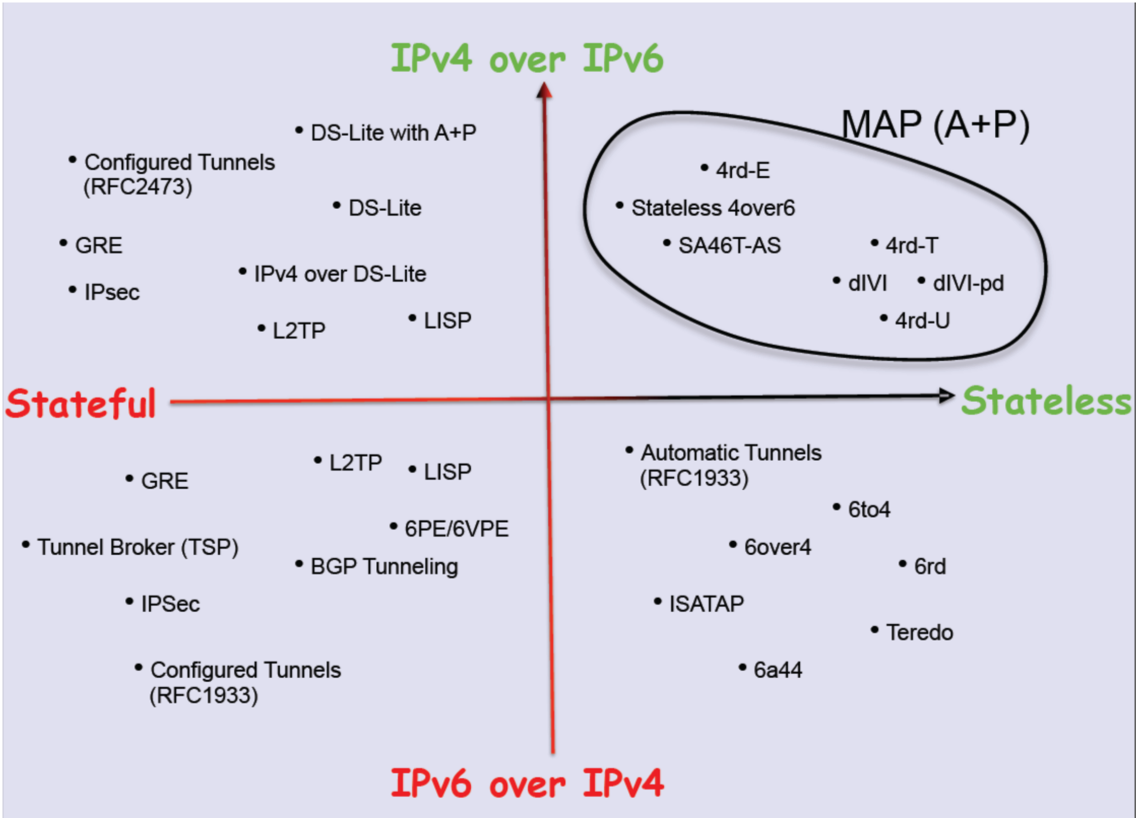


The Plan

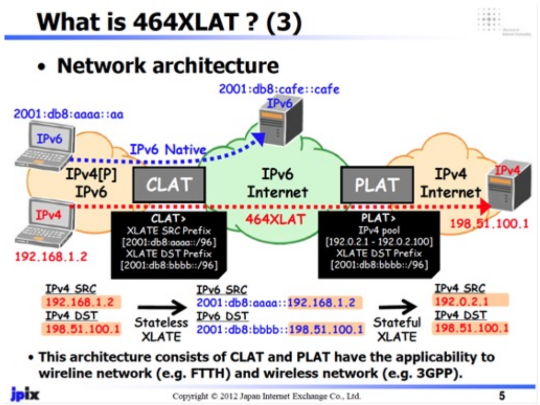
- We would build in a piecemeal fashion “islands” of dual stack networks while we still had adequate amounts of IPv4
- The IPv6 islands were meant to grow faster than the Internet itself
- Over time the dual stack “islands” would link up
- Once the entire Internet was dual stack we could drop IPv4

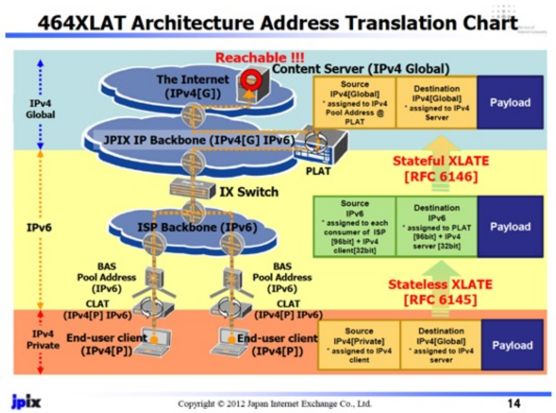
But Dual Stack is hard

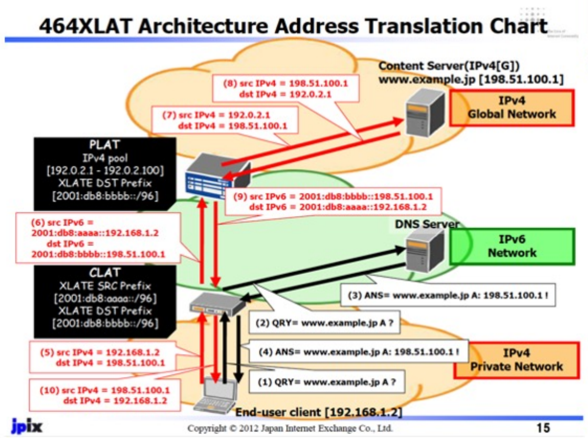
- Our only tools were tunnelling and/or translation



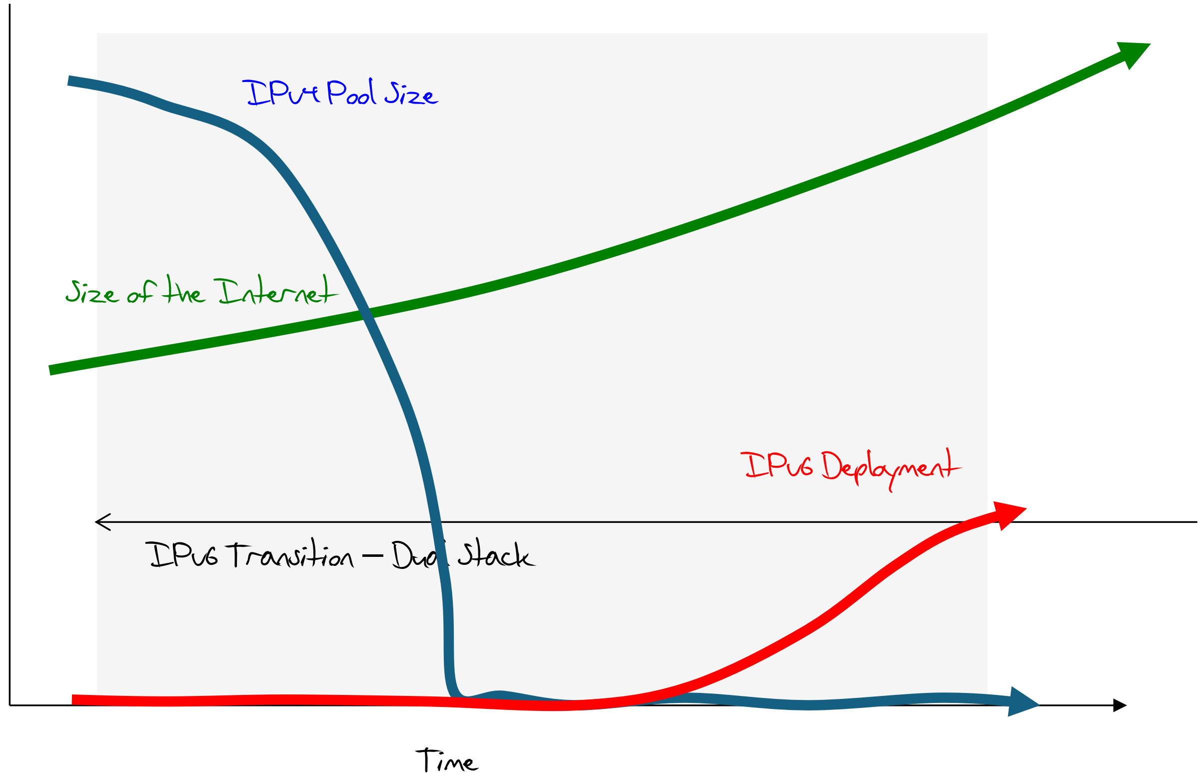
Complexity layered on complexity







The IPv6 Transition Plan as it happened!



Why did we make a mess of this transition?

"The Internet" does not exist

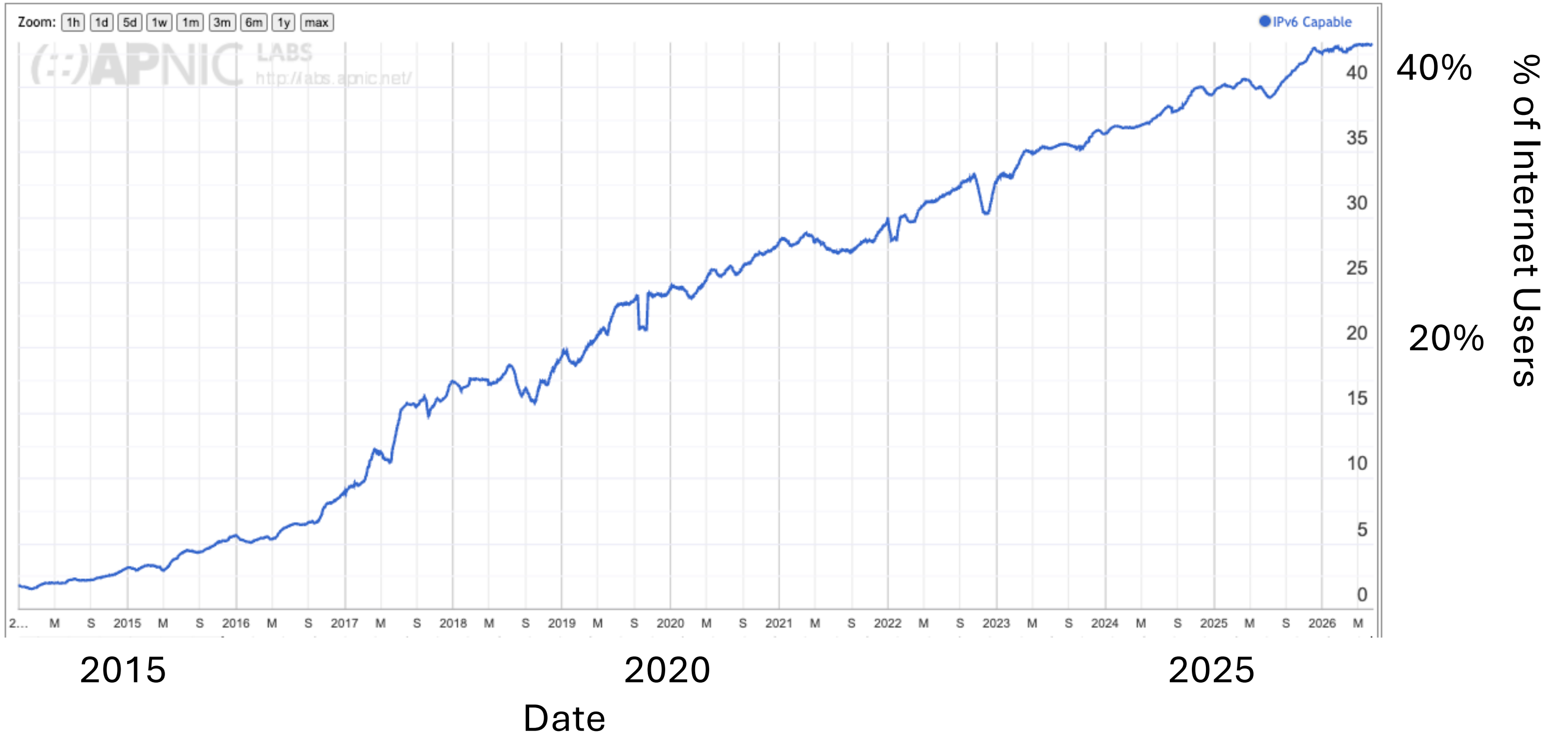
- The Internet is not a singly managed entity
- Noone is there to tell anyone else what they can (or cannot) do
- The Internet is a collection of markets (Goods and services, Transmission and switching, Technologies, …)
- We find it challenging to orchestrate collective synchronised actions across all these distinct activities
- The status quo accumulates a huge amount of inertial momentum
- Which makes it terribly hard to change direction in an orderly manner!

So, how is this transition going?

Slowly!

IPv6 transition progress

Use of IPv6 for World (XA)



It's a lot like this!

Why is this taking so long?

- The transition timetable is not being set by the early adopters
  - There is not enough competitive advantage for early adopters to drive the transition
- The transition timetable is not being set by the late adopters
  - At some point the last to adopt cannot impose a hold over everyone else
- Transition is a form of "majority" decision making
  - It's a case of "critical mass" that will determine at what point dual stack service providers will contemplate dropping IPv4 and completing their part of the transition

Why is this taking so long?

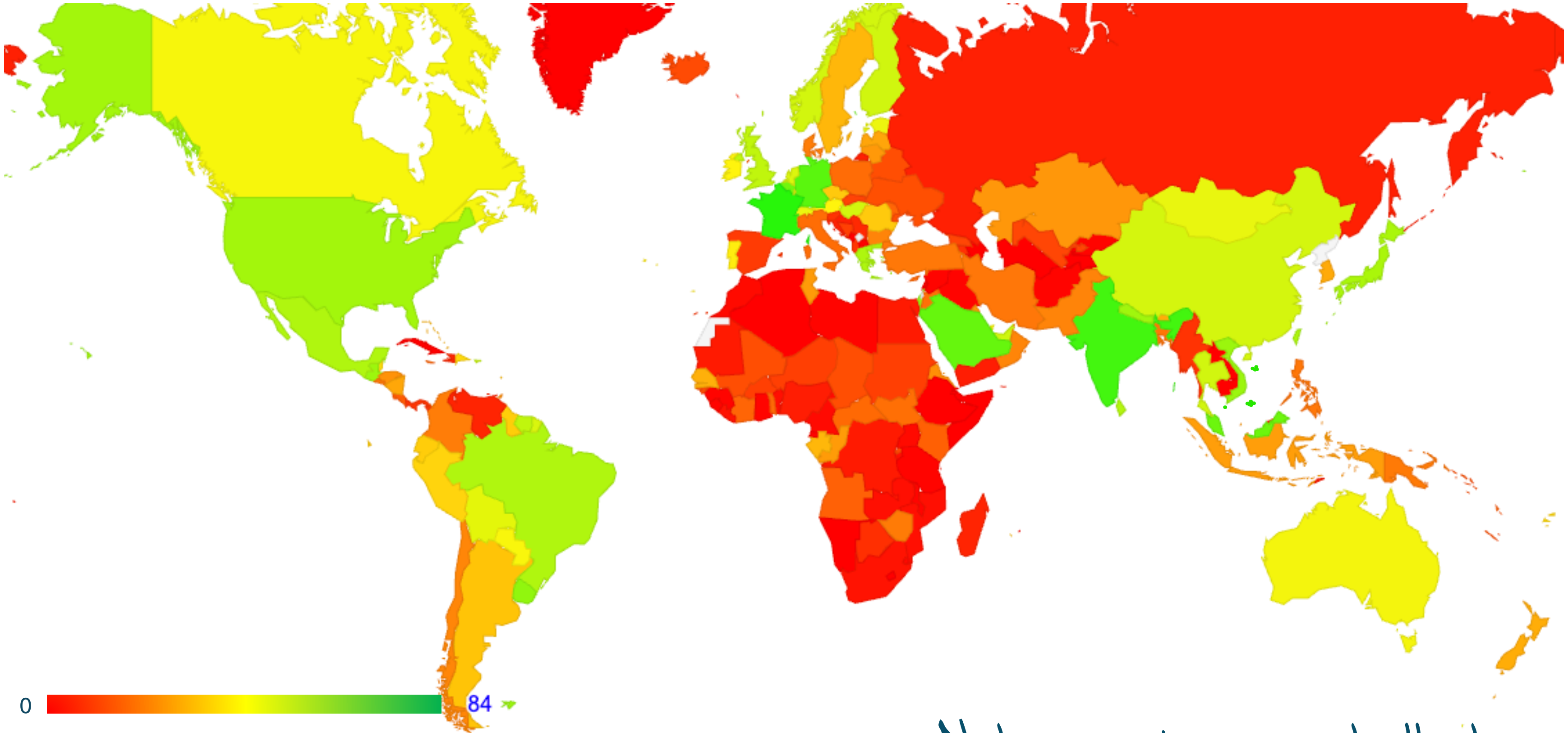
No common sense of urgency - NATs are just too good!

- NATs drive every part of today's internet, and we've adapted the entire IP infrastructure and application space to work in a NAT environment
- And we changed the architecture of the Internet into a client/server model
  - Where clients don't need permanently assigned public addresses
- NATs have allowed us to build a network with some 40 billion connected devices – and we are confident that we can grow further

The IPv6 Transition Scorecard

- ☒ Backward compatibility
- ☒ Realizable individual benefit Limited!
- ☒ Alignment of cost and benefit
- ☒ Market Signaling Limited!
- ❑ Network Fragmentation Mixed

IPv6 Adoption today



Not everywhere – not all at once

Will we ever complete this transition?

- Or is this the wrong question?

Longer Term Trends

Pushing EVERYTHING out of the network and over to the edge!

- Transmission infrastructure is becoming an abundant commodity
  - Sharing technology (multiplexing) is decreasingly relevant
- We have so much network and computing that we no longer have to bring consumers to service delivery points - instead, we are shifting services towards consumers and using the network to replicate servers
- With so much computing and storage the application is becoming the service, rather than just a window to a remotely operated service

Some issues to think about

What matters in today's networks?

- Addressing – IPv4 / IPv6 / NATs
  - Is universal unique end-point addressing a 1980's concept?
- Naming and Name Spaces – DNS evolution?
  - Are "names" a common attribute of the network, or an attribute of a service environment?
- Relative Location Frameworks?
  - In a world of densely replicated service delivery points how does a client rendezvous with the "best" service point? Does the client work it out? Or the network? Or the service?

From Addresses to Names

- An IP address is no longer an endpoint identity - It's an ephemeral session token to allow the network to distinguish the traffic between various active sessions
- Much of what we build upon these days is the name infrastructure (using the DNS)
  - It may not have been a deliberate choice, but we've been building name-based networks for the past decade or so
- And so far, it's been working well for us!
  - How long we can keep this up is anyone's guess!

Thanks!