

# IPV6 Failure as a Market Success

---

## Background

I wrote this in response to a discussion in which people lamented the failure to transition to [IPv6](#) and blamed the marketplace rather than using the failure as a learning experience. The advocates view problems in shifting from IPv4 to IPv6 as a “[market failure](#)”. But perhaps this is a case where the “market” is smarter and recognizes that IPv6 is too much for too little.

A [NAT](#), Network Address Translator is usually known as a “home router”. It allows all the computers on your local network to share a single connection from a provider.

## IPV6

NATs turned out to be the marketplace operator as a most efficient engine in the best tradition of capitalism.

I used to feel bad about making NATs the norm but I’ve since decided that the failure of IPv6 is a market success. It’s similar to the market’s wisdom in choosing [POTS](#) over [ISDN](#). Or, for that matter, [VHS](#) over [Beta](#).

By naïve metrics the market did indeed fail in each case. But in practice it worked – [POTS](#) was far better than ISDN by the measure that counted – cost and simplicity. You could stay online without a meter running. If ISDN didn’t have a meter running (and the implementation in the US wasn’t so botched) it would have been the norm. In fact, my effort at home networking started out as a way to make ISDN “just work”.

NAT gave users control over their own networks. By having the user’s home network to appear as a single device the carriers lost control over the interior networks (home or corporate LAN). The devices simply did not exist on the network as such. This gave reality to end-to-end (or edge-to-edge as people keep misinterpreting end-to-end).

The problem with the NAT is a failure of the early protocols IPv\* that made the IP address matter and compounded the initial problem by comingling naming and routing. Instead of V6 we need to go further and make the relationships between end points outside of the (service provider’s) network the primary center for protocol design because that’s where the edge really is. It isn’t even your computer. It’s the application such as a conversation in Skype.

Instead of IPv6 we need protocols that composite connectivity from the edge rather than continuing to put power in the center with number assignments and the DNS being critical dependencies. More about that if there is interest.

That said I don’t oppose IPv6 – it’s a stop gap measure but perhaps forcing the issue is more important. But for a while I was a V6 advocate in separating “edge” v6 from backbone V6. We could’ve used the v4 prefix to give interior machines a global appearance but I couldn’t get anyone to care about that as a priority.

Instead Microsoft did its own peer protocol which may or may not be bad. But they haven’t worked to promulgate it as an alternative.

For that matter think of all the effort that went into SIP as a rendezvous – why not fix the problem that so maintaining relationships is fundamental rather than just a special case? I see any use of @ as a symptom of market failure – effort being spent on V6 instead of understanding end-to-end and doing it for real.

PS: this whole marketplace failure idea seems rather naïve – efficient is a ratio not an absolute – so before telling me the marketplace failed, tell me your measure.

The purpose of telecom industry is to provide value to society. Instead it captures the value and that would make it a market failure on a grand scale.

## The Internet Itself

This is a related note I wrote in response to a question in the same thread asking what is wrong with today’s Internet anyway. It’s not a full response – just some quick thoughts in outline form. I’m sharing it because others expressed interest.

*Caveat, this is a quick comment rather than a full essay.*

To answer the question of what’s missing we need to think about how to create opportunity and how to make curiosity a sufficient reason to explore an idea. The problem is that if you see the Internet in terms of existing applications and tune for that (as providers are wont since that’s how they add value) then you make it difficult to go off the beaten path (lest you get beaten (OK, bad pun)).

The problem is that the Internet worked very well for its design point in creating opportunity (which is another way of saying that you could create new applications and protocols at the edge). But the design had a number of limitations:

1. The network or networks model is too hierarchical.
2. The IP address is problematic
  - a. No stable handles or routes
  - b. Addresses assigned by providers who also controlled routes
  - c. Compounded by protocols which embed the address in the data (as with [FTP](#))
  - d. 32 bits
  - e. The biggest problem is the comingling of names and routes. This is like the post office needing to track names not addresses.
3. Mainframe design
  - a. You connect to a machine not an application
  - b. Presumed mobility is the exception
  - c. Port number associated with single instances of applications such as [SMTP](#) servers
4. Providers and expedience
  - a. Encryption not necessary by assuming trusted providers
  - b. Since there's no encryption and lots of standards providers can indulge in second guessing "for your own good". This causes perverse behavior. Worse when it's for their own good.
  - c. Use of the IP address gives a provider control of the path and confines traffic to pipes.
5. The fixes are problematic
  - a. DNS is still centralized and compounds the problem by being used as a directory and making names so valuable you can't own your name.
  - b. IPv6 only addresses a small number of the issues while creating new problems with path-dependent addresses.
  - c. VPNs

- i. provide ways to tunnel through all this which is good
      - ii. They fragment the address space which is OK but reusing addresses makes reconciling realms difficult.
  - d. NATs provide a decoupling but
    - i. run afoul of the use of the IP address
    - ii. the LAN acts like a single mainframe
    - iii. As with VPNs the local address (which double as names thanks to IP) aren't globally valid.

For me the big problem is that we've forgotten that the goal is to support applications. Instead the myriad work-arounds are not part of a large whole because the commons is more like a network than an effort to support applications. This may seem to be nitpicking but I argue it's the whole game.

This is what [Ambient Connectivity](#) is about – returning to the good old days when you could just assume you could connect to any other point on the net except this time for real – that means without dependency upon providers or subscriptions or all the mechanisms and training wheels associated with the engineering prototype that is today's Internet.

PS: This curiosity thing comes from a conversation with Gerry Sussman and is a counter to the entrepreneurial meme that says you should only do things that have a business case.