April 2023
Geoff Huston

The Internet Twenty-Five Years Later

This article resulted from a request by The Internet Protocol Journal (IPJ), which will celebrate its 25th anniversary in June 2023. Another version of this article will appear in the June edition of IPJ, see: http://protocoljournal.org

The Internet not quite as young and spritely as you might've thought. Apple's iPhone, released in 2007, is now 16 years old, and YouTube is an ageing teenager at 18, after its initial release in 2005, and these two examples are relatively recent additions to the Internet. The first web browser, Mosaic, was released some thirty years ago in 1993. Going back further, the Internet emerged from its early ARPA roots in the form of the NSFNET in 1986. At the start of 1983 the ARPANET had a flag day and switched over to use TCP as its end-to-end transport protocol. Going back further, in 1974 Vint Cerf and Bob Kahn published the first academic paper describing the both the protocol and the underlying architectural framework of a packet switched network that became the Internet. This achievement was built upon earlier foundations, where there were a number of efforts in the late 1960's that showed the viability of a packet switched approach to computer networking. These packet-switched networking efforts included a program lead by Donald Davies at the UK National Physics Laboratory, an effort in the US in the form of an ARPA project lead by Larry Roberts, and Louis Pouzin's work in France with the CYCLADES network. This, in turn has some of its antecedents in work by Paul Baran at the RAND Corporation on distributed communications and packet switched networks, published between 1960 and 1964. Far from being an immature upstart, the Internet has managed to accumulate a relatively lengthy pedigree.

And it's been a wild ride. The Internet has undergone a number of cycles of economic boom and bust, each of which are right up there with history's finest episodes of exuberant irrational mania following by sobering correction. It's managed to trigger a comprehensive restructuring of the entire global communications enterprise and generated a set of changes that have already altered the way in which each of us now work and play. That's quite a set of achievements in just twenty-five years!

We start this exploration of the Internet's last quarter century in 1998. At that time any lingering doubts about the ultimate success of the Internet as a global communications medium had been thoroughly dispelled. The Internet was no longer just a research experiment, or an intermediate way stop on the road to adoption of the Open Systems Interconnect (OSI) framework. By 1998 there was nothing else left standing in the data communications landscape that could serve our emerging needs for data communications. IP was now the communications technology for the day, if not for the coming century, and the industry message at the time was a clear one that directed communications enterprises to adopt the Internet into every product and service or imperil their entire future in this business. No longer could the traditional telecommunications enterprises view the Internet with some polite amusement or even overt derision. It was now time for a desperate scramble to be part of this revolution in one of the world's major activity sectors. The largest enterprises in this sector, the old-world ex-monopoly telcos, had been caught wrong-footed in one of the biggest changes of the industry for many decades, and this time the concurrent wave of deregulation and competition meant that the communications industry's entire future was being handed over to a small clique of Internet players.

By 1998 the Internet had finally made it into the big time. The job was apparently done, and the Internet had prevailed. But soon after came a new revolution, this time in mobility services, where after a number of very clunky initial efforts by others (does anyone remember "WAP"?), the iPhone entered the market with a seamless blend of sleek design and astounding capability. The mobile carriage sector struggled to keep up with the new levels of rapacious demand for Internet-based mobile data. Riding on the back of this success, the Internet then took on the television networks, competing with the incumbent broadcast and cable systems with streaming video. The story is not over by any means. Communications continues to drive our world, and the Internet continues to evolve and change.

The evolutionary path of any technology can often take strange and unanticipated turns and twists. At some points simplicity and minimalism can be replaced by complexity and ornamentation, while at other times a dramatic cut-through exposes the core concepts of the technology and removes layers of superfluous additions. The technical evolution of the Internet appears to be no exception and this story contains these same forms of unanticipated turns and twists.

Rather than offer a set of unordered observations about the various changes and developments over the past twenty-five years, I'll use the traditional protocol stack model as a template, starting with the underlying transmission media, then looking at IP, the transport layer, then applications and services, and closing with a look at the business of the Internet.

Transmission

It seems like a totally alien concept these days, but the Internet Service Provider business of 1998 was still based around the technology of dial-up modems. The state of the art of modem speed had been continually refined, from 9600bps to 14.4kbps, to 28kbps to, finally, 56kbps, squeezing every last bit out the phase amplitude encoding space contained in an analogue voice circuit. Analogue modems were capricious, constantly being superseded by the next technical refinement, unreliable, difficult for customers to use, and on top of that they were just slow! Almost everything else on the Internet was forced to be tailored to download reasonably quickly over a modem connection. Web pages were carefully composed with compressed images to ensure a rapid download, and plain text was the dominant medium as a consequence. It could only get better, and it did.

The evolution of access networks was initially one that exposed the inner digital core of the network out to the edges. The first approach was ISDN, where the underlying digitised voice circuit was drawn out to the network's edge. At 64bps this was just an inadequate level of improvement over these analogue modems, and the next major step in access technology was to use DSL. DSL still used the last mile of the network into an analogue channel, but instead of running single low speed analogue bearer signal, DSL layered a large collection of individual analogue bearer signals into the access circuit, performing a form of frequency division multiplexing on the basic analogue circuit in a trellis framework. DSL relied on the combination of the telephone company's efforts to operate the copper access circuits within a high-level level of signal quality and noise suppression parameters, and the modem industry's continual incremental improvements in digital signal processing capability. Surprisingly, DSL was often able to achieve speeds of tens of megabits per second through these legacy copper access networks. However, DSL was largely an interim holding position while the search for a viable business model that was capable of underwriting the costs of deployment and use of an open fibre-based access networks was underway that was going to replace the aging copper access network.

The transition into fixed-wire access networks based on fibre optic cable continues. As noted already, the challenge in fibre access networking lies in finding a suitable business model than can sustain the necessary capital investment in replacing the existing copper-based network infrastructure. Some national communities used a model of a public sector program, such as the National Broadband Network program in Australia, while others have remained as dedicated private sector activities, and others take a hybrid approach with some level of local public sector incentives being added into a private sector program. The issue here is that fixed wire residential access networks do not offer compelling investment

opportunities in most cases, with the high upfront capital costs and the generally inadequate levels of take up across the dwellings passed by the access infrastructure acting as disinhibitory factors. It is often the case that a residential community cannot support multiple fibre access network deployments, bringing up the related issue of local access monopolies and the challenge of permitting some level of competitive access across a single physical access network. Nevertheless, fibre access rollouts continue across many parts of the world and the transition of the wired copper network into a fibre access network capable of sustaining hundreds of megabits per connection is still progressing, seemingly in spite of the barriers that exist in many scenarios.

The mobile access network has experienced a completely different evolution, and for many years now the mobile sector has been demand-driven. The first mobile data service networks, introduced in the 1980's, were little more than low-speed digital encoders working across a single voiceband circuit. These 1G networks typically delivered a 2.4Kbps data download capacity. The next generation of mobile services, 2G, was used in the 1990's. It was still predominately a voice service, and while it could theoretically support data access at speeds of 100kbps, this data transfer rate was largely aspirational. The intersection of the Internet and mobile services occurred with the introduction of 3G mobile services. The 3G architecture was capable of pushing IP connectivity directly out to the handset, supporting data transfer speeds of 1–2Mbps. This network capability, coupled with a new generation of handsets, first with the BlackBerry in 2002, and then the iPhone in 2007, transformed the mobile service into a mass market consumer service. The high margins available from the service captured the attention of the traditional voice industry and the result was a large scale opening up of radio spectrum to create an Internet access environment that quickly rivalled the wire-line access market in size, but totally outpaced it in terms of revenue margins. This massive escalation of demand created further pressures on the capacity of the mobile system, and in 2009 the mobile sector introduced 4G services, opening up more spectrum bands, and also adding Multiple-Input Multiple Output (MIMO) radio technologies to the mobile device to achieve greater deliverable capacity to each connected device. Over time these services were configured to deliver peak download speeds of 50Mbps to 100Mbps per device. The industry was also selling hundreds of millions of mobile devices per year. 4G dispensed with circuit-switched services, and it exclusively used packet switching. In 2018 5G was introduced. 5G can be supported more spectrum bands, including a high-band millimetre spectrum at 24-47Ghz. These higher carrier frequencies permit multi-gigabit data services, but they come at a cost of higher density of base station towers to compensate for the lower signal propagation distances associated with higher radio frequencies.

There was a second radio technology that emerged in 1998 that has also transformed the Internet, and it could be argued that it has become so fundamental that it has weaved itself so naturally into our environment that it all but disappeared.[1] The combination of low power radio systems and unlicensed radio spectrum allocation, or WiFi, and subsequently Bluetooth, has been transformational. The combination of efficient battery technology, computer chips that operate with low power consumption, and the unwiring of the last few meters in the home and office completely changed our collective of technology, and it is only because of our desire to use products that are not just portable but unobtrusively wearable, yet powerful enough to be useful that the component technologies such as batteries and processors have been pushed in this directions over the this period. While large bands of radio spectrum space have been allocated to cellular mobile service operators, the intensity of use and the utility of use of radio spectrum peaks in the unlicensed spectrum space used by WiFi and Bluetooth. It could be argued that the economic value of these unlicensed spectrum bands exceeds the exclusively licensed cellular radio systems by orders of magnitude. It could also be argued that the untethering of the last meter of the Internet transformed the Internet, and digital technologies in general, from a specialist pursuit into the consumer product space. In the nineties we used to describe the effort to simplify the use of technology through the term "plug and play". WiFi was the critical technical development that made that term irrelevant by removing any need for the plug, or the socket for that matter!

[1] To adapt a quotation from Mark Weiser, The Computer for the 21st Century, 1997 (https://web.archive.org/web/20141022035044/http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html)

Mobile data services, WiFi and Bluetooth really revolutionised the Internet, taking it from a "destination you visit" to an "always on utility in your pocket". The Internet was now a set of applications on a device that went with you everywhere. Always available, always connected, no matter where you might be or what you might be doing. But this was not exactly the full truth. Head out into remote country far enough, or head onto the world's oceans and your connection options quickly disappeared, leaving only somewhat expensive satellite-based services. These satellite services have been in operation since the early 1960's, but the high launch costs, limited capacity and the competing interests of terrestrial providers have meant that these services were often operated right at the margins of viability. The best example of this is Motorola's Iridium project of the late 1990's, where even before the entire service constellation of satellites was launched, the $5B Iridium project was declared bankrupt. Starlink, a recent entrant in the satellite service area, is using a constellation of some 4,000 low earth orbiting spacecraft and appears so far to have been able to break through this financial barrier. Using reusable launch vehicles, smaller (and lighter) satellites, transponder arrays on board, and a new generation of digital signal processing capabilities, Starlink is in a position to offer retail access services of 100Mbps or more to individual customers. The low altitude of the spacecraft means that the Starlink service competes directly with terrestrial access services in terms of performance. The introduction of inter-spacecraft laser links means that the system is capable of providing a service in any location, and the limiting factor, as with the Iridium effort decades ago, is obtaining the necessary clearances and licenses to have customers located in the respective national geographies. Starlink is certainly revolutionary in terms of capacity, coverage and cost. The questions are whether it is sufficiently revolutionary and whether it can scale up to provide a high-capacity service to hundreds of millions of users. At this point in time these are not easy questions to answer, but the limitations inherent in LEO-based services point to a potential advantage in terrestrial-based access networks. Nevertheless, Starlink is redefining the Internet access market space in many countries, and setting a price performance benchmarks that their terrestrial competitors now have to match.

If we move away from access networks to look at the changes in the "core" of the Internet over the past 25 years, then once more there has been dramatic change. In 1998 the Internet was constructed using the margins of oversupply in the telephone networks. In 1998 the core infrastructure of most Internet Service Providers was still being built by leasing telephone trunk supergroups (E-1 and T-1 circuits, and then E-3 and T-3 as capacity needs escalated, and then OC-1 circuits). While it was not going to stop here, squeezing even more capacity from the network was now proving to be a challenge. 622Mbps IP circuits were being deployed, although many of these were constructed using 155Mbps ATM circuits using router-based load balancing to share the IP load over four of these circuits in parallel. Gigabit circuits were just around the corner, and the initial exercises of running IP over 2.5Gbps SDH circuits were being undertaken in 1998.

In some ways 1998 was a pivotal year for IP transmission. Until this time IP was still just one more data application that was positioned as just another customer of the telco's switched circuit infrastructure. This telco infrastructure was designed and constructed primarily to support telephony. From the analogue voice circuits to the 64K digital circuit through to the higher speed trunk bearers, IP had been running on top of the voice network's infrastructure. Communications infrastructure connected population centres where there was call volume. The Internet had different demands. Internet traffic patterns did not mirror voice traffic, and IP performance is sensitive to every additional millisecond of delay. Constraining the Internet to the role of an overlay placed on top of a voice network was showing signs of stress, and by 1998 things were changing. The Internet had started to make ever larger demands on transmission capacity, and the driver for further growth in the network's infrastructure was now not voice, but data. It made little sense to provision an ever-larger voice-based switching infrastructure just to repackage it as IP infrastructure, and by 1998 the industry was starting to consider just what an all-IP high-speed network would look like, building an IP network all the way from the photon in a fibre optic cable all the way through to the design of the Internet application.

At the same time, the fibre optic systems were changing with the introduction of Wave Division Multiplexing (WDM) in fibre systems. Older fibre equipment with electro-optical repeaters and PDH multiplexors allowed a single fibre pair to carry around 560Mbps of data. WDM allowed a fibre pair to

carry multiple channels of data using different wavelengths, with each channel supporting a data rate of up to 10Gbps. Channel capacity in a fibre strand was between 40 to 160 wavelength channels using Dense WDM (DWDM). Combined with the use of all-optical amplifiers, the most remarkable part of this entire evolution in fibre systems is that a cable system capable of an aggregate capacity of a terabit per second can be constructed today for much the same cost as a 560Mbps cable system of the mid '90s. That's a cost efficiency improvement of a factor of one million in 25 years. The drive to deploy these high capacity DWDM fibre systems was never based on expansion of telephony. The explosive growth of the industry was all to do with supporting the demand for IP. So, it came as no surprise that at the same time as there was increasing demand for IP transmission there was a shift in the transmission model where instead of plugging routers into telco switching gear and using virtual point-to-point circuits for IP, we started to plug routers into wavelengths of the DWDM equipment and operate all-IP networks in the core of the Internet.

It was not just DWDM that has fundamentally changed these core transmission systems in the past 25 years. Two further technologies have been transformational in the fibre optic area. The first is the use of optical amplifiers. Erbium Doped Fibre Amplifiers provide a highly efficient means of signal amplification without the need to convert the signal back unto a digital form and then passing it back through a digital/analogue converter to modulate the next stage laser driver. This has allowed fibre systems to support terabit per second capacity without necessarily have to integrate terabit per second digital systems. The second fundamental change was a switch in signal modulation from a basic on/off signal into a signal modulation technique that uses signal phase, amplitude and polarity to increase the total capacity of a wavelength within a fibre strand. The key technology here are the digital signal processors (DSPs) and as we improve on the track width of conductor tracks in these DSPs we can increase the gate count on a single chip which allows more complex signal manipulation algorithms to be supported which can be used to increase the sensitivity of the DSP function. In 2010 we were using 40nm track silicon chips in DSPs, supporting Polarization Mode Quadrature Phase Shift Keying (PM-QPSK) which allowed a cable to operate with 100Gbps data rates in a single wavelength, or an aggregated of 8Tbps in a fibre strand. In 2023 DSPs now use 5nm tracks, which can support PCS-144QAM modulation of a base 190Gbaud signal, which can support 2.2Tbps data rates per wavelength, or 105Tbps total capacity per fibre strand. A 12-strand cable would have a total capacity of 1.2Pbs.

Such very high-performance fibre cable systems are generally used in submarine cable systems to link data centres between continents. In data centre contexts and other terrestrial scenarios, we are now using 200G and 400G per wavelength fibre systems as the common technology base. The major outcome of this is that, in general, transmission is no longer a scarce resource. It is in every sense of the term an abundant commodity. There is no sense in rationing access to communications capacity, be it short or long haul. This is not only a major change in the economic framework of the communications industry, but in phrasing the way in which we use communications. In a scarce system we trend to use "just in time" delivery mechanisms, passing content across the communications system only when it is needed, while an abundant system allows us to use "just in case" delivery mechanisms, which has had a dramatic impact on the architecture of the Internet. Indeed, this extraordinary increase in the underlying capacity of our communications infrastructure through the past 25 years is perhaps the most significant change in the entire landscape of the Internet, as we'll see when we look at content networking.

In network operations, we are seeing some stirrings of change, but it appears to be a rather conservative area, and adoption of new network management tools and practices takes time.

The Internet converged on using the Simple Network Management Protocol (SNMP) more than a quarter of a century ago, and despite its security weaknesses, its inefficiency, it's incredibly irritating use of Abstract Syntax Notation One (ASN.1), and its use in sustaining some forms of Distributed Denial-of-Service (DDoS) attacks, it still enjoys widespread use. But SNMP is only a network monitoring protocol, not a network configuration protocol, as anyone who has attempted to use SNMP write operations can attest. The more recent Network Configuration Protocol (NETCONF) and the Yet Another Next Generation (YANG) data modelling language are attempting to pull this area of

configuration management into something a little more usable than Command-Line Interface (CLI) scripts driving interfaces on switches.

At the same time, we are seeing orchestration tools such as Ansible, Chef, Network Automation and Programmability Abstraction Layer with Multivendor (NAPALM) and SALT enter the network operations space, permitting the orchestration of management tasks over thousands of individual components. These network operations management tools are welcome steps forward to improve the state of automated network management, but it's still far short of a desirable endpoint. The desired endpoint of a fully automated network management framework is still far from our reach. Surely it must be feasible to feed an adaptive autonomous control system with the network infrastructure and available resources, and allow the control system to monitor the network and modify the operating parameters of network components to continuously meet the service-level objectives of the network? Where's the driverless car for driving networks? Maybe the next 10 years might get us there.

The Internet Layer

If our transmission systems have been subject to dramatic changes in the past quarter century, then what's happened at the IP layer over the same period?

Firstly, we need to consider the elephant in the Internet layer room. There was one fundamental change at the Internet level of the protocol stack that was meant to have all happened some twenty years ago, and that's the transition to IP version 6. Twenty-five years ago, in 1998, we were forecasting that we would've consumed all the remaining unallocated IPv4 addresses by around 2025. That gave us slightly more than 25 years, so there was no particular sense of urgency. We didn't need to ring the emergency bell or raise any alarms. The overall aim was to proceed in an orderly manner. Things took a different course because we failed to appreciate the true impact of the shift of the Internet to mobile devices. All of a sudden, we were dealing with an Internet with billions of users, using billions of new mobile devices, and our comfortable predictions of a stately and steady rundown of the IPv4 address pools was thrown out of the window about as quickly as you could say the word "iPhone". From "all the time in the world" we swung to "no time left to do anything" within a single year. In the five-year period, when mobile services exploded in volume, between 2005 and 2010, the total count of allocated IP addresses rose from 1.5B addresses to 3.1B, from a total address pool of 3.7B addresses. The network had doubled in size, and the time left to complete the transition had shrunk from more than twenty years to a little over one!

At that point all the plans for an orderly transition were thrown out, and many network administrators scrambled to obtain IPv4 addresses, which further depleted IPv4 pools. The central pool of IPv4 addresses, operated by the IANA, was exhausted in February 2011. APNIC depleted its IPv4 pool in April of that year, the RIPE NCC 18 months later, LACNIC in 2014 and ARIN in 2015. We had expected that this situation would motivate network operators to hasten their plans for IPv6 deployment, yet, perversely, this was not the case. Less than 1% of the Internet's user base were using IPv6 in 2011. Five years later, as each of the RIRs ran down their remaining pools of IPv4 addresses, this internet-wide IPv6 user count had increased to just 5%. In 2023 the process is still underway and only some 35% of the Internet's user base has IPv6 capability. I'm not sure anyone is willing to predict how long this anomalous situation of running the IPv4 Internet on an empty tank will persist.

How has the Internet managed to continue to operate, and even grow, without a supply of new IPv4 addresses? In a word, the answer is "NATs". While the Network Address Translator (NAT) concept received little fanfare when it was first published, it has enjoyed massive deployment over the past 25 years, and today NATs are ubiquitous. The application architecture of the Internet has changed, and we are now operating a client/server framework. Servers have permanent IP addresses, while clients "borrow" a public IPv4 address to complete a transaction and return it back to a common pool when they are done. Time-sharing IP addresses, and also using the 16-bit source port field in TCP and UDP, has managed to extend the quasi-IPv4 address space by up to 20 bits, making the IPv4+NAT address space up to a million times larger than the original 32-bit IPv4 address space. In practice, the story is a little more involved than that, and some very large service providers have reached logistical limits in using

NATs compensate the exhaustion of IPv4 addresses. This situation has motivated them to transition to a dual stack mode of operation, and they are relaying on a dual stack host behaviour that prefers to use IPv6 when possible, so the dual stack deployment relieves the pressure on the IPv4 NAT functions.

NATs have prompted a significant change at the IP level in changing the default assumption about the semantics of an IP address. An IP address is no longer synonymous with the persistent identity of the remote party but has assumed the role of an ephemeral session token. The leisurely pace of the IPv6 transition is partly due to this altered role of addresses, as we no longer require every connected device to have a persistently assigned globally unique IP address.

IPv6 and NATs are not the only areas of activity in the Internet layer in the past 25 years. We've tried to change many parts of the Internet layer, but few, if any, of the proposed changes have managed to gain any significant traction out there in the network. The functions performed at the internet layer of the protocol stack are no different from those of 25 years ago. IP Mobility, Multicast and IP Security (IPSec) are largely Internet layer technologies that have failed to gain significant levels of traction in the marketplace of the public Internet.

Quality of Service (QoS) was a hot topic in 1998, and it involved the search for a reasonable way for some packets to take some form of expedited path across the network, while other packets took an undifferentiated path. We experimented with various forms of signalling, packet classifiers, queue management algorithms and interpretations of the Type of Service bits in the IPv4 packet header, and we explored the QoS architectures of Integrated and Differentiated Services in great detail. However, QoS never managed to get a toehold into mainstream Internet service environments. In this case the Internet took a simpler direction, and in response to not enough network capacity, we just augmented the network to meet demand. Again, this is an aspect of the altered mindset when the communication system shifts from scarcity and rationing to one of abundance. We've given up installing additional intricate mechanisms in the network, in host protocol stacks and even in applications to negotiate how to share insufficient network capacity. So far, the simple approach just adding more capacity to the network has prevailed, and QoS remains largely unused.

The switch from circuit switching to packet switching has never managed to achieve universal acceptance. We've experimented with putting circuits back into the IP datagram architecture in various ways, most notably with the Multi-Protocol Label Switching (MPLS) technology. This technology used the label swapping approach that was previously used in X.25, Frame Relay and ATM virtual circuit switching systems, and created a collection of virtual paths from each network ingress to each network egress across the IP network. The original idea was that in the interior of the network you no longer needed to load up a complete routing table into each switching element, and instead of performing destination address lookup you could perform a much smaller, and hopefully faster, label lookup. This performance differentiator did not eventuate and switching packets using the 32-bit destination address in a fully populated forwarding table continued to present much the same level of cost efficiency at the hardware level as virtual circuit label switching. However, there is one aspect of MPLS and similar approaches that has proved to be invaluable for many network operators. A general-purpose network utility has many disparate client networks, and a single packet-switched environment does not allow the network operator to control the way in which the common network resource is allocated to each client network. It also does not readily support segmentation of reachability. Virtual circuit overlays, such as MPLS provide mechanisms to control resource allocation and constrain cross-network leakage, and for many network operators these are adequate reasons to head down an MPLS-like path for their network platform.

Moving sideways at this level of the protocol stack we probably should look at the evolution of routing technologies. The early '90s saw a flurry of activity in the routing space, and various routing protocols were quickly developed and deployed. By 1998 the conventional approach to routing was the use of either IS-IS or OSPF as the interior routing protocol, and BGP-4 as the inter-domain routing protocol. This picture has remained constant right up to today. In some ways it reassuring to see a basic technology that is capable of sustaining a quite dramatic growth rate through many years of scaling, but in other ways it

is less reassuring to see that the unresolved issues we had with the routing system in 1998 are largely still with us today.

The largest of these unresolved issues lie in the trust we place in the Internet's inter-domain routing system. There is no overall orchestration of the Internet's routing system. Each network advertises reachability information to its adjacent networks and selects what it regards as the "best" reachability information from the set received from these same network peers. This mutual trust that each network places in all other networks can, and has, been abused in various ways. The effort to allow each routing entity to distinguish between what is a "correct" item of routing information and what is "false" route has a rich history of initiatives that have faltered for one reason or another. The most recent effort in this space is built upon the foundations of the number system and uses the association of a public/private key pair with the current holders of addresses and autonomous system numbers. This allows these holders to issue signed authorities about the use of these number resources in the context of routing, and by coupling these authorities with the information being propagated in the routing system, it is intended that unauthorized use cases can be detected. This effort, the Resource Public Key Infrastructure (RPKI) has achieved some level of acceptance in the networking space, and in 2023 around one third of all route objects have associated RPKI credentials. The work is still "in progress" as the more challenging aspect of this work is to associate verifiable credentials with a route's propagation through the network that does not impose onerous burdens on the routing system and is not overly fragile in its operation. The extended period where the routing system has operated in an essentially untrustable state has prompted the application layer to generate their own mechanisms of trust. These days it's largely left to the Transport Layer Security (TLS) protocol to determine whether a client has reached its intended server. Given that we've been unable to construct a secured routing system for many decades, the question arises whether there is still the same level of need for such a system that we had some 25 years ago given that the application space sees this as a largely solved problem through the close to ubiquitous use of TLS.

This tension between the Internet layer and the upper layers of the protocol stack is also evident in the way in which we have addressed the perennial issue of location and identity. One of the original simplifications in the IP architecture was to bundle the semantics of identity, location and forwarding into an IP address. While that has proved phenomenally effective in terms of simplicity of applications and simplicity of IP networks, it has posed some serious challenges when considering mobility, routing, protocol transition and network scaling. Each of these aspects of the Internet would benefit considerably if the Internet architecture allowed identity to be distinct from location. Numerous efforts have been directed at this problem over the past decade, particularly in IPv6, but so far, we really haven't arrived at an approach that feels truly comfortable in the context of IP. The problem we appear to have stuck on for the past decade is that if we create a framework of applications that use identity as a rendezvous mechanism and use an IP layer that requires location, then how is the mapping between identity and location distributed in an efficient and suitably robust manner? The transport layer of the protocol stack has also looked at the same space and come up with some interesting approaches as we will see in the next section.

Transport

Back in 1998 the transport layer of the IP architecture consisted of UDP and TCP, and the network use pattern was around 95% TCP and 5% UDP. It's taken all of the intervening twenty-five years but this picture has finally changed.

We've developed some new transport protocols in their period, such as the Datagram Congestion Control Protocol (DCCP) and the Stream Control Transmission Protocol (SCTP) which can be regarding as refinements of TCP to extend a flow control mechanism to apply to datagram streams in the case of DCCP and a shared flow control state over multiple reliable streams in the case of SCTP. However, in a world of transport-aware middleware that has been a constant factor over this period, the level of capability to actually deploy these new protocols in the public Internet is marginal at best. These more recent transport protocols are not recognised by firewalls, NATs and similar, and as a result, the prospects

of widescale deployment of such protocols in the public Internet are not terribly good We seem to be firmly stuck in a world of TCP and UDP.

TCP has proved to be remarkably resilient over the years, but as the network increases in capacity the ability of TCP to continue to deliver ever faster data rates over distances that span the globe is becoming a significant issue. There has been much work to revise the TCP flow control algorithms so that they still share the network fairly with other concurrent TCP sessions yet can ramp up to multi-gigabit per second data transfer rates and sustain that rate over extended periods. The mainstream TCP flow control protocol has been shifting from the conventional RENO-styled protocol to CUBIC. CUBIC attempts to find a stable sending rate and then slowly add flow pressure to the network path to see if greater sending rates can be supported by the network. The response to packet drop remains a dramatic rate drop, but not quite as dramatic as the rate halving of Reno, but nevertheless it is still a drop-sensitive ack-paced flow control protocol.

However, the picture has changed with the introduction of the BBR protocol. Driving the network into the point not only of network queue formation, but right to the point of queue overflow and packet loss is a crude approach. The problem here is that packet loss represents a loss of feedback, and in a feedback-based flow control protocol, this loss of feedback pushes the protocol into a space where it has to pull back its sending rate to re-establish a signal flow. BBR represents a different way of looking at flow control, and BBR attempts to drive the flow to the point of the onset of queue formation in the network rather than aiming at the point of queue collapse. This reduces the latency of the flow and reduces the cost of network switching equipment by reducing the very highspeed fast memory buffer requirements.

This is not the only area of experimentation in changing the TCP congestion control paradigm. Another approach is being explored in the Low Latency Low Loss Scalable throughput initiative (L4S) which is looking at incorporating network signals into the flow control algorithm. Here the network's packet switches write into the packet header when standing queues start to form, using the packet's congestion experienced indication (ECN) signal in the IP header. The receiver of this signal is expected to back off its sending rate in a manner similar to packet loss. The advantage of this approach is that there is no loss of feedback signalling, and the flow reacts to the formation of congestion conditions rather than the end point of queue collapse. However, ECN requires the deployment of ECN-marking equipment, and the effort of synchronising network equipment and transport protocol behaviours is far greater when compared to protocol-only approaches such as BBR.

There have been some other initiatives in the transport space which are also worthy of note.

The first of these is Multipath TCP. The observation here is based around the increasing ubiquity of both WiFi and cellular radio services, and the configuration of most mobile devices to include the ability to access both of these networks. In general, the choice of which network interface to use is a single decision made by the mobile platform for all active applications. When a usable WiFi network is detected, the device will prefer to use that connection for all new connections as it is assumed that the WiFi service will be cheaper for the user and operate at a higher performance level. But if performance is an issue, and resilience is an issue, then can we allow a TCP session to use all the available networks at once, and optimise its use of these multiple network paths to the destination such that the total data throughput is optimised? This is the objective of Multipath TCP, where a single TCP session is broken into a number of sub-sessions, where each sub-session uses a different network path by using a different local network interface. This allows separate TCP states to control the flows passing across each network path to optimise throughput. It also can permit flow migration, allowing a logical TCP flow to switch from one network path to another while preserving integrity. The interesting aspect of this behaviour is that the control of the multipath behaviours is, in the first instance, under the control of the application rather than the host platform. This was an early response to the recognition of increasing capacity and diversity in edge networks, and how we could respond to this situation at the transport session level.

The second initiative, which for me is a fundamental change in transport capabilities and functions, is the introduction of the QUIC protocol. At its simplest level you could say QUIC is a packaging of the

combination TCP and TLS into a UDP wrapping. However, I would suggest that such a description is well short of the mark. QUIC is in many ways a far more ambitious transport protocol, bringing transport to the point where it is better suited to the current application behaviour. QUIC is intended to improve the transport performance for encrypted traffic with faster session setup. QUIC allows for further evolution of transport mechanisms with support for remote procedure calls (RPC). QUIC also has integral support for concurrent session multiplexing that avoids TCP's head of line blocking. QUIC encrypts the payload data, but unlike TLS QUIC also encrypts the control data (the equivalent of the TCP header) and explicitly avoids the emerging TCP ossification within the network by occluding the entirety of the session's control exchange from the network. QUIC is address agile, in that it can react to network-level address renumbering in an active QUIC session, as can occur with the presence of NATs on the network path. QUIC can be implemented in user space, so applications can control their own transport functions. There is no longer a dependence on the platform in terms of the quality of the implementation of the transport service. With QUIC the application exercises a comprehensive level of control of the way the application interacts with the network.

There are a number of lessons to be drawn from the QUIC experience. Any useful public communications medium needs to safeguard the privacy and integrity of the communications that it carries. The time when open protocols represented an acceptable compromise between efficiency, speed and privacy are over and these days all network transactions in the public Internet need to be protected by adequate encryption. QUIC's model of wrapping a set of transactions including both data and control transactions between a client and a server into an end-to-end encryption state represents a minimum level of functionality in today's networking environment.

Secondly, QUIC provides needed additional transport functionality. TCP and UDP represent just two points of transport functions within a broader spectrum of possible transport models. UDP is just too susceptible to abuse, so we've heaped everything onto TCP. The issue is TCP was designed as an efficient single streaming protocol and retro-fitting multiple sessions, short transactions, remote procedure calls, reliable single packet transactions, and shared congestion states have proved to be impossible to implement in TCP.

Applications are now dominant in the Internet ecosystem, while platforms and networks are being commoditised. We are seeing losing patience with platforms providing common transport services for the application that they host, and a new model where the application comes with its own transport service. Taking an even broader perspective, the context of the Internet's success lay in shifting the responsibility for providing service from the network to the end system. This allowed us to make more efficient use of the common network substrate and push the cost of this packetization of network transactions over to end systems. It shifted the innovation role from the large and lumbering telco operators into the more nimble world of software. QUIC takes this one step further, and pushes the innovation role from platforms to applications, just at the time when platforms are declining in relative importance within the ecosystem. From such a perspective the emergence of an application-centric transport model that provides faster services, a larger repertoire of transport models and encompassing comprehensive encryption was an inevitable development.

We have pushed the responsibility for end-to-end authentication into the transport layer with the close to ubiquitous TLS. TLS layers itself above TCP (or merges with the TCP-like function in the case of QUIC) and the client passes the name of the service it intends to connect with to the remote server. The server passes its public key to the client, and the client authenticates this key using its own trust anchors. The server and client then negotiate a session key and proceed with an encrypted session. TLS is robust in almost every respect. The major weakness lies in the highly distributed trust model where there are hundreds of different operators of trusted credentials (certification authorities) and thousands of various registration agents. These entities are placed in a highly trusted role, and they can never lie. The problem is that they have proved to be corruptible from time to time. They typically operate using online services and a successful attack against such platforms can be abused to allow the issuance of trusted public certificates. We have invested considerable time and effort in shoring up this trust framework, but at the same time we've been working to make these public key certificates a commodity rather than an expensive

luxury. The introduction of free certification authorities has succeeded in making these certificates available to all, but at the same time the totally automated certificate issuance process is liable to various forms of abuse. Despite these considerations we've placed the entirety of the burden of service authenticity and session encryption onto TLS, to the point that other related efforts, such as IPSEC, BGP routing security and DNSSEC in the DNS, are generally perceived as optional extras rather than basic essentials to be included the security toolkit.

Applications and Services

This layer has also seen quite profound changes over the past quarter century, tracking the progress of increasing technical capability and increasing consumer demands. In the late 1990's the Internet was on the cusp of portal mania, where Look Smart was the darling of the Internet boom and everyone was trying to promote their own favourite "one stop shop" for all your Internet needs.

By 1998 the AltaVista search engine had made its debut, and these content collation portals were already becoming passe. This change, from compiling directories and lists to active search, completely changed the Internet. These days we simply assume that we can type any query we like into a search engine and the search machinery will deliver a set of pointers to relevant documents. And every time this occurs our expectations about the quality and utility of search engines are reinforced. Content is also changing as a result, as users no longer remain on a single site and navigate across the pages that are found at that site. Instead, users are driving search engines, and pulling the relevant pages without reference to any other material. But it's not stopped there. Search engines are morphing into "instant answer machines" where instead of providing a set of pointers to sources where there is a high level of correlation between the source and the question, the search engine attempts to extract material from the source and show what it believes is the answer to the implicit question in the search term. Even this is just a waypoint in a longer journey, and today we are seeing AI chat bots appearing, where the underlying data set that has been indexed by the search machinery is now being used as a corpus of data to drive an AI chat engine. The interaction is based a natural language model.

If you thought of the Internet as an information resource, then the use of AI in this manner is a disturbing step. In this AI model the responding system generates plausible, but very definitely not necessarily factual, natural language responses to the implicit question in the query. It's challenging to see this path from indexing data sources and matching query terms to the terms used primary sources to one of a natural language generator that produces textual responses that is not grounded in facts nor necessarily derived from primary sources as being progress! Despite such misgivings about the deliberate abasement of the quality of the Internet as an information resource, this shift does fit into a larger picture of the Internet's transformation to mass entertainment vehicle, which is much of the driving force in today's content world.

A related area of profound change has been the rise of social media. The television, radio, film and print industries had evolved to use content mediators, compilers and editors to curate their content, and the widespread deployment of highly capable user devices allowed end users to directly perform content production without the need to engage with mediators or producers. This has transformed many societies, and the social media platforms, including YouTube, Flickr, Facebook, Instagram and Tiktok, have been rocketed into societal prominence, prompting major debates about the role of these platforms and levels of societal influence that such platforms can generate.

Underlying these changes is another significant development, namely the change in the content economy. In 1998 content providers and ISPs were eyeing each other off in a fight for user revenue. Content providers were unable to make pay per view and other forms of direct financial relationship with users work in their favour and were arguing that ISPs should fund content. After all, they argued, the only reason why users paid for Internet access was because of the perceived value of the content that they found on the Internet. ISPs, on the other hand, took up the stance that content providers were enjoying a free ride across the ISP-funded infrastructure, and content providers should contribute to network costs. The model that has gained ascendency as a result of this unresolved tension is that of advertised-

funded content services, and this model has been capable of sustaining a vastly richer, larger, and more compelling content environment.

However, this comes at a price, and in this case the price lies in the motivations of the platforms that perform ad delivery. The critical objective now is to engage the user for longer periods, so that more ads can be presented, and more information about the user's profile can be gleaned. Merely informing a user is largely a transactional interaction, whereas entertaining a user can be far more lucrative in terms of generating advertising revenue because of the longer attention span. This has been highly successful for some content players, particular the current giants of streaming content, and its therefore unsurprising that the largest entities in the content world, such as Alphabet, Microsoft and Amazon and Apple are far more valuable in terms of market capitalization than their counterparts in the carriage world. We are now seeing the next round of the friction between content and carriage where the access network operators are arguing that the content players should contribute to the costs of access carriage.

The Domain Name System (DNS) also merits a mention in this section. From one perspective little has changed in this space, and the DNS name resolution protocol hasn't changed to any appreciable extent. In some sense that's true, but at the same time there have been some significant changes.

The first of these changes is in the adoption of DNSSEC, a framework that allows DNS clients to validate the answers that they receive from the DNS. The DNS has always been a point of security vulnerability on the Internet in that it has always been prone to various forms of attack where false answers are substituted in place of the genuine answer. DNSSEC provides a digital signature record to each normal record, and also implements an interlocked chain of signatures to link to the key associated with the root zone. A client may request the signature record to be provided with the normal response, and then make further requests to construct the validation chain all the way to the root zone. Successful validation assures a client that the data provided in the original response is authentic and current. The root zone of the DNS was first signed in 2010, but adoption of DNSSEC has been slow. While the addition of such a validation mechanism is undoubtedly a step forward in protecting users against various forms of name-based interference, the cost is increased fragility of the DNS and increased resolution times. One underlying issue is that the addition of digital signatures to a DNS response is highly likely to push the DNS into sending large responses, and large responses over a UDP-based transport is prone to fragmentation-based unreliability, and the switch to use TCP also takes time. What this has implied is that the path to adoption of DNSSEC has been slow, despite the obvious protections it can provide regarding potential tampering with the DNS.

The second major theme of change in the DNS concerns the larger issue of pervasive monitoring in the DNS, highlighted by the Snowden revelations of 2013. Most Internet transactions start with a call to the DNS, and the meta-data contained in DNS queries and responses provides a rich real-time profile of user activity, both in general and potentially on a user-by-user basis. This has prompted a concerted effort to improve the privacy aspects of the DNS as a protocol. One approach has been to take the existing use of DNS across a TCP session and add Transport Layer Security (TLS) to the TCP session, so the contents of the interaction between the client and the DNS server are impervious to third party inspection or manipulation. This can be taken a step further with DNS over HTTPS/2 where the DNS payload has a lightweight HTTP wrapper in addition to TLS. This allows DNS traffic to be melded in with all other HTTP traffic as a further step of obscuring DNS transactions. More recently we've seen DNS over QUIC, using QUIC's faster session start times and fast open capabilities to improve the performance of the arrangement and DNS over HTTPS/3 which combines QUIC with HTTP object semantics. The primary focus of this work has been the part of the DNS where the client's stub resolver interacts with a recursive resolver, as this scenario identifies the client. The useful property of this part of the DNS is that the same client / server setup is used repeatedly, so either a long-held secure transport session, or a fast-reopen session can amortise the high cost of set up of a reliable secure session over many subsequent queries, making the overall cost of such secure transport arrangement more palatable.

Such measures still have some security issues, as the recursive resolver is privy to both the client's identity and the DNS queries that they are making. Recent work has been undertaken on an "oblivious" model

of DNS operation, where the recursive resolver function is split in two and two layers of encryption of used. The client talks to the first party, a DNS relay over an encrypted session, and passes it a query which has been encrypted using the public key of the second party, the recursive resolver. The relay resends the encrypted DNS query to the recursive resolver to resolve. The first party knows the identity of the client, but not the DNS query that is being made. The second party knows the DNS query, but not the identity of the client.

This work on DNS privacy has extended into the scenarios of the recursive resolver taking with authoritative name servers, although it's unclear as to the extent of the security benefits (as the end user is not identified directly in such queries), nor is session reuse as feasible in this scenario.

In many ways applications and services have been the high frontier of innovation in the Internet in this period. An entire revolution in open interconnection of content elements has taken place and content is now a very malleable concept. It's no longer the case of "my computer, my applications, my workspace" or "your server, your content" but an emerging model where not only the workspace for each user is held in the network, but where the applications and services themselves are part of the network, and all are accessed through a generic mechanism based around permutations of the HTTPS access model. This is the world of so-called "cloud services". The world of cloud services takes advantage of abundance in computation, storage, and communications resources and rather than a network facilitating users to connect to service delivery points, the cloud model inverts the model and attempts to bring replicant copies of content and services closer to the user. If distance equates to cost and performance in the networking world, then the cloud model dramatically shortens the distance between consumer and content, with obvious implications in terms of cost and performance reductions. The clouded Internet is capable of achieving extremely challenging performance and cost objectives by changing the provisioning model of the content and service from "just in time" on-demand service to "just in case" pre-provisioning of local caches so that the local cache is ready if the service is accessed by a local client.

Cyber Hostility

We still are under relentless attack at all levels. We are beset by data leaks, surveillance, profiling, disruption, and extortion.

Attacks are now commonplace. Many of them are brutally simple, relying on a tragically large pool of potential zombie devices that are readily subverted and co-opted to assist in attacks. The attacks are often simple, such as UDP reflection attacks where a single UDP query generates a large response. The source address of the query is forged to be the address of the intended attack victim, and not much more needs to be done. A small query stream can result in a massive attack. UDP protocols such as SNMP, the Network Time Protocol (NTP), the DNS, and memcached have been used in the past and doubtless will be used again.

Why can't we fix this problem? We've been trying for decades, and we just can't seem to get ahead of the attacks. Advice to network operators to prevent the leakage of packets with forged source addresses was published nearly two decades ago, in 2000. Yet massive UDP based attacks with forged source addresses still persist today. Aged computer systems with known vulnerabilities continue to be connected to the Internet and are readily transformed into attack bots.

The picture of attacks is also becoming more ominous. Although we previously attributed these hostile attacks to "hackers," we quickly realised that a significant component of them had criminal motivations. The progression from criminal actors to state-based actors is also entirely predictable, and we are seeing an escalation of this cyber warfare arena with the investment in various forms of vulnerability exploitation that are considered desirable national capabilities.

It appears that a major problem here is that collectively we are unwilling to make any substantial investment in effective defence or deterrence. The systems that we use on the Internet are overly trusting to the point of irrational credulity. For example, the public key certification system used to secure web-

based transactions is repeatedly demonstrated to be entirety untrustworthy, yet that's all we trust. Personal data is continually breached and leaked, yet all we seem to want to do is increase the number and complexity of regulations rather than actually use better tools that would effectively protect users.

The larger picture of hostile attack is not getting any better. Indeed, it's getting very much worse. If any enterprise has a business need to maintain a service that is always available for use, then any form of in-house provisioning is just not enough to withstand attack. These days only a handful of platforms can offer resilient services, and even then, it's unclear whether they could withstand the most extreme of attacks.

A constant background level of scanning and probing goes on in the network, and any form of visible vulnerability is ruthlessly exploited. One could describe today's Internet as a toxic wasteland, punctuated with the occasional heavily defended citadel. Those who can afford to locate their services within these citadels enjoy some level of respite from this constant profile of hostile attack, while all others are forced to try to conceal themselves from the worst of this toxic environment, while at the same time aware that they will be completely overwhelmed by any large-scale attack. It is a sobering thought that about one-half of the world's population are now part of this digital environment. A more sobering thought is that many of today's control systems, such as power generation and distribution, water distribution, and road-traffic-control systems are exposed to the Internet.

What makes this scenario even more depressing is the portent of the so-called Internet of Things (IoT). In those circles where Internet prognostications abound and policy makers flock to hear grand visions of the future, we often hear about the boundless future represented by this Internet of Things. This phrase encompasses some decades of the computing industry's transition from computers as esoteric pieces of engineering affordable only by nations to mainframes, desktops, laptops, handheld devices, and now wrist computers.

Where next? In the vision of the IoT we are going to expand the Internet beyond people and press on using billions of these chattering devices in every aspect of our world. What do we know about the "things" that are already connected to the Internet? Some of them are not very good. In fact, some of them are just plain stupid. And this stupidity is toxic, in that their sometime-inadequate models of operation and security affect others in potentially malicious ways.

If such devices were constantly inspected and managed, we might see evidence of aberrant behaviour and correct it. But these devices are unmanaged and all but invisible. Examples include the controller for a web camera, the so-called "smart" thing in a smart television, or the controls for anything from a washing machine to a goods locomotive. Nobody is looking after these devices. When we think of an IoT we think of a world of weather stations, webcams, "smart" cars, personal fitness monitors, and similar things.

But what we tend to forget is that all of these devices are built on layers of other people's software that is assembled into a product at the cheapest possible price point. It may be disconcerting to realise that the web camera you just installed has a security model that can be summarised with the phrase: "no security at all," and it's actually offering a view of your house to the entire Internet. It may be slightly more disconcerting to realise that your electronic wallet is on a device that is using a massive compilation of open-source software of largely unknown origin, with a security model that is not completely understood, but appears to be susceptible to be coerced into being a "yes, take all you want" device. It would be nice to think that we've stopped making mistakes in code, and from now on our software in our things will be perfect. But that's hopelessly idealistic. It's just not going to happen. Software will not be perfect. It will continue to have vulnerabilities.

It would be nice to think that this Internet of Things is shaping up as a market where quality matters, and consumers will select a more expensive product even though its functional behaviour is identical to a cheaper product that has not been robustly tested for basic security flaws. But that too is hopelessly naive.

The Internet of Things will continue to be a marketplace where the compromises between price and quality will continue to push us on to the side of cheap rather than secure. What's going to stop us from further polluting our environment with a huge and diverse collection of programmed unmanaged devices with inbuilt vulnerabilities that will be all too readily exploited? What can we do to make this world of these stupid cheap toxic things less stupid and less toxic? So far, we have not found workable answers to this question.

Our ability to effectively defend the network and its connected hosts continues to be, on the whole, ineffectual. Anyone who still has trust in the integrity of the systems that make up the digital world is just hopelessly naive. This is a toxic and hostile space, and we still have no idea how we can shift this to a different state that can resist such erosive and insidious attacks. But somehow, we are evidently not deterred by all this. Somehow each of us have found a way to make the Internet work for each of us.

The Business of the Internet

As much as the application environment of the Internet has been on a wild ride over the past 25 years, the business environment has also had its tickets on the same roller coaster ride, and the list of business winners and losers include some of the historical giants of the telephone world as well as the Internet-bred new wave of entrants.

In 1998, despite the growing momentum of public awareness, the Internet was still largely a curiosity. It was an environment inhabited by geeks, game players and academics, whose rites of initiation were quite arcane. As a part of the data networking sector, the Internet was just one further activity among many, and the level of attention from the mainstream telco sector was still relatively small. Most Internet users were customers of independent ISPs and the business relationship between the ISP sector and the telco was tense and acrimonious. The ISPs were seen as opportunistic leeches on the telco industry; they ordered large banks of phone lines, but never made any calls; their customers did not hang up after 3 minutes, but kept their calls open for hours or even days at a time, and they kept on ordering ever larger inventories of transmission capacity, yet had business plans that made scribblings on the back of an envelope look professional by comparison. The telco was unwilling to make large long term capital investments in additional communications infrastructure to pander to the extravagant demands of a wildcat set of internet speculators and their fellow travellers. The telco, on the other hand was slow, expensive, inconsistent, ill-informed and hostile to the ISP business. The telco wanted financial settlements and bit level accounting while the ISP industry appeared to manage quite well with a far simpler system of peering and tiering that avoided putting a value on individual packets or flows.

This was never a relationship that was going to last, and it resolved itself in ways that in retrospect were quite predictable. From the telco perspective it quickly became apparent that the only reason why the telco was being pushed to install additional network capacity at ever increasing rates was the ISP sector. From the ISP perspective the only way to grow at a rate that matched customer demand was to become one's own carrier and to take over infrastructure investment. And, in various ways, both outcomes occurred. Telcos bought up ISPs, and ISPs became infrastructure carriers.

All this activity generated considerable investor interest, and the rapid value escalation of the ISP industry and then the entire Internet sector generated the levels of wild-eyed optimism that are only associated with an exceptional boom. By 2000 almost anything associated with the Internet, whether it was a simple portal, a new browser development, a search engine, or an ISP, attracted investor attention, and the valuations of internet startups achieved dizzying heights. Of course, one of the basic lessons of economic history is that every boom has an ensuing bust, and in 2001 the Internet collapse happened. The bust was as inevitable and as brutal as the preceding boom was euphoric. But, like the railway boom and bust of the 1840's, once the wreckage was cleared away, what remained was a viable, and indeed a valuable, industry.

By 2003 the era of the independent retail ISP was effectively over. But it reshaped itself dramatically with the introduction of mobile services. It was the old telco sector that had secured spectrum allocations in the bidding ways for the early 2000's and while they had thought that mobile voice would be the reason why these investments would make sense, it was mobile internet services that proved to be the lasting service model. In this period the Internet was the amalgam of the largest of the original ISP, the transformed cable television operators and the mobile providers. Each national regime was populated with some 3 to 5 major service providers and the business started to stabilise around this model.

Into this world came the content world of the Internet, using cloud-based models of service delivery to circumvent communications bottlenecks in the long-haul transit sector. They embarked on service models which included advertiser-funded models of content generation and delivery and direct subscription models, and the result has been so effective that the value of this sector is far greater than the traditional ISP and carriage sector. The content world is now the major funder of subsea cable systems, and the carriage world has been very reluctantly pushed into an undistinguished commodity role as a result.

This is reflective of a broader process of technology permeation. The telephone world used the network as the major focus of technology and investment. The edge devices, telephone handsets, were simple cheap devices, whereas network switches and transmission elements were built to exacting and expensive standards. As we attached computers to the edges of the network, these devices were able to tolerate a broader spectrum of network behaviours and had a lower base level expectation of behaviour. Consequently, value has moved out from the core of the network to its edges.

But this process has also been reflected within these edge devices. We started with a model of a highly capable and complicated operating system platform, and relatively simple applications that made use of platform services. Some 25 years ago the release of Windows 98 was a big thing, and rightly so. As these edge devices become more capable, have higher processing capability, and more local storage applications have elected to take on more of the responsibility in terms of the user's experience. In doing so they are no longer reliant on the release schedules of the platform provider, no longer as concerned about the level of control being exercised by this platform provider and gaining an essential level of self-control. Modern browsers (well Chrome, and a few far smaller follow travellers) are far more complex than most operating systems, and they continue to subsume functions and roles previously carried out by the platform. With DNS over HTTPS the task of DNS name resolution can be transformed to an application function, rather than a common platform function. With QUIC, the transport protocol itself has been subsumed into the application space.

Not only have we seen the commoditisation of the network over the past twenty-five years, we've seen similar commoditisation pressures on the end device platforms and on the operating systems used on these devices. Even the browser space has been commoditised. The brunt of competitive differentiation in this industry has been pushed up the protocol stack into the content and service economy, and there is the distinct feeling that even in that space competitive differentiation perhaps a misnomer, and what we have is a synthetic form of competition between a select small group of digital service delivery behemoths that in any other time and context would probably be called a cartel.

What Now?

It's been a revolutionary quarter-century for us all, and the Internet has directly or indirectly touched the lives of almost every person on this planet. Current estimates put the number of regular Internet users at one half of the world's population.

Over this period some of our expectations were achieved and then surpassed with apparent ease, while others remained elusive. And some things occurred that were entirely unanticipated. At the same time very little of the Internet we have today was confidently predicted in 1998, while many of the problems we saw in 1998 remain problems today.

This is a work-in-progress where the next quarter-century will probably see the same level of intensity of yet more confronting structural changes to the global communications sector. And that is a somewhat scary prospect given the collection of other challenges that are confronting us all in the coming decades. At the same time, I think it would be good to believe that the Internet's debut in our world has completely rewritten what it means to communicate, rewritten the way in which we can share our experience and knowledge, and, hopefully, rewritten the ways in which we can improve the ways we can work together on these other existential challenges for humanity and our planet.

Disclaimer

The above views do not necessarily represent the views or positions of the Asia Pacific Network Information Centre.

Author

Geoff Huston AM, B.Sc., M.Sc., is the Chief Scientist at APNIC, the Regional Internet Registry serving the Asia Pacific region.

www.potaroo.net