

# The IP Scorecard

Geoff Huston  
June 2001

The initial research work that underpins the architecture of the Internet commenced in the 1960's, and the basic specification of the protocols used by the Internet were completed by the mid-1970's. That's almost 30 years ago, and much has changed in that time. In the 1970's computers were large pieces of ironware that occupied entire rooms and supported hundreds of simultaneous users. A minicomputer was a hefty piece of metal that occupied a most of an equipment rack. The number of computers in the world numbered in the thousands, and they were the preserve of an exclusive cadre of technocrats. Computers moved only in earthquakes, and the concept of ubiquitous mobile computing was a distant dream. Data circuits were modified voice channels, and data speeds of 56Kbps were considered fast. The contrast of this environment to that of today is truly striking. What is surprising is that a communications protocol developed in that period was defined with sufficient generality and extensibility that it now one of the foundational protocols of the global communications industry. IP has scaled in almost every metric by a factor of tens of millions. IP circuits now operate at speeds of up to 10 billion bits per second, and the network spans hundreds of millions of users and connects a similar number of end systems. And the core protocol, IP, remains essentially unaltered. By any form of reckoning that's an impressive achievement.

Of course not everything in the original IP protocol specification has proved to be useful in today's environment. Its interesting to review the architecture of the Internet and see what has worked well and what is still proving to be a challenge today. So lets undertake a quick scorecard of IP.

## What has Worked

The basic architecture of IP is the **end-to-end model**. Documented originally as a research paper only in 1984, the "End to End Arguments in Systems Design" summarizes the basic strength of the IP protocol. The basic proposition is that end-to-end data transfer functions can only be performed correctly by the end systems themselves, and not by the intermediary networks. Any network, however carefully designed, will be subject to failures of various forms. Rather than have the network maintain the state of each active transaction, and have the responsibility to recover in the event of failure, the end-to-end architecture passes the responsibility for integrity of communication to the end system. The result is that the IP protocol assumes that the network has some degree of unreliability, and it is left to the end systems to identify and repair any errors that occur within the transmission. The outcome of this architecture is that IP makes minimal assumptions about the precise nature of the network, and this has allowed IP to be used on a broad range of underlying network types, and to support communications across a concatenation of a set of quite different network types. While it may be less common today, it was certainly quite reasonable in the late 1980's to see an end-to-end IP session pass across a sequence of Ethernets, FDDI rings, X.25 networks, and digital trunk circuits.

Splitting the protocol into two end-to-end transport services, **UDP and TCP**, was also a powerful design decision. Applications that required integrity of communication were able to use TCP as the transport protocol. The additional overheads of the initial handshake, and the maintenance of a shared state between sender and receiver support a transport service that allows integrity of communication. If the transaction is a short query response, or requires some form of external clocking of the signal, as in real time media streaming, UDP provides a more basic service.

Without the initial handshake and the maintenance of a shared state, UDP has minimal overhead. More fundamentally, this split of transport protocols ensured that IP was not a single service network. The same network can be used to support various forms of real time voice, short query applications (such as the DNS) and extended high volume efficient data transfer simultaneously, without making any particular specialized demands on the underlying network.

The **32 bit address** headers used in the IP protocol headers was an interesting decision at the time. Considering the computing environment of the mid-70's, using a single address architecture capable of supporting as many computers as the world's population can only be described as an act of inspired faith in a vision of ubiquitous computing. Other protocols of the time commonly used 8 bit addresses, capable of supporting networks of 256 computers. By comparison the 32-bit address space was simply massive.

The operational decision to set up a **single global registry of address allocations**, so that each IP network could use a unique fragment of this address space was also a significant factor in the success of IP. When two IP networks interconnected there was no need to renumber all the computers in either network, nor was there the need to set up a series of application gateways to mediate between the two networks.

**The decoupling of routing protocols from the basic protocol specification** was also a far-reaching decision. The scaling of the Internet from tens of computers to hundreds of millions have required the deployment of a number of generations of routing protocols. However, the basic forwarding mechanism of IP, stateless hop-by-hop destination-based forwarding, has remained constant. The architectural aspect of IP which is at work here is that of modularization, where various components of the protocol are decoupled from each other. This allows individual components of the network protocol suite to be refined without having to go through a massive exercise of upgrading the protocol stack in every single computer.

The modular approach was also used in the TCP protocol suite. TCP uses an **adaptive feedback mechanism** to control the speed of the data transfer. The sender uses the feedback from the receiver to assess the current characteristics of the network, including the current end-to-end delay and the packet loss rate. The mechanisms by which TCP senses network congestion, and the way the sender reacts to such conditions have been refined over the years. The refinement still allows an original TCP implementation to interoperate with the most recent, while allowing the more recent implementations to make more efficient use of the network.

And last, but not least, in this scorecard, is **the open specification of the IP protocol suite**. Not only were the protocol specifications openly available to all, but also reference implementations were also openly available. Indeed, the process of protocol specification was one that was conducted in an inclusive and open fashion, an approach that continues with the Internet Engineering Task Force today. The other aspect of this open specification was an emphasis on functional interoperability. The IETF creed of "rough consensus and running code" has been a powerful means of ensuring very widespread adoption of IP technology. In short, it works.

## The Challenges

Before we all pat ourselves on the back and run to the bar for a refreshing ale to celebrate a job well done, it is necessary to point out that there still remain a number of very significant challenges in IP. Lets see what we can place on the other side of the scorecard.

As the Internet grows **routing** and the related area of traffic engineering continue to present challenges. Each order of magnitude of growth of the Internet has implied a need to refine the routing protocols to scale to the new dimensions of connectivity and policy. In addition, the hop-by-hop forwarding paradigm tends to aggregate traffic on major trunk routes, and this

aggregation of capacity demand may be falling foul of the clear channel carriage capacity of optical systems. It appears prudent to consider the definition of traffic engineering protocols within the Internet that manage to distribute traffic loads across a set of alternative network paths.

**Mobility** remains a real challenge for IP. With the advent of widespread adoption of the combination of personal digital devices and various forms of wireless connectivity, the mobile communications environment wants to break free of plain old voice and embrace the broader capability of the Internet. The numbers of mobile devices are set to dwarf the current numbers of conventional IP systems, and the efforts to add seamless mobility to the IP protocol suite remain challenging. The basic issue is that the IP protocol combines the notion of identity and location into the single IP address, while mobility requires a decoupling of these two concepts.

Much of the environment of the Internet relies on a **distributed trust model**. There are vulnerabilities in the protocol suite that are a result of this distributed trust environment, as trust without explicit authentication is always a risky proposition in a public communications environment. The ability to forge email headers and distribute vast amounts of unsolicited mail is but the tip of the distributed trust iceberg. Much remains in the effort to add explicit authentication as a precursor to trust.

One could generalize this further and point out that **identity** is a weak concept in the IP protocol. If we want to support mobility as well as various security models, introducing some form of location-independent identity into the IP protocol model may well be an effective direction.

**Multicast** remains elusive. The potential use for various forms of collaborative applications that share a common communications state is significant. Multicast offers a mechanism of collaboration where there is no synchronizing master server, but instead a collection of end systems, which share a single state. While such systems offer greater resiliency and scaling properties, the operational support structures for multicast within the network remain a significant barrier to widespread deployment.

While we are on elusive topics, don't forget **quality of service**. The ability to offer different levels of network response to different classes of applications or different classes of clients remains an area, which is replete with potential technology solutions and yet still conspicuous by its absence in the public Internet. The issue appears to be that the solutions all focus on different parts of a broader, and as yet undefined, service architecture. It appears that there are still missing pieces in this broader architecture, and more impetus is required to understand how to identify and fill in these architectural gaps.

**Wireless** is also a challenge for IP. The IP protocol specification, and the TCP protocol in particular, make some inherent assumptions about the network, particularly relating to the stability of the round trip timers and the loss characteristics. Wireless can alter these properties, and can force TCP to be very conservative about how much data can be passed through the network. If the promise of 3G high-speed wireless services is to be achieved we will need to examine how to further refine TCP to operate efficiently in this environment.

While an address space that spans 4 billion potential systems may have appears to be a large number, the inexorable onslaught of Internet ubiquity is challenging this limit. One approach appears to be to deploy **IPv6**, which uses 128-bit addresses. While this does provide an elegant answer to the limitations of the current version 4 protocol, widespread deployment of IPv6 in the Internet is still awaiting some form of a kick-start. The widespread use of various forms of **network address translation** measures as a network boundary technology indicates that the current Internet has already adopted a different interim solution to this problem. Of course such interim measures are not without their limitations, but for many, this solution is considered to be good enough for their purposes. It may well be that the definition of an functional technology

solution is, in itself, no longer enough for adoption within the Internet, and the considerations of commercial deployment will have to be factored in to refinements of IP.

So I'm afraid that the bar will have to wait for a little while yet. This side of the IP scorecard indicates that there is still much work to be done. Perhaps it will always be the case, and perhaps that's a good thing. To quote Harald Alvestrand, the chair of the Internet Engineering Task Force, "*If you're not moving, you're dead!*"

---