Observations on the Beginnings of the Internet*

David Walden

My involvement

I believe I was the first Internet programmer: I began code design for the first Internet packet-switch in 1968. In the following years, I participated in creation of various now well-known network protocols.

Of course, these days, when I tell people I was involved at the beginning of the Internet, they usually say, "just like Al Gore," and then laugh. However, once they understand I really was there at the beginning of the Internet, they often ask, "did you know then what it would become?" In 1968-69, I didn't know (and I don't think anyone else knew) exactly what it would become. However, I did know what we were doing was a big, important change, especially since the phone company was so disparaging of it. But, by about 1972, with the coming of networked email, I did have an image of what Internet would become — we began talking about, "every person and every process being linked to every other and an Internet interface in every toaster and doorknob." My great surprise has been that we didn't get to where we are today twenty years ago.

Seeking a common user network

An often heard story about the beginning of the Internet is that it was to provide a communications system that would continue to work in the face of a nuclear attack. This myth is untrue. The Information Processing Techniques Office (IPTO) of the Advanced Research Projects Agency (ARPA) of the Department of Defense was responsible for initially creating the Internet; and, in those early days, IPTO played the role of U.S. "ministry of technology" with regard to computing and communications (curiously, U.S. political ideology doesn't actually permit us to have a ministry or national policy for something

*Presented at a conference of people interested in micro computers, etc., at Asilomar State Park, California, April 24, 2002.

as important as technology). IPTO was directing much R&D funding at making computers more interactive in a variety of ways. IPTO was funding a couple of dozen research institutions such as MIT, UC Berkeley, Carnegie Mellon, RAND Corporation, Stanford Research Institute, the University of Utah, and so forth, and saw an opportunity to simultaneously push the state of the art and perhaps avoid having to continue to buy new mainframe computers for each of its research contractors.

In 1968, each computer vendor had one or more proprietary operating systems, there was no uniformity of byte size or word length, at least two major character sets were in common use, and communication networks were one-off special purpose terminal concentration systems or special purpose message switching systems, with each system dedicated to one application such as a specific airline reservation system or a specific logistics system. In particular, the message switching systems typically did not send their message traffic in real time, and the functions of the particular application were tightly intertwined with the communications functions rather than layered on top of a general purpose communications system. In fact, the circa 1968 systems often couldn't send arbitrary binary data — they could only send types of data the data forms built into the application were expecting.

Thus, ARPA began to plan a "resource sharing computer network" so that many types of computers, users and applications could share the same network and communicate with each other as desired.

Inventing packet switching

In 1968, after a year or more of planning, IPTO released a Request for Quotation, seeking bidders to develop a packet-switching backbone network. The small team at Bolt Beranek and Newman Inc. (BBN), of which I was a member, bid on and won the contract to develop the packet-switches for what was then called the ARPANET. (In my view, packet switching is an even more important innovation than the Internet. The Internet was built on packet switching; but the Internet is only one of many packet switching applications.) The packet switching system we developed had a set of capabilities well beyond the prior state of the networking art (which may seem surprising since many of the characteristics I am about to mention are so commonplace today). For instance:

- the packet-switching system was able to convey arbitrary binary data between users
- it interleaved packets from different user transmissions in a way that resulted in low cross-network latencies and high circuit utilization
- using 50 kilobit per second circuits (highly unusual in those days of 110 bit per second terminal connections to computers), it could send packets across the country in milliseconds (fast enough for real time character echoing)

- it used dynamic routing (versus static routing set up at sysgen time as was typical for networks then) which enabled automatic adaptation to line and switch ups and down, including addition to the network of new switches and lines — this was probably the first instance of a major distributed, parallel computation
- more generally, it was designed for unmanned decentralized operation — no real-time control from a central location
- it included real-time reporting of network operation and performance to a central monitoring center and provision for cross network release of new versions of the system (the difficult trick was not designing the new release software so it could propagate itself across the network — the difficult trick was making it possible to revert to the old release if the new release didn't work right)
- it had very high performance (in my experience it is usually not possible to tune up inefficient code after the fact; rather, it has to be written from scratch to have high performance)

One aspect of this initial packet-switching system that we soon realized was a mistake was having a network address field of 8 bits — allowing 64 packet switches with four user computers on each switch.

We developed this first packet switching system in 9 months. In the years since, I have been asked how was it possible to design, develop and deploy such an innovative system so fast. I think there are four reasons for our success:

- we had a small team of smart, motivated people — always a good approach
- we were working on a cost-plus-fixed-fee contract — thus, we could concentrate on doing the job at hand rather than being distracted by issues of financial risk
- we only had 12 kilobytes of memory for the program — thus, we could not really create a very big program
- because we were first, we had no backward compatibility issues — if one wants to create a new router today, I suppose one must develop a million lines of code (or perhaps several million)

Inventing protocol stacks via RFCs and the NWG

In parallel with development of the packet-switching backbone, one or two representatives of each designated user site (UCLA, SRI, UCSB, University of Utah, etc.) began to meet together to figure out how they would use the network. The user sites were all academic or (essentially) non-profit. The user site representatives were all graduate students or young engineers. Thus, the emphasis was on what made sense technically rather than on the agendas of for-profit companies or national authorities as often happens in traditional standards efforts. The group's goals were: intercommunication of processes on many different computers and operating systems, and remote access to applications by any network user. This resulted in: an open, layered design for process-to-process computing, and what we today call client-server computing.

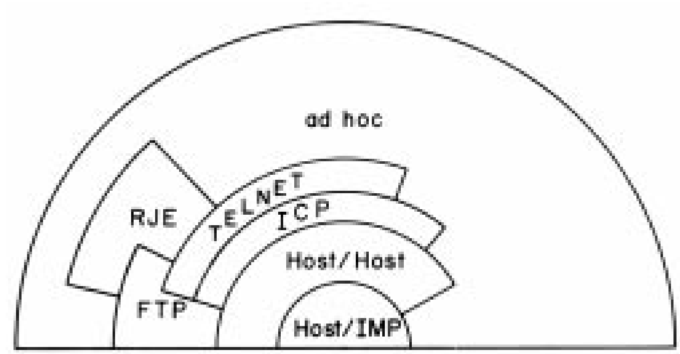

Figure 1: From the paper entitled “Host-to-host Protocols,” D.C. Walden, *Network Systems and Software*, Infotech State of the Art Report 24, page 298, Infotech Information Limited, Nicholson House, Maidenhead, Berkshire, England, 1975

Each element in the layered architecture was meant to be accessible and changeable independent of the other layers. Looking at Figure 1 — one of the first published sketches of protocol layering — NCP is the initial version of what has been replaced by TCP, FTP supports transfer of files of arbitrary data from one machine to another, Telnet allows a user at one location to act as if he or she is a user of an application at another site (and it defined the network virtual terminal and negotiation of a mutually compatible set of terminal characteristics). Other protocols or applications could be layered on any part of the existing layering.

ARPA apparently had no specific plan for the activities of the representatives of the user sites other than letting smart people talk to work something out. There was no membership; anyone who wanted to participate was welcome. They communicated with each other in terms of “Requests for Comments” rather than “draft specifications of standards”; and they called themselves the Network Working Group (NWG). All this was indicative of a common quest for technological coherence and compatibility rather than efforts to push company agendas. NWG participants engaged in collegial engineering discussion, and nothing was locked in concrete. Dave Clark of MIT has characterized the NWG culture as one of “strong consensus and running code,” meaning that Internet innovations resulted from someone caring enough about something and how to implement it to get something actually running that would excite use by a lot more people — stand-alone theories, unending standards debates, and official decrees were uninteresting. In time, the NWG evolved into the IETF and Internet Society, the latter of which was started by Vint Cerf when he became afraid that some institution with actual authority would take over control of the Internet.

Expanding the Internet

One of the biggest decisions made in the early days of the Internet was to avoid the then standard practice of charging for use by time and distance. Rather, the decision was explicitly made to charge a fixed price per year for a given size access pipe. In this way, it was felt, the cost would get buried in an institution's overhead and forgotten, and would not inhibit people from using the Internet and Internet use would grow. I still think it's the best way, because people want to know their costs up front, and then want to forget cost and just use the capability.

The first big Internet application, of course, was networked email, and that led to ARPA explicitly encouraging participation in network discussion groups. But, there were also many other "apps," often making up logical subnetworks of users. From 1972, experience was sought communicating among various networks and kinds of networks (i.e., internetting), and by 1974 TCP/IP had been specified, based on ideas from a number of contributors. Distributed operating and file systems were also developing in the early 1970s. Another big step, of course, was the creation of the Domain Name System.

One thing discussed and tried early (in 1969 and 1970) was remote execution of code (Jeff Rulifson's DEL and Michel Elie's NIL), but it never really got off the ground. I'm not sure why — certainly the ARPA LISP community had been interchanging (but not in real time) interpretable code well before this time.

However, success of the Internet was far from certain. Computer vendors reacted to the Internet by bringing out proprietary network architectures to try to keep customers locked to into their own equipment, and they weren't fond of the idea of connecting to the equipment of other vendors; the telephone company continued to disparage the Internet concept; some other countries tried to push alternatives to the Internet as it was emerging from the U.S.; and so forth..

Nonetheless, over the next couple of decades the Internet grew. I probably need to consider the contributing factors more carefully, but roughly speaking, contributing to the growth of what we now call the Internet were:

- PCs getting invented and going everywhere, particularly to company branch offices, where they were a way to escape from the mandated computing standards of the central IS authority
- LANs connecting the PCs within branch offices
- bridges connecting the LANs and routers connecting the branches
- Unix spreading TCP/IP (with particular help from SUN), albeit to a limit community

- Netware (as I remember) supporting a limited form of Internet access
- ARPA fighting off rival international standards to TCP / IP

Eventually, the computer vendors had to bow to customers's demands to include TCP / IP along with their proprietary network architecture, thus enabling users to find relatively easy solutions to many interconnection problems and reinforcing the value of the vender independent Internet technology. In the 1990s the WWW exploded on the scene (based on strong consensus and running code, like many new Internet capabilities before it) with its http protocol, html format, and GUI interface that allowed anyone to do what we internet pioneers had been doing more or less manually for years. What Malcolm Gladwell calls "the tipping point" finally was reached — the point, as happened earlier with fax or answering machines — where one thinks, "so many other people have that, I better have it too."

Looking back

Why did packet-switching and the Internet get off the ground when it did, starting in 1968-69, and evolve so successfully?

- The time was right in 1968 — the cost of computing in some sense dropped below the cost of communications, and packet switching could be implemented.
- The U.S. government in the form of ARPA funded some groups of smart people to do some things ARPA thought should be done — despite free marketeers' ideological abhorrence, ARPA "picked winners" — and the government continued to push the Internet for many years, forcing it on a largely unwelcoming world.
- The Internet was designed for decentralized operation in support of interoperability (despite myriad commercial pressures to the contrary) with automatic adaptation to network changes
- The evolution was based on coherent engineering — but without waiting for perfection — rather than on vendor or national authority politics.

The continuing evolution of the Internet, it seems to me, is now much more dominated by commercial concerns. I'm sure this is a good thing in some ways. I'm sure many of you from Silicon Valley think it is all for the best. However, being an old guy now — who can no longer do anything technical, but can only talk of what he used to do — I wonder if the old way might have been a better way.

Thank you very much.

Acknowledgments

I worked with and learned from many people in the years I was involved with the beginnings of the Internet. Much I have said was probably said first, and

better, by someone else. In particular, to prepare for this presentation, I reread a forward by Steve Crocker to an unpublished book of the early RFCs.