THE END OF THE END-TO-END ARGUMENT

[David P. Reed](http://www.reed.com/contactinfo) (http://www.reed.com/contactinfo)

April, 2000

Author's Note:

I have some hope of using this note and any other soapbox I can find to point out why the end-to-end approach creates value. There's too much BS out there linking the end-to-end approach solely to anti-government, anti-monopoly emotionalism, and not enough thought about how value is created in a decentralized world where preserving options and flexibility pays off for all.]

Can it be over?

I still remember as if it were yesterday: that day in Marina del Rey in the late '70's when we split TCP into TCP and IP. After months of intense lobbying by me, Danny Cohen, and Steve Crocker, with support from John Shoch, we agreed to architect the primary protocols of the Internet with only datagrams at the center. Vint Cerf and Jon Postel were persuaded to take a risk on a new style of network architecture, based on a radical decentralization of function.

You can bet it was controversial. No large scale network had ever been architected in this way. The language of networking was defined in terms of "sessions", "flows", and other notions that allowed every switchpoint, every routing agent, etc. to know the purpose and meaning of the bits it was transporting.

Danny Cohen made the case that packet voice innovation required this new structure. The streams and sessions of virtual circuit protocols amplified the effect of bursty traffic to create unintelligible speech. He made a compelling argument that only overprovisioning and adaptive protocols at the endpoints could solve the problems of sending real-time traffic over a heterogeneous network.

I and John Shoch pointed out that the needs of computer-computer connectivity could not be satisfied by traditional point-to-point connections. Much computer-computer traffic involved exchanges of single packets, and often the pattern forms a web of relationships among many computers, which would respond to a request by forwarding or broadcasting messages to many partners. We pointed out the potential for applications that would assemble information rapidly from many sources (anticipating, but not inventing, the structure that supports the World-Wide Web invented 15 years later).

My office-mate at MIT, Steve Kent, now chief scientist at BBN Technologies, recognized that, in a heterogeneous network, encryption and key management cannot be done at the network level without introducing unacceptable security risks. He further recognized that by moving security functions out off the network, a wide variety of security regimes could co-exist.

The idea of a heterogeneous backbone based on high-performance, overprovisioned transports that would provide only best-efforts routing and delivery of datagrams had breathtaking implications. It could scale with few builtin limits. It would not require a single central authority to define what applications and devices could be connected to the network, or what new protocols could be invented and deployed.

This idea of radical simplification was captured in a paper I wrote with two MIT colleagues, Jerry Saltzer and Dave Clark, called The end-to-end argument in systems design. In that paper we argued that many functions can only be completely implemented at the end points of the network, so any attempt to build features in the network to support particular applications must be viewed as a tradeoff. Those applications that don't need a particular feature will have unnecessary costs imposed on them to support the other applications that benefit. We argued that building in such functions is rarely necessary, and that systems designers should avoid building any more than the essential and common functions into the network.

This design approach has been the bedrock under the Internet's design. The e-mail and web (note they are now lower-case) infrastructure that permeates the world economy would not have been possible if they hadn't been built according to the end-to-end principle. Just remember: underlying a web page that comes up in a fraction of a second are tens or even hundreds of packet exchanges with many unrelated computers. If we had required that each exchange set up a virtual circuit registered with each router on the network, so that the network could track it, the overhead of registering circuits would dominate the cost of delivering the page. Similarly, the decentralized administration of email has allowed the development of list servers and newsgroups which have flourished with little cost or central planning.

Yet just when the possibilities hoped for by those folks in Marina del Rey are proving true, and just when the impact of solid-state physics, integrated optics, and software radio are creating unprecedented exponential growth in network capacity, we are starting to hear the call for centralized management, for that same centralized management that we associate with the phone companies.

It seems that "broadband" services "require" that new capabilities be built deep into the network. We "see" the need to have the network have knowledge of who is at the endpoints in order to personalize service to the users. "Experts" claim that packet voice requires specially defined "quality of service" to be built into the network.

What's changed? Was the end-to-end argument wrong?

I don't think so.

What we are seeing now is the same debate we had back in the months leading up to that day in Marina del Rey. It's the same tradeoff being considered. Should we optimize today's applications and patterns of usage by building functions into the network? Or should we find ways to optimize today's applications by building as little as possible into the core of the network?

Back in Marina del Rey, in the mid-to-late '70's, it was not at all obvious what computer networks were good for. The existing data networks (such as the Arpanet, Tymnet, and IBM's SNA) were used primarily to connect data terminals (remote consoles and printers) to time-shared mainframe and mini- computers, or for the radical new idea of "file transfer" between computers. The ideas of distributed data-sharing applications, group information sharing, and packet voice were for dreamers. We could not prove to the skeptical network engineers that only a few years later no one would want to "log in" over a teletype to a remote TENEX machine shared with hundreds of other users across the United States. The users of the day demanded such capabilities, and their "requirements" did not include a network built on datagrams, or with the core functionality of the network moved out to the endpoints.

This sort of argument is exactly what we see today. Today's applications (eCommerce storefronts, telephone calls routed over IP networks, streaming video broadcast of Hollywood movies, and banner-ad-sponsored web pages) are being used to justify building in idiosyncratic mechanisms into the network's core routers and switches. Though it is clearly not possible to meet the requirements of today's hot applications solely with functionality in the network's core, we are being asked to believe that this is the only possible architecture. Implicitly, we are being told that the impact of building these structures into the network is worth the cost of erecting major barriers to future innovation.

In addition to economic friction against innovation, we are creating points of control, where a new class of "trolls" are being permitted to set up shop under our network bridges. These trolls (the companies who develop, and their customers who deploy and operate these special mechanisms) must be consulted and are required to bless any new protocols or applications. Just ask a company like RealNetworks, which must negotiate with firewall vendors, ISPs and other troll-like intermediaries to clear paths for its innovative streaming media protocols. In the Internet's end-to-end design, the default situation is that a new service among willing endpoints does not require permission for deployment. But in many areas of the Internet, new chokepoints are being deployed so that anything new not explicitly permitted in advance is systematically blocked.

What dreams could be bigger than today's applications? Why should we not start building into today's Internet backbone a new kind of network intelligence that optimizes e-commerce transactions, video broadcast, and isochronous phone calls? Let me list a couple, knowing that, as a dreamer, I cannot prove the reality of my dreams of the future.

Gadget internetworking. While there are only billions of people that can sit in front of terminals, there may be hundreds of billions of devices that will perform various functions for us, which will need to communicate with each other. Motion sensors, security alarms, intercoms, thermostats, refrigerators, swimming pools, baby monitors, and electronic instruction manuals all need to coordinate their efforts on our behalf. Just as the web combined with the banking network and overnight delivery to make e-commerce happen, the systems of synergy among devices cannot be anticipated, but will be huge. But when we've built into the network the notion that each new device corresponds to a human subscriber who must be registered and pay a monthly fee (as we are doing in high-speed home networking), we make the problem of defining new synergies very hard. And when we optimize the network to work well only for static web pages, we lose the flexibility to support very different patterns of communication.

Collaborative creative spaces. With broadband networks we are reaching the point where "pickup" creation is possible - where a group of people can create and work in a "shared workspace" that lets them communicate and interact in a rich environment where each participant can observe and use the work of others, just as if they were in the same physical space. Yet the architects who would make the network intelligent are structuring the network as if the dominant rich media communications will be fixed bandwidth, isochronous streams, either broadcast from a central "television station" or point-to-point between a pair of end users. These isochronous streams are implicitly (by the design of the network's "smart" architecture) granted privileges that less isochronous streams are denied - priority for network resources. There are no mechanisms being proposed in these architectures to allow new applications that may be more "important" squeeze out isochronous traffic. Is it really the case that tight timing requirements of packets in a voice stream means that those packets' delivery should always take precedence over events with loose short-term timing, but vast societal impact? That is what these network engineers take for granted.

Is it the end for the End-to-end Argument?

I would argue that we need more than ever to understand it and to apply it as we evolve the network. Sadly, ignorance and a lack of critical thinking puts it at risk. Future potential is hard to visualize, but our inability to make out the details should not justify locking the doors against new ways to use the network.