

MAIN MENU:	CALENDAR	SYLLABUS	DISCUSSION	ADMINISTRATION
NOTICES:	Welcome to Electronic Commerce!			

**Technology for the Digital Lawyer II:
The Architecture of the Internet**
[Thursday, September 6, 2001]

READINGS

The Fundamentals: Packet Switching

The Internet, of course, is simply a network of computers. That is, all computers on the net are able to communicate with each other. Fundamental to this ability is the concept of "packet switching."

Packet switching is simply the technique of utilizing a shared communications link (i.e., a wire) by slicing the information to be communicated into small pieces, transmitting the small pieces, and reassembling the pieces at the other end of the transmission. That is, if the information one wanted to communicate was the phrase "HELLO THERE," packet switching would involve splitting the phrase into two packets: "HELLO" and "THERE", perhaps. The two packets would then be transmitted (and travel) across the net independently. When both arrived at their destination, the packets would be reassembled, and "HELLO THERE" would appear.

For example, say that we want to send the message "HELLO THERE" from location A to location B:

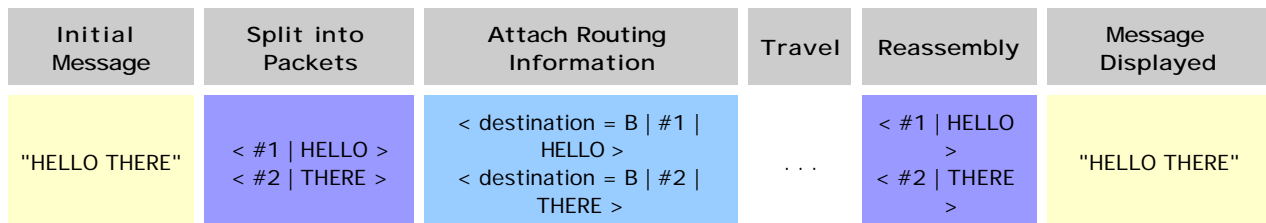


Figure 1.

In contrast to packet switching is the older method of "circuit switching," used most widely for telephones. In circuit switching, a complete connection between the two devices (phones, computers, etc.) is established for the duration of the call. Sometimes this is done physically (think of the old-fashioned telephone operator physically plugging in wires to a connection board). Nowadays, of course, this is all done with electronics and software, but the concept is the same.

Sending our "HELLO THERE" message from A to B:

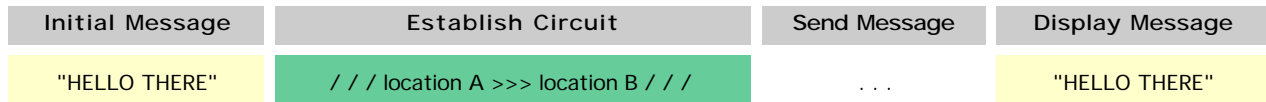


Figure 2.

A big advantage of packet switching is that it allows for much greater sharing of connection resources. Allowing millions of packets to flow over a given communications link turns out to be a more efficient use of the bandwidth (*i.e.*, the amount of connection available) than dedicating a complete circuit to every connection.

Another advantage is that packet switching is more "scalable" -- it allows for faster growth of connection traffic. In packet switching, as the links get busy -- as too many packets crowd into the available bandwidth -- the packets slow down, but they don't (in theory) stop altogether. In contrast, circuit switching typically allows only a finite number of connections; after that limit is reached, you get a busy signal. This difference has a number of implications:

David S. Isenberg, *The Dawn of the Stupid Network*, ACM Networker 2.1, February/March 1998.

A First Look at Internet Standards: TCP/IP Explained

In the Internet networking context, each packet includes (usually in what is called a "header") the destination address, among other items. This allows each packet to effectively stand alone as it travels across the 'net.

The addressing system of the 'net is specified by the TCP/IP protocol. TCP/IP is a set of standard operating procedures that allow all the computers hooked to the net to communicate with each other. In fact, TCP/IP is two separate protocols: the TCP ("terminal connection protocol"), which generally specifies the transportation scheme, and the IP ("Internet protocol") which sets forth the way packets are created and addressed.

Under TCP/IP, each device hooked to the network has a unique "IP address" - a series of four numbers, each ranging from 0 to 256. In a complete IP address, the numbers are separated by "dots": for example, my office computer has an IP address of 130.91.144.105.

When I want to communicate (via TCP/IP) with another computer, I send packets to that computer. Each packet I send includes the destination address (and my source address, in case packets get lost or damaged in transit) and will be "routed" across the network.

The IP address, then, is the street address of Cyberspace, the basic location descriptor of the Internet. It is what allows surety that the communications sent to a particular computer will be received by the destination device. All devices on the Internet must use the IP address system to be recognized by the rest of the Internet - without recognition, the device does not actually "exist" on the Internet. This "enforcement" of the IP address standard is not upheld by government decree, but rather by the "force" of coordination and the desire for network interoperability.

Putting It All Together: Layers & Protocols

It is useful to think of computer networking as a series of "layers" -- with the interaction with the user at the "top" and the actual physical hardware at the bottom. In a greatly simplified manner, we might depict this as:

the user

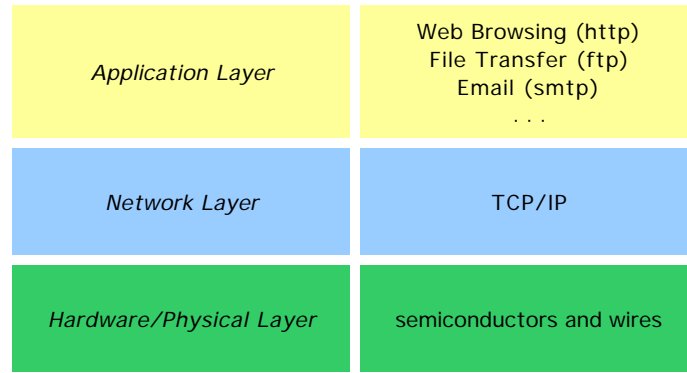


Figure 3.

The applications are the computer software programs we're used to interacting with. Many of these have their own standard operating procedures (or protocols), which allow programs from different providers to work together. For example, essentially all Internet email programs use the SMTP protocol. Other applications (such as Napster and others) use specific protocols more closely associated with the program itself.

The network layer (TCP/IP) allows the applications to access data communication and transportation functions in a standard way. For example, when you send an email, your email program (e.g. Netscape Messenger, Eudora, Outlook Express) formats your message using the SMTP protocol -- so it can be read and understood by any email application. Yet to communicate that message across the 'net, the network layer uses the TCP/IP protocol to make, address, and transmit packets containing the SMTP data. These packets, of course, travel through the physical layer (in the form of electrical signals, typically).

Networking I: Routing the Packets

Up to this point, we've glossed over the way that packets are transmitted from computer to computer. That is, we've viewed the system as follows:



Figure 4.

The reality is more complex. Specialized computers, called "routers," perform the task of directing packets to their proper destination.

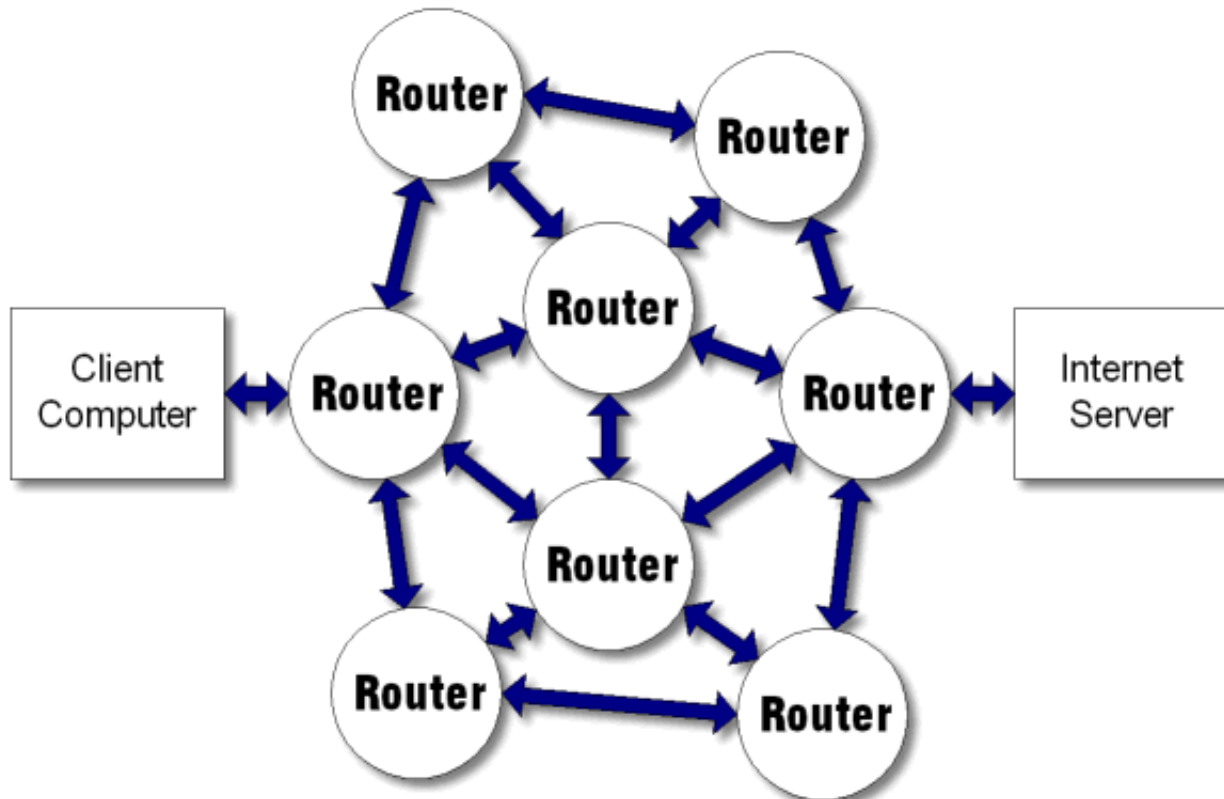


Figure 5.

Routers might be thought of as sort of electronic traffic cops -- they tell the packets where to go and send them on their way. That is, each packet includes a destination address: a description of where the packet is supposed to go, expressed as an IP address. A router contains information about the direction the packet must travel to get to various addresses. Using that information, the router retransmits the packet in the appropriate direction -- usually to another router.

Think of routing this way: say you wanted to travel from Philadelphia to Los Angeles. You show up at the Philadelphia Airport and announce that you want to go to Los Angeles. The Airport staff tells you that the quickest way to get to Los Angeles is through Chicago. So you go to Chicago, announcing upon your arrival that you want to go to Los Angeles; the staff there directs you to a plane to San Francisco. Once in San Francisco, you are put on a plane for Los Angeles.

While this might seem like an inefficient way to get from here to there, this is how a packet gets routed across the 'net. Each router contains information about the "routing" required to send a packet to other locations on the 'net. Packets flow through a router at very high speed, each being directed to what the router's information tells it is the best/fastest way to travel to the packet's destination.

One benefit of Internet routing is that it can be changed quickly. That is, the router information tables can be updated instantly, redirecting packet flow away from an area of network congestion, or damage, etc. Continuing our plane travel analogy, imagine that you arrived at the Philadelphia Airport and were directed to Dallas instead of Chicago, because of bad weather in Chicago.

Each stop a packet experiences in its travels is called a "hop." Various software programs, called "traceroutes," allow users to view and analyze the hops. For example, here are results of traceroutes from my office computer in the law school to a couple of web servers across the Internet

Penn Law School >>> Stanford University

```

Find route from: 130.91.144.105
to: www.stanford.edu (171.64.14.238)

1 lawschool-gw.upenn.edu. (130.91.144.1
): 0ms
2 external-gw-fe2.upenn.edu.
(165.123.217.1 ): 0ms
3 local.upenn.magpi.net. (198.32.42.249
): 1ms
4 remote.abilene.magpi.net.
(198.32.42.134 ): 4ms
5 nycm-wash.abilene.ucaid.edu.
(198.32.8.46 ): 8ms
6 clev-nycm.abilene.ucaid.edu.
(198.32.8.29 ): 20ms
7 ipls-clev.abilene.ucaid.edu.
(198.32.8.25 ): 26ms
8 kscy-ipls.abilene.ucaid.edu.
(198.32.8.5 ): 35ms
9 dnvr-kscy.abilene.ucaid.edu.
(198.32.8.13 ): 46ms
10 snva-dnvr.abilene.ucaid.edu.
(198.32.8.1 ): 71ms
11 198.32.249.161 (198.32.249.161 ):
71ms
12 stan--sunv.pos.calren2.net.
(198.32.249.74 ): 72ms
13 i2-gateway.stanford.edu.
(171.64.1.214 ): 72ms
14 core3-gateway.stanford.edu.
(171.64.1.222 ): 72ms
15 core6-gateway.stanford.edu.
(171.64.3.97 ): 73ms
16 leland-gateway.stanford.edu.
(171.64.1.230 ): 74ms
17 www4.stanford.edu. (171.64.14.238 ):
74ms

```

Penn Law School >>> London School of Economics

```

Find route from: 130.91.144.105
to: www.lse.ac.uk (158.143.192.210)

1 lawschool-gw.upenn.edu.
(130.91.144.1 ): 1ms
2 external-gw-fe.upenn.edu.
(165.123.237.1 ): 1ms
3 local.upenn.magpi.net.
(198.32.42.249 ): 0ms
4 remote.abilene.magpi.net.
(198.32.42.134 ): 4ms
5 nycm-wash.abilene.ucaid.edu.
(198.32.8.46 ): 8ms
6 ny-pop.ja.net. (193.62.157.209 ):
8ms
7 us-gw2.ja.net. (193.62.157.17 ):
75ms
8 london-bar1.ja.net. (128.86.1.43 ):
76ms
9 ulcc-gsr.lmn.net.uk. (146.97.40.34
): 76ms
10 atmr-ulcc.lmn.net.uk.
(194.83.100.202 ): 75ms
11 lse.lmn.net.uk. (194.83.101.198 ):
85ms
12 cat1-rsfc.lse.ac.uk.
(158.143.220.141): 84ms
13 www.lse.ac.uk. (158.143.192.210):
79ms

```

Figure 6.

Some explanation of what you're seeing:

- Each line represents a "hop" for the packet -- i.e., the passage through a router. Here, my trip to Stanford took 17 hops.
- Each router has a unique address (and name). The first entry on each line (after the number) is the URL (uniform resource locator, using the domain names system) for the router. The second entry, in parentheses, is the IP address of the router.
- The numbers following the IP addresses (i.e., 13ms) represent the length of time it took for the packet to reach the given router. For example, it took 74ms to reach the www4.stanford.edu machine.

Note that routers are often named suggestively of their locations. For example, in the above example, it appears that my packet to Stanford went through Cleveland, OH (hops 6-7), Indianapolis, IA (hops 7-8), Kansas City, MO (hops 8-9), Denver (hops 9-10), before getting to the Stanford systems. (Note also that all the routers ending in "ucaid.edu" are on the famed "Internet 2" system - more about this below.)

Where do you think the packets to the London School of Economics went?

Next, try this yourself. Using the link below, you can run a traceroute from all over the world to your computer. Run a few traceroutes, and try to guess where the packets are going:

TraceRt.org, [Multiple Traceroute Service](#)

[select a location to trace to, and click the "trace" button]

Networking II: Understanding the There There

The "Internet," then is a network of networks -- the vast **interconnection** of routers and computers. For example, the Penn network looks something like this:

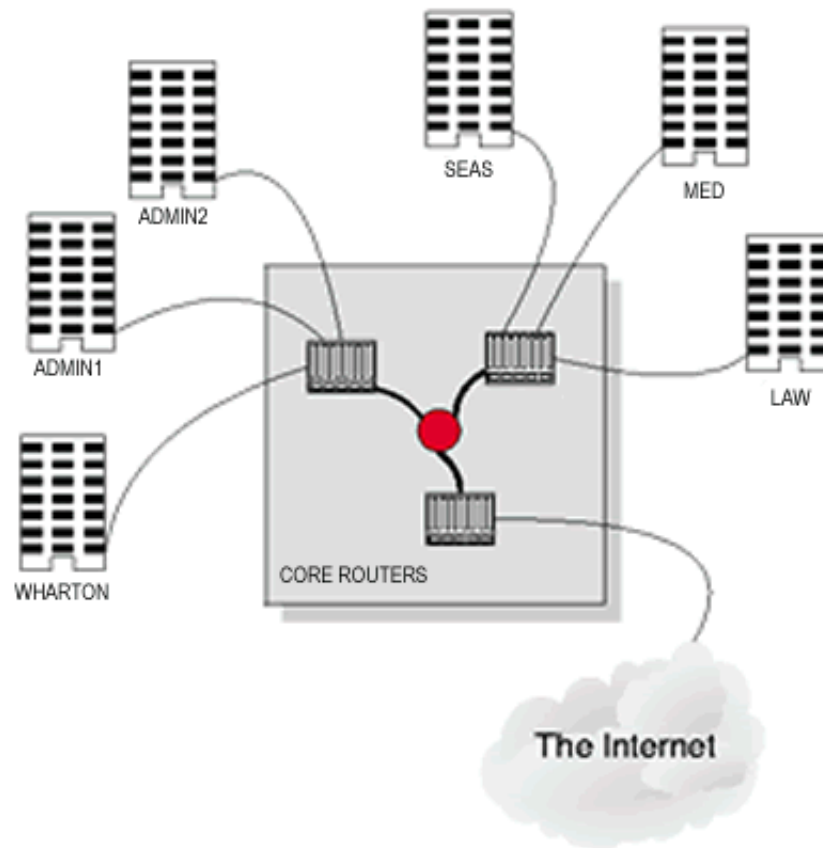
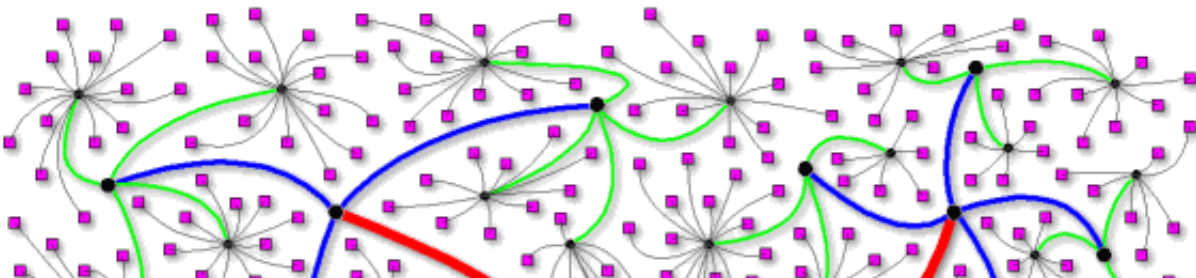


Figure 7.

But by looking more broadly, we can see that interconnecting all these networks results in something like this:



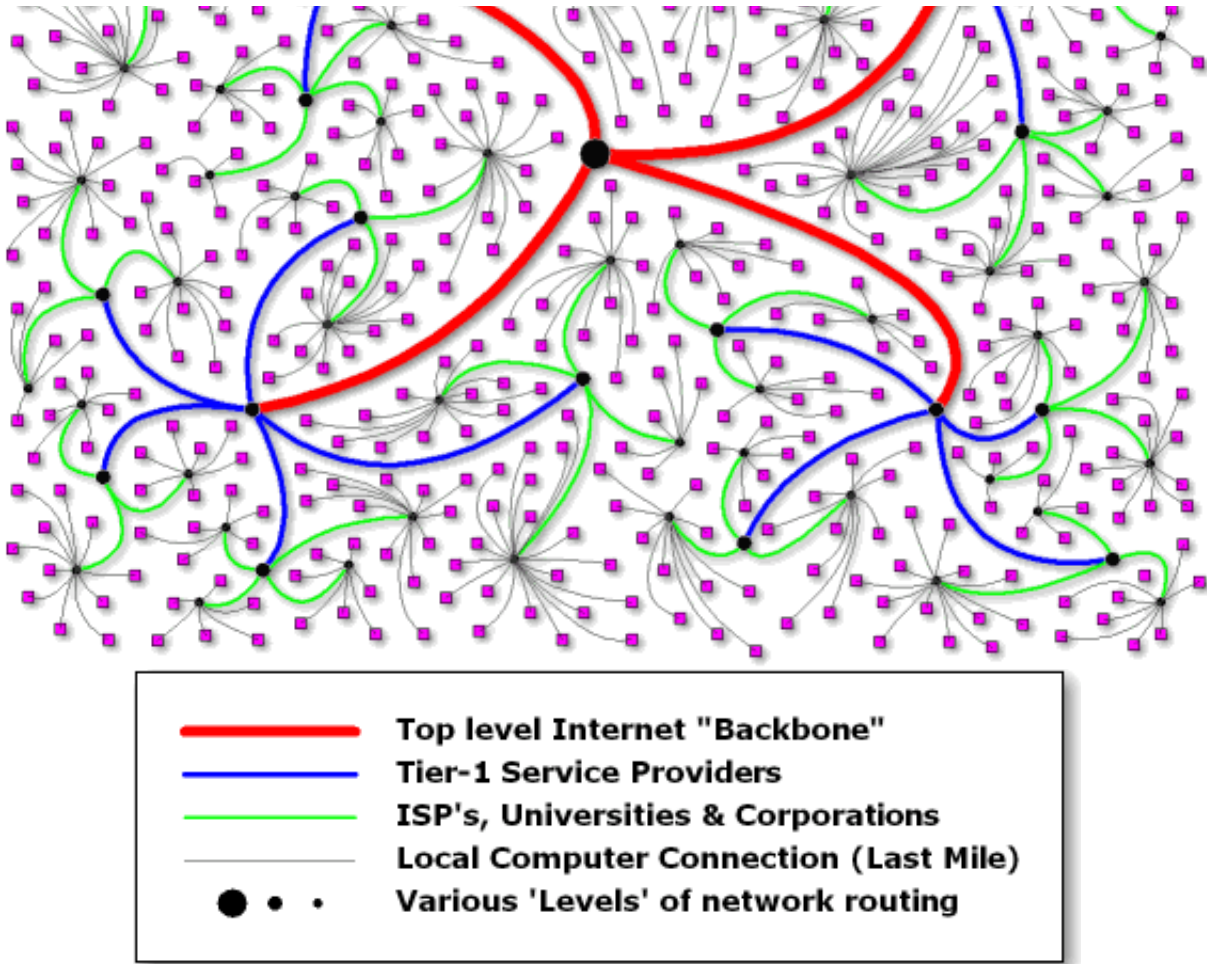


Figure 8.

The image above is a simplified representation of the "network of networks" that is the Internet. Note that all connections are not equal: for speed and efficiency, certain links offer higher speed and capacity over long distances. At the top end of this range is what is called the "backbone" -- very high speed, high-capacity, long-distance connections. Think of these as interstate highways. (Except, of course, these highways are privately owned, by major telecommunications companies, such as Sprint, AT&T, and Worldcom.)

For example, this is the backbone map for the *Abilene* network, which is developed by the Internet2 project (and is the way that our packets to Stanford traveled earlier):





Figure 9.

Within a particular region, the "Tier 1" providers (again, typically private companies) offer shorter high-speed links (these are the blue lines in Figure 4 above). These links provide the connection between the major backbones and the Internet Service Providers (ISPs), Universities, and large Corporations - which provide the interface between the users and the Internet. The University of Pennsylvania uses the Magpi network (a partnership between Penn and others) and as a Tier 1 provider:

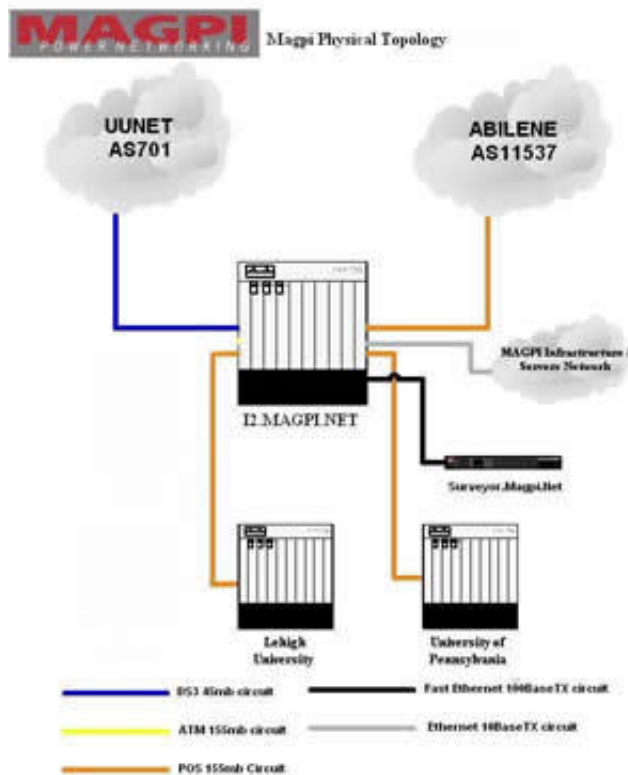


Figure 9.

So if we return to the traceroute shown in Figure 6, we can see that the packet from the law school traveled across several different "networks" to reach Stanford -- starting with the Penn network (Figure 7), then the Magpi network (Figure 9), and across the country on the Abilene network (Figure 8) -- even before reaching the various networks serving Stanford. Consider whether this layer-upon-layer networking structure might have legal or business

implications.

Finally, we'll add one (small) additional bit of complexity. You might have noticed that the depiction of the Internet in Figure 8 is not particularly "distributed" - with a single point of interconnection in the center of the diagram. In point of fact, there is no such "single point" of interconnection on the Internet. This is because the various providers (at all levels, but especially the backbone level) interconnect with each other broadly, allowing traffic to travel on each others' networks -- called "peering." When you factor in peering and duplication (i.e., multiple providers), the Internet is better depicted as:

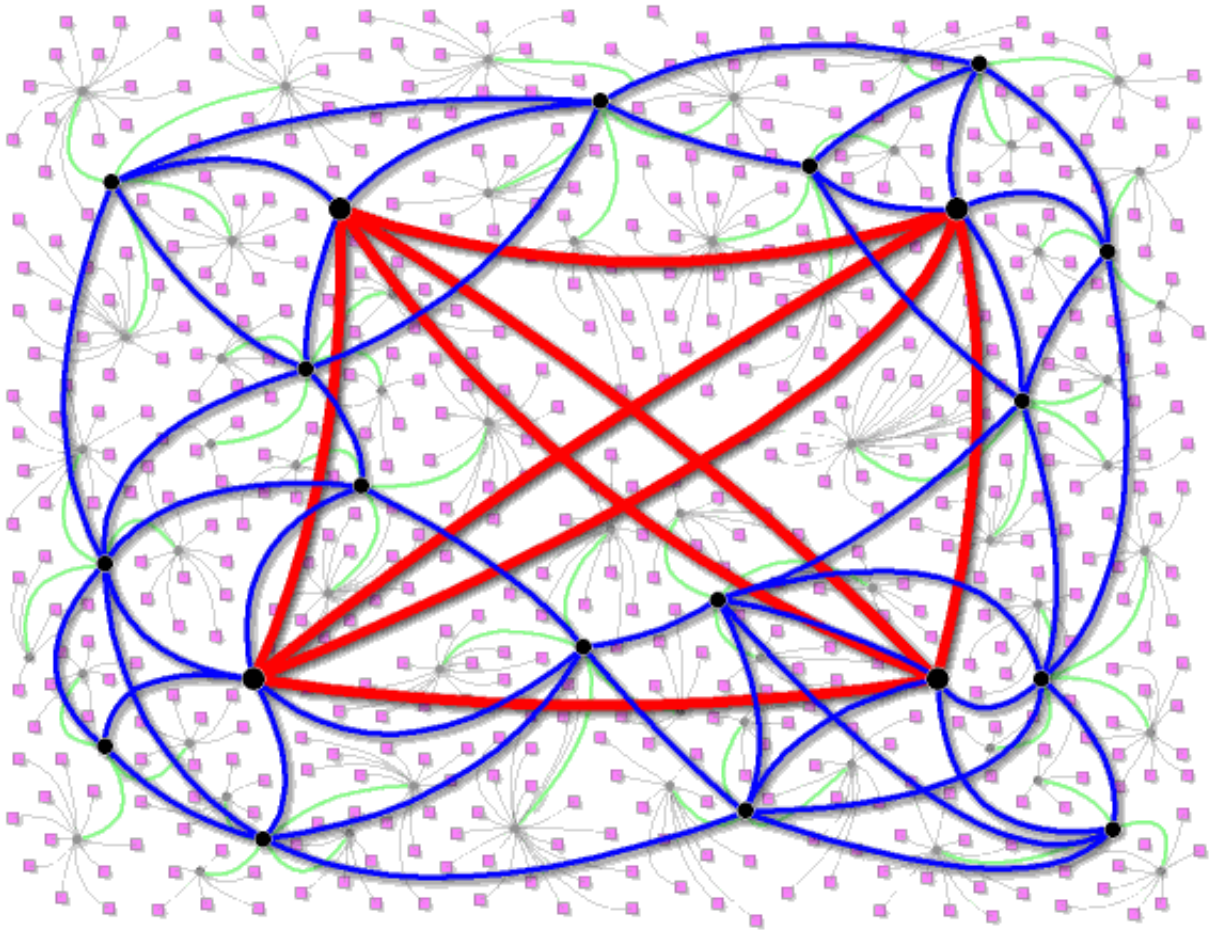


Figure 10.

These peering arrangements are not always simple and uncontested, as the following short article explains.

Denise Pappalardo, When private peering arrangements go bad, Network World, June 11, 2001.

Networking III: Understanding the Network

Now that we've considered the basics, we turn to a couple more advanced topics.

Geography and the Network

Many people suggest that geography -- that is, realspace geography -- doesn't matter on the Internet. Yet a deeper look finds that geographic location is both ascertainable and important on the Internet. As you review the following, consider the legal, societal, or business implications of the rise of geography online.

For example, the "NetGeo" service offers a database linking geographic data with IP addresses. For example, the entry for my office IP address (130.91.144.105) returns the following record:

TARGET: 130.91.144.105 NAME: UPENN-SUBNET CITY: PHILADELPHIA STATE:
PENNSYLVANIA COUNTRY: US LAT: 39.96 LONG: -75.20

[Cooperative Association for Internet Data Analysis \(caida\), NetGeo Service](#)
[exercise - find the location of yahoo.com]

Also, consider the implications of the increasing popularity of services that push data to the "edges" of the Internet, such as Akamai:

[Akamai, Overview of EdgeSuite Technology \(2001\)](#). [[Shockwave](#) (free download)]

Are All Packets Created Equal?

Thus far, we've assumed that all packets that travel on the Internet are treated equally. In fact, this is close to the reality today. Yet many suggest that this is an inefficient use of network resources, and that Internet network should incorporate differentiation between packets -- typically called Quality of Service (QoS) systems.

[Internet2, Overview of QoS \(2001\)](#). [[Shockwave](#) (free download)]

Furthermore, there are a number of new technologies that provide real-time analysis and processing on TCP/IP packets themselves:

[Packeteer, Packetshaper Overview \(2001\)](#).
[review the technology -- consider what it does]

NOTES & QUESTIONS

1. The Virtues of End-to-End. What David Isenberg calls the "stupid network" (i.e., the decentralized nature of packet switching and TCP/IP technology) is often referred to as "end-to-end" network design. Conceptually, such a design pushes the "intelligence" -- the complexity, the decisionmaking, the actual applications -- out to the ends of the network connection, while the middle is as simple and straightforward as possible. Take the world wide web for example: all the "action" takes place at either the user's end (the browser application) or on the server end (the web server application); the network itself simply moves the packets from place to place. What are the advantages and disadvantages of this approach? Think in terms of speed, complexity, ability to grow. Now consider the non-technical implications of such a policy choice: does an end-to-end design make it harder or easier to fashion legal frameworks for the Internet/eCommerce? Should that matter?

The rise of Akamai, Quality of Service (QoS) and traffic management (i.e., *Packeteer*) applications demonstrates that many in the industry see an increasing need for intelligence in the network. Why would an eCommerce player move towards a more intelligent network? Are the arguments only technological in nature, or is there a legal or policy component as well?

Finally, as the Internet and eCommerce matures, is there reason to think that end-to-end has outlived its usefulness?

2. Who Runs the Internet? One common misperception is that the Internet is akin to the public highway system -- a sort of public facility, open to all. The reality is that, formally, at least, the Internet is very much a private business enterprise, with hundreds of providers at various levels, most with profit as the motive. How do the various "owners" of the Internet get paid?

Also consider who has "leverage" in the Internet network. Is there a company or group of companies that has a large influence upon the traffic flows across the Internet? Is this something to think about in terms of eCommerce legal frameworks?

COPYRIGHT © 2001 R. POLK WAGNER.



The Dawn of the Stupid Network

By David S. Isenberg

Originally published as the cover story of ACM Networker 2.1, February/March 1998, pp. 24-31.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists requires prior specific permission and/or a fee. Copyright ACM1072-5220/98/0200 \$5.00

In recent history, the basis of telephone company value has been the sharing of scarce resources -- wires, switches, etc. - to create premium-priced services. Over the last few years, glass fibers have gotten clearer, lasers are faster and cheaper, and processors have become many orders of magnitude more capable and available. In other words, the scarcity assumption has disappeared, which poses a challenge to the telcos' "Intelligent Network" model. A new type of open, flexible communications infrastructure, the "Stupid Network," is poised to deliver increased user control, more innovation, and greater value.

Telephone companies (telcos) have always pushed technology improvements that promote the smooth continuation of their basic business. They invented the stored program control switch in the 1970s, as a move toward cost reduction and reliability. Programmability also made possible certain call routing and billing services. In the 1980s, phone companies began marketing these services as the "Intelligent Network." Technology continued its trajectory of improvement, but because technology began to change the value proposition in ways that the old business could not assimilate, the telcos seemed to "fall asleep at the switch" at the core of their network. Meanwhile, the Stupid Network – based on abundant, high-performance elements that emphasized transmission over switching, as well as user control of the vast processing power at the network's edges – was taking shape.

KEEP IT SIMPLE, STUPID

"Keep it simple, stupid," or KISS, is an engineering virtue. The Intelligent Network, however, is anything but simple; it is a marketing concept for scarce, complicated, high-priced services, surrounded by features like 800 service, call waiting, and automatic calling card validation. These are intertwined in the network architecture in a plethora of service adjuncts, each with its own systems for operations, provisioning and maintenance. This complicated, centrally controlled amalgam of systems is born of a single application - two-way real-time voice communication.

So what exactly is a Stupid Network? George Gilder observed more than five years ago, "In a world of dumb terminals and telephones, networks had to be smart. But in a world of smart terminals, networks have to be dumb." In a Stupid Network, control passes from the center to the edge, from the telco to users with an abundance of processing power at their fingertips. The center of the network is based on plentiful infrastructure – cheap bandwidth and switching – that is about as smart as a river. The water in a river, like a data object in a Stupid Network, gets to where it must go adaptively, with no intelligence and no features, using self-organizing engineering principles, at virtually no cost. Bits go in one end and come out the other. Data flows – like water – define the movements and channels within the system.

Eric Clemons, a professor at the University of Pennsylvania's Wharton School of Business, makes the distinction between strategy and doctrine. "Strategy," he says, "is learning how to deal with dogs. Doctrine is about belief: "Dogs don't do that." Telco doctrine, formed in the age of monopoly and scarce infrastructure, is rarely examined explicitly. When there was only one telephone company, what Ma Bell did defined how things were. So today, even though the new era of competition requires clear thinking and new beliefs, telco culture inextricably mixes doctrine and strategy.

THE CIRCUIT-SWITCHED LEGACY

The Intelligent Network concept has its roots in the first software-controlled switches in the 1970s. In those days, working with computers meant writing code to save a byte here and an instruction cycle there. Current software practices, such as object-oriented programming, were too inefficient for prime time and were relegated to the confines of academia.

Thus, telephone network equipment was designed in a climate of scarcity. Consider the local exchange, represented by the three digits of a telephone number that follow the area code (the nxx in the pattern nxn-nxx-xxxx). The local exchange "owns" the last four digits of a telephone number. Theoretically, a local exchange can serve up to 10,000 telephones, e.g., with numbers 762-0000 through 762-9999. The design assumption, though, is that only a certain percentage of these lines, maybe one in 10, are active at any one time. But should these assumptions change temporarily (e.g., an earthquake in California) or permanently (calls to AOL lasting several times longer than normal voice calls), the network hits its limit. Then, getting a dial tone becomes a matter of try, try again.

Even more assumptions have now changed permanently. Before 1996, long distance carriers like AT&T, MCI and Sprint cranked a precise set of capacity planning equations that told them which switches would need more circuits, which routes needed to be upgraded over the next year, and where to plan for new switches. Suddenly, increased Internet usage threw the telcos an unplanned 60% increase in data traffic. Suddenly, some points of their network hit capacity. The telco fallback position had always been to lease capacity from their rivals, but this wasn't available either, because the other telcos were maxed out as well.

"INTELLIGENT" NETWORK SERVICES

In the late 1970s, telcos became fixated on their expensive investments in computer-controlled switching, and were intrigued by the prospect that they could do "intelligent" things with these investments. They reduced the cost of running the network and formed a platform for revenue producing services geared toward call set-up and billing. The concept of network control was extended to let digital switches communicate with databases (known as Service Control Points) and signal processing systems (Intelligent Peripherals).

Intelligent Network (IN) specs were meant to encourage telecom equipment vendors to design their equipment to work in a multi-vendor environment, so telcos would not be locked into one supplier. In addition, IN equipment was designed to work with certain customer systems and databases. Some common Intelligent Network services include: routing calls to a number other than the one the caller originally dialed (the basis of 800 service); caller options ("press 1 for customer service," etc.); and supplying calling party numbers directly to customers for database lookup (which is why I must call my bank from my home phone when my new ATM card arrives in the mail).

STUPID IS BETTER

Stupid Networks have three basic advantages over Intelligent Networks – abundant infrastructure; underspecification; and a universal way of dealing with underlying network details, thanks to IP (Internet Protocol), which was designed as an "internetworking" protocol. Some key "two-fers" emerge from these basics: Users gain end-to-end control of interactions, which liberates large amounts of innovative energy; innovative applications are rapidly tested in the marketplace; and innovative companies attract more capital and bright people.

Abundant Infrastructure

In a Stupid Network, if you have congestion, you just add more connections, bandwidth, switching or processing power. If you want reliability, you add more routes or more redundancy. If you need more intelligence for features or services, you add it at the endpoints. As early packet network visionary Paul Baran points out, it is possible "to build extremely reliable communications links out of low-cost unreliable links, even links so unreliable as to be unusable in present networks."

Even as the costs of networks have dropped, capacity has improved manyfold. At the dawn of the digital transmission era, for example, you could run 1.5 megabits – 24 calls – on a coaxial cable as thick as your ankle. Today, network providers routinely put several tens of gigabits – a few hundred thousand calls – on a single glass fiber as thin as a human hair. Switching used to be scarce, too, but now it is equally abundant. Where a human operator could set up maybe 100 calls an hour, modern computer controlled switches, such as Lucent's 4ESS, can now complete about 1 million calls in the same hour. Furthermore, if you consider that routing a single packet is equivalent to setting up a call, routers can now set up 3.6 trillion "calls" an hour. And the prices for these components have come way down: Today when you buy a Gigabit Ethernet switch, you get 1,000 chunks of 64 kbps throughput (each equivalent to a phone call) for every dollar.

This leads to two different models of capital investment. In the telco model, network expansion is a big decision that requires expert engineers, detailed Erlang models of congestion, and several consultants - and takes months, if not years, to implement. But if you're

running a Stupid Network, expansion is no problem. A few hundred gigabits? Put it on my credit card. More switching capacity? Take it out of petty cash.

Underspecification

The Intelligent Network is tightly specified for voice. All other data types require special leased access lines, or awful kludges like modems. The Stupid Network is underspecified - this means bits-in, bits-out. It is nothing special for underspecified networks to carry voice, music, bank balances, e-mail or TV on the same facilities. You stuff bits in one end of the network, and they find their way to the other end of the network. Packets carry their address with them, and out they come at the other end, right where you want them to be.

Underspecification also means that there is little thought for congestion control. So what if there is congestion, or even crashes? On the whole, the convenience of underspecification more than compensates for the occasional jam-up. And you can always add more "infrastructure."

Internetworking

Internet Protocol points the way to a key property of Stupid Networks. The foremost design goal of IP is to cross multiple, physically different networks. To IP, it doesn't matter if the underlying transport is circuit, SONET, Ethernet, Bitnet, FDDI or smoke signals. An IP application works the same no matter what the underlying network technology. This makes the details of how a network works irrelevant (including how "intelligently" it is engineered).

IP neatly takes the provider of the physical network infrastructure out of the value proposition. No matter how intelligent a telco's network might be, if it is running IP, its intelligence is reduced to commodity connectivity. Networks that run IP are left with one main source of distinction: how much connectivity they provide. Thus, the Internet that we know and love is a "virtual network" – a "network of networks" – that is independent of wires and transport protocols.

Because IP makes the details of the network irrelevant, all that matters is that the bits sent by your machine are received by my machine, and vice versa. In an IP communication application, users don't care how the "call" is set up, or even if there is a telephone call that forms part of the communication path between endpoints.

User Control

This means that users are in control of their interactions. Suppose, for example, that two users want to bring a third party into an interaction; they just do it. An IP-connected user does not need to order special three-way connectivity service from the networking company. All that user needs to do is write (or install, or use) a program that sends packets to two different destinations and receives from both of them.

A BOOST TO INNOVATION

This ability to "just do it" liberates huge amounts of innovative energy. If I have a Stupid Network and I get an idea for a communications application, I just write it. Then I send it to my buddy, and my buddy can install it, too. If we both like it, we can send it to more people. If people really like it, then maybe we can charge for it - or even start our own company. Yahoo!

In contrast, the Intelligent Network impedes innovation. Existing features are integrally spaghetti-coded into the guts of the network, and new features must intertwine with the old. For example, until recently you could not get Caller ID for an incoming call when you were on the phone. To fix this, Bellcore had to invent a low-tech, low-functionality, high-complexity protocol called Analog Display Services Interface (ADSI). Call waiting with caller ID, however, would be a no-brainer under Internet telephony – a packet containing caller ID information could be treated just like any other packet arriving at my system, to be interpreted by the end-user IP telephony application.

New Internet capabilities are coming online that will fuel Stupid Network innovation. Internet Protocol Version 6 (IPv6), stabilized in 1995, is becoming available now. Its capabilities include expanded address space, real-time functionality, mobility management and carrier selection, hooks for authentication and data integrity, multicasting, and easy coexistence with and migration from the current IPv4 standard. IPv6 capabilities in the hands of innovators will foster whole new areas of applications.

A QWEST FOR BANDWIDTH

Perhaps the leading proponent of the Stupid Network is Qwest Communications, which is building a huge 16,000-mile-plus SONET based network that will reach 125 cities by next year. According to Qwest's executive VP of products, Nayel Shafei, "Qwest is leveraging the greatest currency of all – our unlimited bandwidth - to shape the future of telecommunications." The company is about to go international, too; it will turn on its 1,400-mile Mexican backbone in mid-1998, and it recently announced a trans-Atlantic link.

The Qwest network will run native IP over SONET, according to CEO Joe Nacchio. This is an industry trend; Sprint, for example, recently went ATM-less on its SONET/IP backbone. This is possible because physical layer infrastructure is becoming more abundant and endpoints are becoming more capable.

Qwest is using its network to compete with established telcos on price. Qwest now offers long-distance telephone service over its IP backbone at 7.5 cents per minute, a 25% discount over the now industry-standard dime.

Shafei maintains that the sound of telephone service over Qwest's network will not have the glitchy, scratchy quality that is associated with Internet telephony today. Calls will be mainlined into the jugular of Qwest's unlimited bandwidth. Consumers will use Qwest's IP long distance service by making a local call on a normal telephone, thus accessing a circuit-to-IP platform by Vienna Systems, a Newbridge Networks affiliate. The Vienna platform will not compress the voice; it will simply packetize the raw, 64 kbps signal and send it via IP. Shafei claims that the quality will be virtually as good as circuit-switched voice.

Several other entrepreneurial companies have tested the waters for Stupid Network innovations. Vocaltec, for instance, was the first company to commercialize the fact that you could use the Internet for voice communication. If I have Internet access, and you have Internet access, and we're both running Vocaltec software, we can talk to each other for as long as we like, for no incremental cost, no matter where in the world we are. Though voice quality is still not ideal, and delay can interfere with the flow of a conversation, look for this new communications niche to merge with other Internet applications to create new value.

Placeware, a Xerox PARC spin-off for interactive multimedia meetings and conferences over the Internet, mixes Internet telephony with data sharing, presentation graphics, and a crude representation of the meeting space. Demos of the technology seem to add a lot to voice conferencing, and it gives a more participatory, less self-conscious feel than a video conference.

BEYOND QOS TO SIMPLE STUPIDITY

Intelligent Network advocates point out that networks need to treat different data types differently. Right now, they're absolutely correct. There is a network for telephony, another network for TV, and proprietary leased-line networks for financial transactions – and none of these are ideal for public Internet traffic. You need to have low delay for voice telephony, the ability to handle megabit data streams with ease for TV, and low error rates and strong security for financial transactions.

Quality of Service (QOS) is an intermediate step in the journey from separate networks to a single, simple Stupid Network. QOS, in standard telco thinking, means a repertoire of different ways of handling each type of data on a single network. If the Stupid Network is to become a bona fide integrated service network, it will need to carry all kinds of data with different needs.

But suppose technology improves so much that the worst QOS is perfectly fine for all kinds of traffic, without a repertoire of different data handling techniques. Suppose, for example, that everyday normal latency becomes low enough to support voice telephony, while at the same time allowing enough capacity for video, plus data integrity strong enough for financial transactions. This would be a true Stupid Network – one treatment for all kinds of traffic.

Skeptics might say that there would have to be dramatic improvements in networking technology for this to happen. Well, we're getting there. Routing switches from Madge and Foundry recently showed performance impressive enough to conclude that routing latency and jitter (variation in packet arrival time) may soon be a negligible issue. But these were lab tests, not field usage, and packet losses were as high as 1% under some conditions. So we are not there yet - but perhaps we will be soon.

PLAYERS IN THE NEW ORDER

Still other technologies of abundance, with the potential to break the telcos' foot-dragging hegemony, have attracted interest from entrepreneurial vendors. Here are some of the leading candidates:

- **LMDS:** This technology provides a wireless broadband last-mile path to the Stupid Network. The FCC LMDS auction, completed on March 25, opens U.S. markets to deployment and service over the next couple of years. The two big bidders were Nextband and WNP Communications. Equipment manufacturers Hewlett-Packard, Stanford Telecom, Texas Instruments, Tadiran and others also will be beneficiaries.
- **CDMA:** Another wireless data access method; Qualcomm is still well positioned. Also watch Broadband CDMA, an emerging open standard that can deliver from fractional T-1 on up. Interdigital is one leading B-CDMA player.
- **Gigabit Ethernet switching:** This technology has moved from laboratory to the marketplace in a remarkably short time. While Ethernet has been synonymous with Local Area Networks in the past, "Neighborhood Networks" are replacing ATM as the vehicle of choice for campus nets. Can real neighborhoods be far behind? Gigabit Ethernet players include Bay Networks, Cisco, 3Com, Cabletron, Foundry, Extreme Networks, and many others.
- **Cable Modems:** Players include set-top box makers General Instruments and Scientific Atlanta, plus Motorola, Hybrid Networks and others. Cable provider Comcast is a good bet for cable modem service – Bill Gates thinks so, anyway (to the tune of \$1 billion).
- **The power companies:** Also worth watching, following Nortel's announcement last November of technology to deliver data to end users over power lines.

All these infrastructure improvements are rapidly making the telcos' Intelligent Network a distinctly second-rate choice. The bottom line, though, is not the infrastructure; it is the innovation that the Stupid Network unleashes. The Stupid Network assures the next paradigm-breaking, market-making "new thing." The only question is who will become the next Netscape, the next Microsoft – or the next Ma Bell. And that's not a stupid question.

The netWorker website that time forgot (v1.1 only) is <http://www.acm.org/networker/default.htm>

Posted on 6 June 1998

Sponsored by:



NetworkWorldFusion

This story appeared on Network World Fusion at http://www.nwfusion.com/archive/2001/121755_06-11-2001.html

When private peering arrangements go bad

Cable & Wireless shuts out 14 ISPs, including PSINet.

By DENISE PAPPALARDO

Network World, 06/11/01

Cable & Wireless likely didn't make any new friends on the Internet last week when it started enforcing its newly revised peering policy, cutting off service to more than a dozen ISPs.

Cable & Wireless terminated private peering network agreements with 14 ISPs, including troubled PSINet, which Cable & Wireless said no longer met the ISP's peering requirements. Private peering is the act of two ISPs establishing dedicated connections to their respective networks. These connections are used to exchange traffic that is destined for each ISP's network. ISPs do not pay service fees for these connections.

After much discussion, Cable & Wireless restored PSINet's connections late on June 5 after PSINet signed a letter of intent stating that it would soon meet Cable & Wireless' requirements. It appears that PSINet was the only ISP of the 14 that was given a reprieve. In the meantime though, Cable & Wireless and PSINet were unable to directly exchange traffic for three days and customers suffered.

One Cable & Wireless customer who asked to remain anonymous told *Network World* that it was forced to come up with alternative means to communicate with partners on PSINet's network. The official at the Southwest chain of stores say it lost 300 transactions that were being sent from a vendor on PSINet's network to the company's headquarters. They had to fax these transactions until Cable & Wireless restored the peering connections.

Clearly this was not the only Cable & Wireless customer affected - most customers associated with PSINet experienced delays or were unable to reach other users. While this situation was set in motion over the termination of a single private peering agreement, it demonstrated the frail nature of the Internet with its series of sometimes loosely interconnected, providers.

ISPs that were cut off by Cable & Wireless can simply go to another provider or set up public peering connections at the congested Metropolitan Area Exchanges or Network Access Points on the Internet. But small and midsize ISPs would be driven out of business if all of the large ISPs changed their policies and started charging all providers with smaller networks.

Standards in peering

Cable & Wireless' private peering requirements make it more difficult for small and midsize national ISPs to set up cost-free direct network connections with the company. Here's how Cable & Wireless' requirements compare:

ISP	Cable & Wireless	UUNET	Genuity	AT&T
Peer network requirement	OC-48	OC-12	OC-3	OC-3

Cable & Wireless, like all of the large IP network service providers, has policies that govern which ISPs are peers and which should pay for

transit service. The idea behind private peering is that both ISPs are exchanging about the same amount of traffic and are operating similar networks.

Cable & Wireless requirements include: the peer must operate an OC-48 backbone with a network operations center that includes a contact person available around the clock. The requirements that Cable & Wireless includes in its policy are not unusual, but its specific OC-48 network prerequisite is. For example, WorldCom's UUNET says its peers must operate an OC-12 network, Genuity says its peers at a minimum must operate an OC-3, as does AT&T.

This requirement eliminates the majority of some 4,000 ISPs from setting up private peering connections with Cable & Wireless. "There are about 10 to 20 ISPs that operate OC-48 networks," says Pat Sheridan, a director at Cable & Wireless. Some of these ISPs include Williams Communications, Level 3 Communications, UUNET, AT&T and Genuity, but not Verio, Savvis Communications or Infonet.

While Cable & Wireless' rules are more stringent than its competitors', the ISP may also be more strictly enforcing its rules. PSINet was not shut down because it does not have an OC-48 network, but because it wasn't sending Cable & Wireless enough traffic. Peers are restricted on how much traffic they can send, but they are also expected to maintain a certain level of traffic.

Cable & Wireless' traffic requirements are the same as Genuity's, AT&T's and actually more liberal than UUNET's. But no other ISP shut down PSINet's peering connections at press time. And while Cable & Wireless says that it's likely leading the pack with its new requirements, others do not agree.

"Some companies engage in posturing to see if the rest of the industry will follow, but that's likely not going to happen in this case," says Craig Uthe, IP product management director at AT&T.

"Some of the smaller ISPs might resent this new policy and will then just switch to another provider, especially if no one follows," he says.

That may be fine with Cable & Wireless. Some have theorized that the ISP is simply trying to develop a new revenue source. Transit fees or wholesale Internet access services can easily cost \$20,000 to \$100,000 per month for OC-3 to OC-12 connections. Or it may be trying to reserve its network resources for paying enterprise customers. But the service provider says it is simply updating its peering policy to match its upgraded network.

All contents copyright 1995-2001 Network World, Inc. <http://www.nwfusion.com>

SEARCH

[Advanced Search](#)

- PLEASE CONTACT ME
- PACKETEER CONTACTS
- CUSTOMER SUPPORT
- PARTNERWEB
- DEMO ROOM

PRODUCTS

- PACKETSHAPER
- APPVANTAGE
- APPCELERA
 - APPCELERA ICX
 - APPCELERA ISX
- CENTRALIZED MGMT
 - POLICYCENTER
 - REPORTCENTER
- EXTENSIONS
- PACKETCARE

PACKETSHAPER

PacketShaper is an application traffic and bandwidth management system that delivers predictable, efficient performance for applications running over the WAN and Internet. The combination of its **classification, analysis, monitoring, and control** capabilities enable network administrators to keep critical traffic moving at an appropriate pace through bandwidth bottlenecks and prevents any single type of traffic from monopolizing the link.



[Product Literature](#)

Feature highlights include:

- Automatic Traffic Discovery: PacketShaper systems automatically identify all applications running across the network. Please see the PacketShaper data sheet for a list of applications.
- Extensible traffic class definitions: Create custom criteria for measuring and controlling traffic. Criteria include address, subnet, port, and URL. Combinations of the criteria are supported, enabling more targeted monitoring and control.
- Real-time traffic monitoring: Traffic network utilization (peak and current rates) for the link and by application.
- Monitor response time for each application. Differentiate network delay from server delay.
- Monitor network efficiency: Determine how much bandwidth is wasted from retransmissions.
- Threshold all performance metrics and automatically notify an administrator via email or SNMP trap when threshold is crossed.
- Policy-based enforcement of application priorities and bandwidth allocation: Directly control bandwidth allocation by application, server, or user to proactively prevent congestion related application performance problems.
- Traffic marking for DiffServ- or MPLS-enabled network
- On-board historical reporting

PacketShaper systems provide typical monitoring features that provide network administrators with valuable intelligence to control their application performance and maximize existing network resources.

By relying on monitoring as a baselining and strategic function rather than for passive observation, PacketShaper products enable organizations to discover and classify applications, analyze their performance, and then enforce policy-based bandwidth allocation based on their business importance. PacketShaper systems generate an array of reports to validate performance results, ensuring that applications are indeed aligned with business priorities.

4 STEPS TO CONTROLLING APPLICATION PERFORMANCE

1.) CLASSIFY



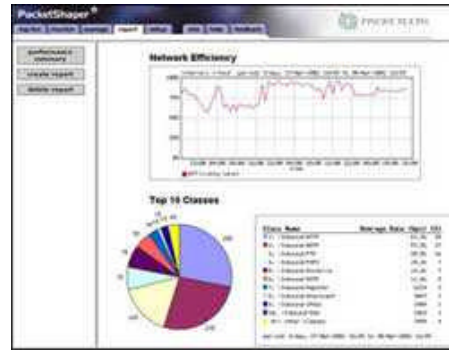
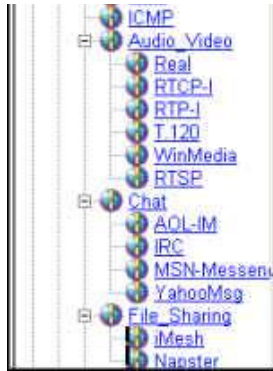
2.) ANALYZE



3.) CONTROL



4.) REPORT



PacketShaper and Your Network: Models and Topologies

-
- [Home](#)
 - [Company](#)
 - [Products](#)
 - [Solutions](#)
 - [Investor](#)
 - [News/Events](#)
 - [Careers](#)
 - [Programs](#)
 - [PartnerWeb](#)
 - [Support](#)

Copyright © 1996-2001 Packeteer, Inc. All rights reserved.