
Facts of Life: The Citizen's Guide to Network Engineering

BY RICHARD BENNETT | FEBRUARY 2011

Supporters of net neutrality fervently believe that the Internet must always consist of dumb pipes.

As we enter the second decade of the controversy over Internet neutrality, the issues stand in sharp focus. While the debate embraces a broad array of issues in innovation theory, technology, economics, and law, one quasi-technical issue dominates the controversy: Supporters of net neutrality fervently believe that the Internet has always consisted of *Dumb Pipes* free of network operator management and that it must continue to be so organized in the future. While many net neutrality advocates consider government regulation of the Internet potentially very harmful – especially in regimes such as China and Iran that do not respect human rights – they generally regard the loss of neutrality as a greater threat that justifies the risks inherent in increased government control. On the other side, advocates of an ever more capable Internet dismiss *Dumb Pipes* as an anachronism that never represented a genuine ideal. In order for the Internet to replace the single-purpose networks of the past – such as the telephone and cable networks – it needs to provide a better level of service than a dumb network can offer. Building intelligence into a network alongside capacity doesn't limit its value to the community; it enhances it, providing it's done correctly.

This is an old dispute. The Dumb Pipes notion was part of a substantial technical debate around Local Area Network (LAN) engineering in the 1980s and 90s. One contingent argued that scarce semiconductor resources should be dedicated to making network chips as fast as possible with no management constraints, and another argued that smart management was more beneficial at the margin than raw speed. This technical debate has become a policy controversy, as we see in the FCC's Open Internet order which essentially mandates Dumb Pipes in its *anti-discrimination* rule.¹

Dumb Pipes also masquerades as a point of philosophy with social implications stretching far beyond mundane engineering concerns. It resonates with Internet buffs, boosters, and activists because it seems to represent the populist ideals of democracy and decentralized decision-making they desire in social and political systems. To claim that societies mimic their technical systems in substantive ways is to endorse folk beliefs in homeopathic magic and the primitive's law of contagion; it is also the basis of net neutralist professor Lawrence Lessig's assertion that *code is law*.²

To claim that societies mimic their technical systems in substantive ways is to endorse folk beliefs in homeopathic magic.

In objective engineering analysis, Dumb Pipes is more nostalgia and factual error than profound insight. While it may once have been productive for hardware designers to choose between raw speed and smart management, it is no longer. Mass-market network interface chips have become faster and cheaper than the finicky, high-end lightwave transceivers that are the true limiting factors in network performance on wireline networks, and mass-market semiconductors are faster than wireless networks. The production of raw bandwidth has become the bottleneck for network economics, so there is no realistic alternative but to make the most of the finite network capacity we have.

It's also the case that applications don't all have the same requirements. In order for broadband networks to be all things to all people, they need the versatility that only can be provided by the addition of smart management to raw bandwidth. This maxim, which holds true for all general-purpose networks, is ultimately dictated by fundamental facts about networks, economics, and applications. The failure to understand these facts has caused the debate over Internet regulation to be much more fractious than it should have been. This insight doesn't end the discussion about network regulation, but it does put the most important issues in perspective: Rather than banning Quality of Service (QoS) differentiation, as many proposed net neutrality bills and regulations have proposed, policy makers must focus on the means of ensuring that differentiation is ubiquitous, productive, and standardized.³

A LITTLE NETWORK HISTORY

"Network Neutrality,"⁴ "Stupid Networks,"⁵ the "End-to-End Network,"⁶ and the latest vogue, "Application-Agnostic Network Management"⁷ are all variations on the Dumb Pipes (DP) concept that dominated debates over Local Area Network (LAN) design in the 1980s. The leading contenders were Ethernet on the "dumb" side and various managed systems (best represented by the IBM Token Ring) on the other. In 1985, Ethernet was a 10 Mbps broadcast network in which all packets from all stations had roughly the same priority, while the IBM Token Ring was a four Mbps round-robin system in which some stations and some applications could assert priority over others. The Token Ring was

The ideal of “Fat Dumb Pipes” can never be achieved because the technology that makes bandwidth available also allows it to be consumed.

designed the way it was in order to provide the best service to time-critical applications (process control and multi-media) and to serve as a building block in an enterprise-wide configuration. In large enterprise networks, packets often needed to cross LAN boundaries, and one use of priorities was to grant higher priority to packets transiting a LAN than to those that started and finished on the same LAN; there are complex mathematical reasons why this is good practice in large networks. Ethernet was a much less ambitious office-oriented system for sharing departmental laser printers and file systems that relied on a control system pioneered by ALOHA, a radio-based wide area network deployed across the Hawaiian Islands in 1971.⁸

The conventional wisdom holds that Ethernet won the LAN wars, but this is a simplification. While we all use a technology today that we call by the name “Ethernet,” today’s local area network is not the Ethernet of our grandfathers. Modern Ethernets are built around a semiconductor-controlled active switch with the capability to prioritize packets in greater detail than either its ancestor or the Token Ring could. It also features an efficient system for isolating failed network members that corrected a shortcoming of both historical Ethernet and Token Ring, and it runs at a range of speeds that were unthinkable in the good old days, up to multiple gigabits per second.

Today’s Ethernet is a synthesis of the best networking ideas of the 1980s plus 25 years of collaborative improvement of a common norm. The adoption of the IEEE 802.3i (10BASE-T) standard for Ethernet over Twisted Pair in 1990 effectively ended the LAN wars in a draw.⁹ Nevertheless, the DP notion that had been a rallying cry of the Ethernet side continues to resonate within the Internet policy community to this day, largely because modern-day DP advocates fail to appreciate the context of the original debate and its conclusion.



Figure 1: 1985 Ethernet cable (left) and 2010 Ethernet cable (photo: Richard Bennett)

The historical issue was not simply a matter of finding the best way to design networks; it was more precisely focused on determining the best way to design them *within the*

limitations of the computer and semiconductor technologies of the 1980s. The over-arching issue was how to make the most of a pool of logic gates that is scarce by today's standards. The Ethernet side argued that devoting all the logic to making packets flow quickly and relatively reliably was better than dividing them between speed and control. The goal of Ethernet design was simply to create a network with so much capacity that no single computer would ever be able to use all of it. If this goal had been achieved (or even were achievable in principle,) questions about application needs would indeed be academic, as networks would always have bandwidth to spare.

But the goal was not achieved, and will never be achieved because the technology that makes bandwidth available also allows it to be consumed. No real-world network can ever have so much capacity that it can never be overloaded. The same technologies that allow network switches to produce bandwidth allow personal computers to consume it. Today's networks not only fail to offer more capacity than a room full of personal computers can consume, they fail to offer more than a *single* well-equipped computer can consume when operating at maximum potential.

TWO FUNDAMENTAL DRAWBACKS OF DUMB PIPES

DP advocates insist that the quality of the Internet experience is overwhelmingly dictated by bandwidth. They therefore insist that network management systems must be volume-based "application-agnostic" systems, effectively blind to diverse application needs and dependent on the false notion that networks can provide more capacity than network devices can consume. DP critics maintain that network applications are sufficiently heterogeneous as to require "application-centric" network management systems tailored to their distinct needs and the realities of network technology.

Dumb Pipes Can Never Be Fat Enough

The application-centric view is correct as a technical matter: When a network can correctly determine application needs, it can allocate resources in such a way as to produce the greatest good for the greatest number. If it can't identify needs, it can only provide a generic service to all comers that would only represent efficient allocation if all applications had the same, generic requirements or if bandwidth were infinite.

In a literal, technical sense, the DP argument is simply wrong: A generic network can never provide the greatest good to the greatest number and bandwidth cannot be infinite. As this is the case, and transparently so, there must be a non-technical rationale for the continuation of the DP notion, and indeed there is.

The basis of the modern DP argument is economic: DP advocates believe that the power to optimize network behavior around application needs is a gateway to harder forms of management aimed at lining network operator pockets to the detriment of the consumer. Therefore, depriving network operators of this power will produce a better outcome for consumers and innovation, non-discriminatory (and preferably flat-rate) pricing. The innovation rationale is asserted in Wu's and van Schewick's books.¹⁰

Dumb Pipes Can Never Be Cheap Enough

The Internet was designed to utilize multiple networking technologies at the same time.

It turns out that the economic rationale for DP is no better than the technical one, however. Just as it's impossible to create a network with more capacity than end systems can use, it's also impossible to build one in which the intrinsic cost of consuming bandwidth is equal to or greater than the cost of supplying it.¹¹ On the margins, bandwidth is produced by hardware and consumed by software; hardware has recurring costs, but software doesn't. The existing installed base of personal computers can offer exponentially more traffic to the Internet core as a simple effect of loading and running new application software.

The vast majority of the resources in today's personal computers and smartphones that could cause bandwidth to be consumed are inactive most of the time. A single personal computer with a Gigabit Ethernet interface is capable of consuming all the bandwidth on the fastest consumer-grade broadband circuit in the world, if continuously active; 100 such computers are jointly capable of saturating the fastest backbone network link, and 10 can saturate the typical path through an Internet Exchange switch.

Given that hundreds of millions of such systems are currently connected to the Internet, the only thing that prevents severe overload today is the fact that popular applications use the Internet much more lightly than they could: A popular new application can change the calculus overnight simply by using a slightly higher portion of the capacity of existing PCs. We've seen this happen many times on the Internet, from the deployment of the Web itself, to Napster, and on to BitTorrent. It's no accident that the highest bandwidth applications relate to the acquisition of "free" content such as pirated movies; when content has a price, the price moderates the appetite for its acquisition, but when it doesn't, network management must do the job of moderating aggressive bandwidth applications. A recent study presented at ITIF found that at least a quarter of Internet traffic consists of infringing uses that can only be controlled by such management.¹²

Good Enough for General Use

Network engineering doesn't strive to produce systems that are infinitely open or totally free, it aims to achieve realistic and attainable goals: A reasonably high degree of utility for a reasonably low cost is generally the path to success in network design. In a literal sense, every computer network of any size experiences saturation billions of times a day, but nobody complains or even notices because networks are unsaturated for even greater periods of time: Each Ethernet packet consumes all of the bandwidth of each backbone link that carries it, but only for a single microsecond. Network users are accustomed to particular experiences of network performance, such as the speed with which a web site loads day after day, or the visual quality of a streamed TV show. The user experience does not depend on absolute bandwidth abundance or lack of cost; it's calibrated to the expected personal experience. Consequently, networks are designed around practical goals, not soaring ideals.

WHAT MAKES THE INTERNET SPECIAL

Without question, the Internet's general approach to network design is a major advance over all previous communication technologies. Before the Internet, networks were designed to support one and only one application: The telegraph, telephone, radio, and TV

networks were all seamlessly integrated, highly cohesive systems designed around single applications.

Application Diversity

The Internet broke from this tradition somewhat by accident: The application that it set out to support – the sharing of remote computing facilities – is intrinsically more variable in its own right than telegraphy or telephony. When a researcher uses a remote computer, he or she interacts with a pair of computer *programs*, not just computer *systems*. Computer programs are diverse by their nature, so taking the support of remote computing as a network design goal creates a mandate for a more flexible network than would be needed for a more narrow application. Thus, the Internet must be different in kind from other networks, not just different in emphasis or degree.

In fact, computer programs are nearly as diverse as the imagination, and their modes of network resource utilization reflect their variability. Some produce a great deal of output from little input, and some do the reverse, summarizing a large data base into a simple chart or graph. Some programs crunch a great deal of data with minimal user interaction, while others constantly interact with the human user. And some programs don't interact with humans at all, doing their business by interacting with other programs, machines, and sensors. A program-to-program network needs to serve the whole range of these behaviors, while the telegraph network simply needed to send and receive telegrams from one operator to another. The Internet was designed to support a single application in theory that turned out to be multiple applications in fact.

Before the Internet, networks were designed to support one and only one application.

Technology Diversity

The Internet is special in another respect as well: It was also designed to utilize multiple networking technologies at the same time. Before the Internet, the application-based networks were all very uniform with respect to technology. Each telegraph network contained the same essential technology, as did each telephone, radio, and TV network. The content of the messages varied, but their formats and the means by which they were delivered – their protocols – were the same from one network to another. More importantly, these discrete networks did not interconnect with each other.

The Internet was designed from the beginning to support the interconnection of a wireline packet-switching network called ARPANET with a terrestrial radio network called PRNET as well as a satellite network that bridged the United States with England, SATNET. So the design of the Internet didn't reach as far "up" into application space as its predecessors or as far "down" into network engineering space. The Internet occupies a happy middle between rapidly advancing applications and evolving networks. As the name implies, it concerns itself with connecting networks to networks, not with applications or fundamental networking issues. Hence, the Internet is more a virtual network than a physical one. It doesn't need to concern itself with physical network problems and can focus instead on devising uniform network interfaces, standard methods of using diverse physical networking technologies.

Structure

Any computer system designed to support diverse applications over diverse network systems must have a modular design. This means that the overall system design will consist of a number of components, beginning with a core set of mandatory elements and continuing through a group of optional application and network elements. In the Internet's case, the only core element is the very simple Internet Protocol, and every other element is optional, including TCP; there are in fact a number of simple process control systems that don't include TCP, including some Internet infrastructure components.

Consequently, the software that makes the Internet work consists of one mandatory module and several optional ones.

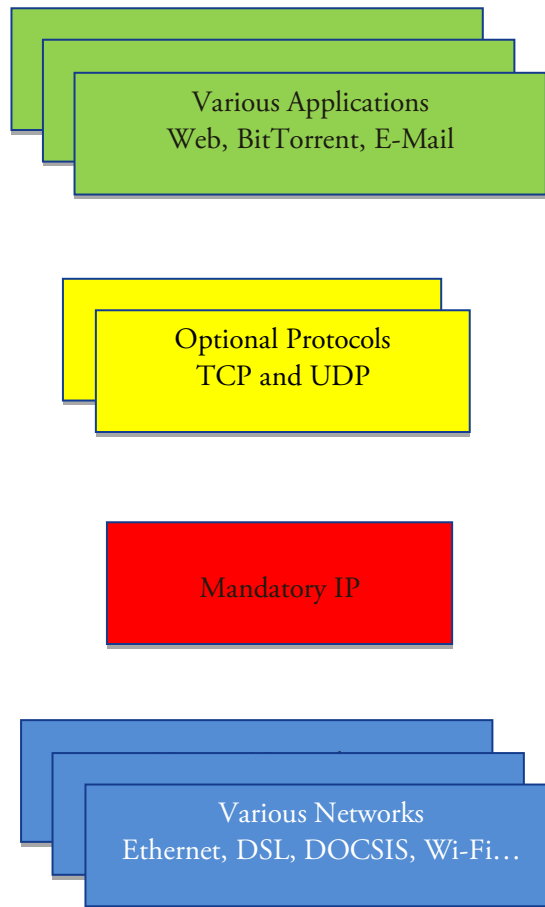


Figure 2: Rough Organization of Internet Protocols

The Internet's software modules actually have complex relationships with each other, such that some modules are dependent on others, and some modules add value to others. Without a network, the Internet Protocol is incapable of delivering a packet. But without IP, one network can't communicate with another network. So IP depends on networks, and at the same time, IP adds value to networks by enabling them to *Internetwork* with each other.

TCP, the function that enables each end-user program to communicate with other end-user programs, depends on IP to cross network boundaries and on networks to transmit packets on physical wires or radios. So TCP depends on IP, and at the same time adds value in the form of a function that permits computer programs to communicate with each other regardless of how many networks need to be crossed and traversed to make this possible. Taking into account the notions of dependency increasing in one direction in the software hierarchy and value increasing in the other, it becomes clear that network software modules naturally organize themselves in vertical fashion. Consequently, network designers

have developed a model of internal network software organization that stacks layers on top of each other.

The best known version of this model is the International Standards Organization's *Open Systems Interconnection Reference Model* (OSI-RM) developed in the late 1970s, composed of seven functional layers. Conventionally, the OSI-RM is represented as a simple vertical arrangement of protocol layers (one on top of the other.) This representation is a simplification, however. As originally conceived, the OSI-RM Reference model acknowledged the special role of network management in real networks. It did this by creating a unique place for a network management application, and by requiring each layer to extend a special interface to the management function. The paper that introduced the OSI-RM contains the following diagram:¹³

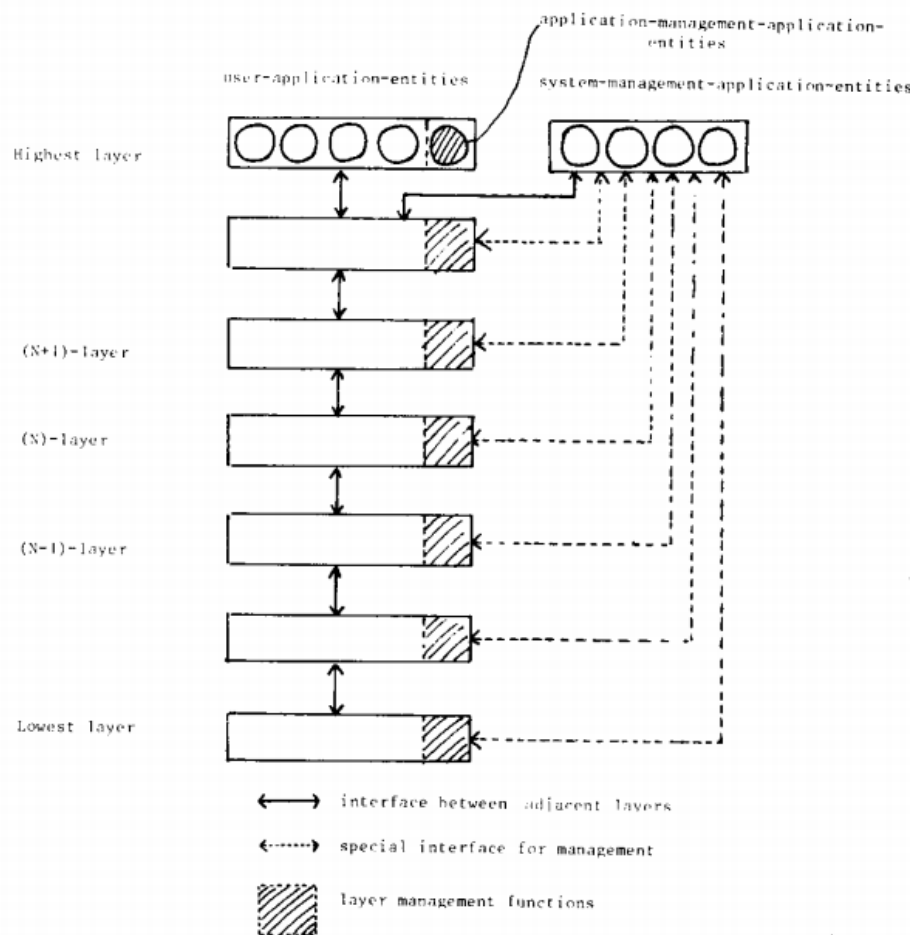


Fig. 12. A representation of management functions.

Figure 3: Classical OSI Reference Model

The diagram is intended to show that each protocol layer is connected not only to its upper and lower neighbor layers, but also to the network management function. The OSI-RM was organized this way in order reinforce a crucial point about network systems: every

network component involved in data transfer also has a role to play in network management. Without network management, functions cannot reliably perform their roles, and unless network management has privileged and direct access to each function, it can't be effective. If network management has to traverse Layers 5 and 6 to query or configure Layer 4, failures in Layer 5 or Layer 6 prevent the communication from taking place, so network management is allowed to bypass the layering arrangement.

While the OSI-RM doesn't strictly describe the structure of the Internet protocols, it's reasonably close. OSI and the Internet protocols are both derived from a common pair of ancestors, ARPANET and CYCLADES.¹⁴

Administrative Diversity

The Internet is also administered in a somewhat different fashion than its predecessors, although this difference is more a matter of degree than of kind. Because of network effects, the value of a network is related to the population that it serves. A network that only reaches the people in one country is less valuable than one that allows users to communicate internationally, for example. For most of the telephone system's history, national monopolies ruled outside the United States and within the U. S. the Bell System held a *de facto* monopoly. Consequently, making an international call required interconnection of discrete networks according to technical and administrative agreements between governments.

The Internet was originally administered by a single agency, the Defense Advanced Research Projects Administration (DARPA,) but over time came to decentralize administration, first to academic consortia such as CSNET and MERIT and ultimately to the competing commercial firms who run the Internet today. Commercial Internet carriers overlap and compete with each other; hence opportunities for interconnection and motivations for changing carriers are more numerous within the Internet ecosystem than they ever were within the transnational scope of the telegraph and telephone systems.

Interconnection Agreements

Interconnection agreements are closely related to network administration. Network operators not only administer their own networks, they manage interconnection agreements with other networks as well. The Internet only works because network operators choose to interconnect, after all.

Rate of Change

Finally, the key technologies that stimulate and support the Internet are parts of an advanced engineering ecosystem that enables more rapid advance than did pre-Internet technologies: The Internet benefits from advances in integrated circuits, fiber optics, radio frequency communication, and software engineering. Each of these pushes the Internet to advance, and enables it to pull new generations of applications out of the lab and into the marketplace. Thus the Internet is rightly described as the "greatest engine for innovation the world has ever seen;" it actively promotes and encourages innovation as no previous network ever did.

WHAT APPLICATIONS NEED

While applications are as varied as the colors of the rainbow, it's possible to create taxonomies of application requirements for network services. The Internet employs a "packet-switching" model of communication in which information is presented in packages known as "datagrams" or "packets." Each datagram contains instructions to the network regarding its destination, lifetime, payload, and desired class of service, and is signed by its originator.

Application Taxonomy

A simple taxonomy based on the most important aspects of packet behavior and treatment tells us all we need to know about network design for application diversity.

1. The first element is *scope*. The packet's destination address identifies a network as well as a network interface card on the host computer or server that the sender desires to contact. Some applications desire only to communicate across a local network, such as the network printing function that's commonly used in home networks. We want our printers to be accessible to people inside our families, but not generally accessible to the outside world, and one convenient way to accomplish this is simply to limit access to our printers to devices inside the home network. Applications that cross oceans have effects on networks that are much more profound than those that are limited to local scope, of course.
2. The second element is *volume*. Some applications, such as e-mail and web browsing, deal in relatively small volumes of information while others, such as video streaming and peer-to-peer (P2P) file sharing, deal with large volumes. The typical e-mail is a few hundred bytes of information, but the typical movie is two or more billion bytes (gigabytes) in size. Applications affect each other principally by the number of packets they present to the network.
3. The third element is desired *latency*. Web browsing is successful if most packets are delivered within two seconds, while VoIP applications such as Skype require their packets to transit the network in less than two tenths of a second. When it takes bit longer to access a web site, the user can sit and wait, but the Skype packet that takes too long must be ignored because the audio stream has moved on and the packet is no longer valuable.
4. The fourth element of application behavior is *accuracy*. All applications want the packets they receive to be correct, but they have different notions of what "correct" means. As noted, a VoIP packets are considered incorrect and discarded if they arrive too late. File transfer programs for the most part require each received packet to be an exact copy of the corresponding sent packet. A program file that isn't received as an exact copy of the original has no value. Applications have different notions of accuracy, related to their tolerance for lost packets.
5. A fifth dimension of application requirements is *performance*, which can be measured in various ways. For file transfer applications, the only element of performance that matters is the arrival time of the last packet. A file transfer is not successful unless the entire file arrives, and the last packet in the file transfer

determines the time that the total transaction takes. File transfer applications have no concern about the arrival rate of individual packets, only the average of all the packets in the file; therefore, if a number of packets arrive close together, in a clump, and another group of packets arrive after a long delay, the file transfer application is unperturbed as long as the average of the clumps and pauses come out to a nice figure. VoIP has a different way of measuring performance, closely related to its insistence on short latency for each packet. VoIP (and other streaming applications) wants packets to arrive at regular intervals with a minimum of variation (called *jitter* by engineers) between them.

6. The final dimension of application diversity is *cost*. Some applications only exist to make a particular service available at lower cost than similar applications from another supplier, so the success of the application must be judged by its cost.

Having isolated the elements of application diversity, we can now see how they're combined in the current group of popular applications on the Internet. This taxonomy will help us explain the claim that networks need to be application-centric.

Some application requirements are at cross-purposes with each other.

APPLICATION	SCOPE	VOLUME	LATENCY	ACCURACY	PERFORMANCE	COST
E-mail	Global	Low	Minutes	Exact	Low	Low
Web Browsing	Global	Medium Low	Seconds	Exact	Medium	Low
P2P File Transfer	Global	High	Minutes to Hours	Exact	Low	Low
Personal Video Streaming	Local	High	Seconds	Tolerant	High	Low
Personal File Backup	Local	High	Minutes to Hours	Exact	Low	Low
VoIP	Friends	Low	200 ms.	Tolerant	High	Medium
Video Conferencing	Friends	High	200 ms.	Varies between audio & video	Very High	High
Gaming	Global	Low	75 ms.	Exact	High	Medium
Tele-robotics	Private	High	50 ms.	Exact	High	High
Remote Sensing	Global	Varies	Varies	Exact	Varies	Medium
Program Update	Global	High	Hours to Days	Exact	Low	Low

Figure 4: Network requirements of common applications.

The differences in application requirements are quite deep. In principle, there are only two ways for a network to satisfy all of the requirements: It can provide a single, indiscriminate service that satisfies the needs of all applications, or it can discriminate or differentiate in such a way that each type of application is handled according to its particular requirements. Given that some of the requirements are apparently at cross-purposes with each other – the combination of high volume and low latency is inconsistent with low cost, for example –

it's unlikely that this approach is tenable in the long run (we'll address why it seems otherwise to some neutral Internet advocates shortly.)

The networking technology that we have today has capacity limits and becomes more expensive in proportion to the quantity of data that it handles and the delay boundaries it observes. While the price per bit of optical networking systems halves every eight months, the electronics that switch packets between optical fibers halve every 18 months, and wireless is on a 30 month cycle. Consequently, volume affects cost today and will for a quite some time. The efficient support of diverse applications on cost-constrained networking technologies argues for differentiated treatment.

Why the Internet Works

At this point, any net neutrality advocates who happen to be reading along are shrieking: "But what about the Internet? It's a *best-effort, end-to-end* network that doesn't discriminate between applications, and look at how well it works! Are you crazy?" This is an honest reaction, and it deserves an explanation.

Indeed, the Internet as a whole doesn't pay attention to applications: Packets whizzing through the backbone networks operated by Global Crossing and Level 3 that connect one Internet Service Provider's "eyeball network" to another aren't sped up or delayed based on the application that generated them. There are a number of reasons for this, however, some good and some not so good. The not-so-good reason is that the modern commercial backbone networks are doing their job the same way that the NSFNET did the same job before the Internet was taken out of government hands and placed in the private sector. Large, cooperative systems are resistant to change, either for the worse or for the better. The phenomenon known as "Internet Ossification" is simply the Internet's resistance to change.¹⁵

The good reason is more subtle and very important. The Internet is a hierarchical system composed of data links or "pipes" of different capacities:

- Each last mile pipe in the DSL portion of the Internet has a capacity of 1 – 50 Mbps downstream and about a fifth of the downstream rate on the upstream side.
- DSL pipes join a switch called a Digital Subscriber Line Access Multiplexer (DSLAM.)
- DSLAMs attach to a higher speed network, typically 155 to 622 Mbps ATM or 1 Gbps Ethernet, to feed another switch known as the Broadband Remote Access Server or Broadband Services Router (BSR).
- BSRs attach to a common Internet edge router using 1 to 10 Gbps Ethernet. The edge router is another kind of switch.
- The edge router joins other backbone networks at a colocation center or Internet Exchange Point (IXP) through another Ethernet switch, commonly running at 10 Gbps.
- Backbone networks typically aggregate multiple 10 Gbps IXP links onto 40 to 100 Gbps Ethernet links over long distance switches (inside the "Internet cloud" in the diagram.)

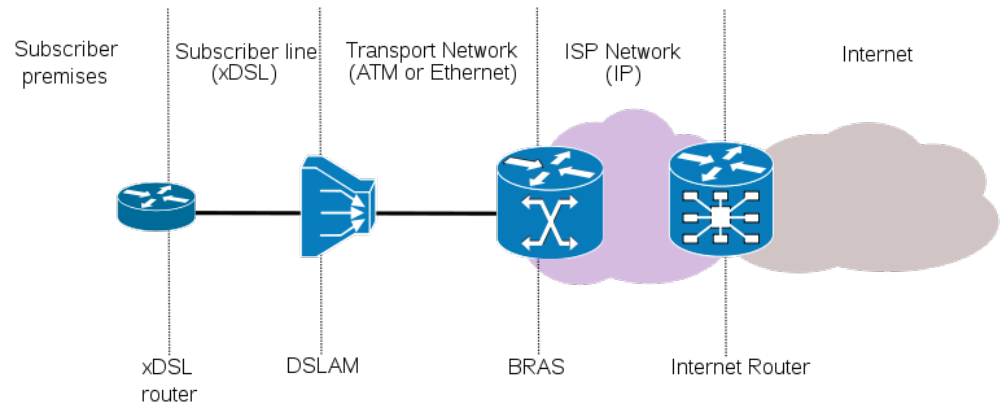


Figure 5: Regional DSL Network elements

Combining multiple inputs onto a common output is known as “multiplexing” in engineering. Each step up in the switching hierarchy from DSL to 100 Gbps Ethernet is a multiplexing operation, and each step down is a de-multiplexing operation. When network architects create multiplexing hierarchies, they have to decide how many inputs to bring into each switch. One way to do this is simply to limit the number of inputs to the capacity of the output: If the switch supports 1 Gbps Ethernet on the input side and 10 Gbps Ethernet on the output side, the designer could limit the number of input ports to 10 and ensure a congestion-free, if costly, network. This approach (“worst-case engineering”) would create the following hierarchy:

1. A 100 Gbps backbone link feeds 10 IXP ports.
2. Each IXP port feeds 10 BRAS’s at 1 Gbps.
3. Each BSR feeds six DSLAMs at 155 Mbps.
4. Each DSLAM feeds 60 subscribers at 2.5 Mbps (upstream.)

In this scenario, each subscriber would be guaranteed to avoid congestion on the way up, all the way from the DSL modem to the backbone (except for the congestion created by joining the in-home Ethernet to the much slower DSL connection, that is; more on that later.) So is this Nirvana, a network that’s totally free of congestion, apart from the local congestion that the user can control by tweaking the settings in the home router?

As it turns out, the answer is “no” because we haven’t done all the work yet. It’s not enough to simply upload packets to the backbone; we also need to download them from the backbone to the home router, and this is where our picture gets complicated. When we’re combining a number of lower speed data pipes into a higher speed pipe, the arithmetic is simple: we simply sum the inputs to get the capacity of the congestion-free output. The arithmetic for de-multiplexing is not as easy as dividing the higher capacity input by the capacity of the outputs because we have to account for the particular traffic flow to each of the outputs, which is determined by the applications that are running.

Statistical Modeling

It is almost never the case that all users of a multi-user network will load the network equally at any given time. We come closer to average use over long periods of time such as

The Internet has the ability to adapt to temporary fluctuations in load, but the deployed infrastructure has finite capacity.

months or years, but on a second-by-second basis the differences are quite dramatic. People who are loading web pages generate a great deal of traffic in half-second bursts of 300 kilobytes, VoIP users generate steady stream of 30 Kbps, video streamers consume 2 Mbps at standard definition and 8 Mbps at high definition, and most people do nothing at all with the network during any given second (they're either reading web pages or e-mails or they're away from their computers.) Networks need to function well on sub-second intervals because of the way the applications they support work; as Figure 4 illustrates, some applications can be affected by delays of tenths of a second or less.

When a 10 Gbps pipe empties into 10 one Gbps pipes, congestion occurs when more than tenth of the traffic capacity of the fat pipe must empty into any *single* skinny pipe. At any given second, the entire capacity of the fat pipe may be addressed to a single skinny pipe, it may be evenly distributed or anywhere in between, and it may even be empty. The only sure way around the possibility of congestion would be to ensure that every download connection is just as large the one that feeds it (potentially as large as 100 Gbps with present technology.) This would be absurdly expensive, raising the cost of the home router from its current \$100 level to \$200,000 or more. Consequently, network operators create statistical models of traffic flow and design their networks from predictions about acceptable levels of temporary congestion. This practice originated in the telephone networks with so-called Erlang Models.¹⁶ In fact, the Internet's genius comes directly from the fact that its underlying technology – packet-switching – allows networks to be designed statistically. Statistical networks can assign and reassign bandwidth-on-demand, cheaply and efficiently. The telephone, in contrast, allocates the same 3 kilohertz bundle of bandwidth to each caller whether they're talking, listening, sending a fax, or downloading a file. In practice, broadband networks employ statistical design throughout; it's good for everyone.

Consequently, the networks that comprise the Internet are provisioned according to predictions about:

1. The amount of network traffic users will generate
2. Traffic distribution across routes
3. The relative weights of upstream and downstream traffic

All of these are ultimately predictions about application behavior. The Internet has the ability to adapt to temporary fluctuations in load, but the deployed infrastructure has finite capacity.

User behavior is difficult to predict because it's driven in part by forces external to networks, ranging from licensing to economic conditions to the whims of fashion. When the FCC investigated Comcast's treatment of P2P applications in 2008, many pundits declared that P2P was essential to the Internet's future, but P2P use as a percent of overall Internet use has declined since then; streaming is the new darling.¹⁷ A network engineer who adapted bandwidth allocation from asymmetrical to symmetrical in order to serve P2P better in 2009 missed the boat on the rise of asymmetrical streaming applications such as Netflix in 2010 and disadvantaged his network.

Proper provisioning is important, but it's a long term activity; management has to deal with short term network behavior. Net neutralists often get the balance of these activities wrong, and seek to resolve problems by over-provisioning that are more properly handled by more effective use of actual, existing networks.

Application Profiling

We're now in a position to understand why the Internet can support as many applications as it does without actively differentiating among them: The secret is application profiling. Network technologies are devised and network provisioning decisions are made on the basis of the known characteristics of popular applications and predictions about their use. The most popular application on the Internet as long as it's been a privately owned system is web browsing. The characteristics of web browsing are well understood:

- Uses TCP rather than UDP
- Uses 4 – 16 Virtual Circuits at a time
- Downloads much more data than it uploads
- Traffic consists of bursts rather than steady streams.
- Each burst is about 300 Kilobytes

All the network operator has to do in order to ensure happy customers is provision the network around these characteristics and the Web hegemony will do the rest; popular non-web applications such as e-mail are migrating to the Web, after all. Internet users probably don't even appreciate the fact that the Web is only part of the Internet.

The Web has an important unintended characteristic that makes it an ideal application to build a network around: It leaves a great deal of bandwidth on the table. It does this in part because of its "bursty" nature: We load web pages and then we read them, and while we're reading we're not loading the network. But even when a Web page is loading, a great deal of available bandwidth goes unused because of the nature of Jacobson's Algorithm, the fundamental element of the Internet's end-to-end congestion control system.¹⁸

The Web uses a number of TCP Virtual Circuits, each of which begins in the "Slow Start" condition that requires it to stop and wait a number of times while transmitting its initial series of packets. While the Web waits for packets that TCP artificially delays, other applications are free to use the Internet (even other Web sessions.) Tweaking the Internet to make the Web work well also allows it to support other applications with modest network requirements, such as narrowband VoIP. Neutralists who insist that the Internet works well without management are implicitly promoting the benefits of application profiling, albeit of a limited sort.

HOW THE INTERNET NEEDS TO CHANGE

The Internet is tuned for the Web, an application that leaves bandwidth on the table, and consequently has been able to get by for the last 15 years with very little active traffic management. This set of circumstances, which many observers regard as fortuitous, has come to represent a barrier to further application innovation. Most of the innovation that we've seen in recent years has come in the form of Web applications. The 10 most popular

web sites would lead most lists of the Internet's leading innovations since the creation of the web itself:

1. Google
2. Facebook
3. YouTube
4. Yahoo
5. Live
6. Baidu
7. Wikipedia
8. Blogspot
9. Tencent
10. Twitter

Figure 6: Ten Most Popular Websites (source: <http://mostpopularwebsites.net/>)

The emergence of media streaming among the Internet's most popular Web sites suggests that it's on its way to usurping the Web and becoming the Internet's dominant application.

Challenges to Web Hegemony

The leading non-web application innovations, VoIP and Peer-to-Peer, rely on the bandwidth that the Web leaves on the table: VoIP for narrow-band, bi-directional communication, and P2P for file sharing with emphasis on the use of upstream resources the Web uses sparingly.

Media streaming is a special category. YouTube, Netflix, and to a lesser extent Facebook, are media streaming applications with a Web-based user interface. One of the many virtues of the Web is its ability to serve as a portal for streaming media applications by providing search and other user interface functions, and then to allow arbitrary types of media to permeate the user interface membrane and function end-to-end without the Web server's involvement. When we select a movie to watch from the Netflix web site, Netflix establishes a session between the Netflix player on a consumer-owned machine and a video server on at a nearby Internet exchange or colocation center. The actual streaming process bypasses the Web, and indeed bypasses much of the Internet itself.

The emergence of media streaming among the Internet's most popular Web sites suggests that it's on its way to usurping the Web and becoming the Internet's dominant application. Cisco's traffic study predicts that video will exceed 91 percent of global consumer traffic on the Internet by 2014.¹⁹ Not only has media streaming replaced the Web as the dominant form of Internet traffic, integrated applications are replacing the Web as the user interface of choice on mobile devices.²⁰ The decline of the Web isn't so much a movement away from "openness" as it is a pragmatic rejection of the Web browser itself as a user interface (UI.)

The Web UI presumes that the browser will run on a machine with the set of user interaction devices that Tim Berners-Lee had on the NeXT workstation on which he wrote his first browser: a large screen, a keyboard, and a mouse, but touch-screen devices have none of these features natively, and can only simulate them with limited success. The Web's user interface programming language, HTML, includes operations such as "on mouseover" that can't be correctly simulated on the capacitive-touch screens found on the

Apple iPad and similar devices at all; Web-based user interface programming tools such as Adobe Flash are essentially unusable on such devices in their present form. The next version of HTML, HTML 5, is meant to correct these issues, which it does by performing functions that were formerly provided to Web browsers by tools such as Flash. HTML 5 also enables application developers to bypass the browser and interact with the user more directly. The Internet is entering a post-Web era in terms of content and user interaction.

Net Neutrality Implications of the Post-Web Era

Most net neutrality advocates express concern about the new forms of UI construction and the policies of the device-maker app stores that mediate applications, but they recognize that regulators have little jurisdiction over them. Media streaming is directly within the scope of network regulation, however, as it directly pertains to network facilities and network operator terms of use.

Streaming media brings new “billable events” to the Internet in the form of movies, albums, and TV programs that the user pays to experience. Most billable events on the traditional Web are advertisements. The Web user is not directly involved in the exchange of money between advertisers and Web sites, and therefore has no particular expectation of any standard of network performance for advertising-supported content. Some days, YouTube video plays well, and other days it doesn’t; few users call YouTube support to complain on the days when it doesn’t because they always get what they pay YouTube for.

Advertisements are most effective on large-screen devices with free space for ads around their main content pane and a simple way to dismiss intrusive pop-ups and temporary redirections. Small-screen and hands-free devices are crippled by ads and therefore application developers targeting the mobile space tend to rely on software sales and service subscriptions to monetize their efforts. “*Information Wants to be Free*” proponents see subscriptions as abomination, but have little hope of regulatory action to prevent such transactions from taking place between consenting adults.

When content viewing becomes a billable event, users have an expectation of quality they didn’t have before, and as content shifts from text-based Web pages to streaming media, the statistical Internet has a harder time performing to expectations. The web page is an event of short duration, and if it fails to load instantly, it generally loads eventually with no loss of quality or corruption of content. We can’t say these things about media streaming, an event of long duration in which packets can’t be lost and recovered repeatedly without affecting the user experience. Unlike fleeting Web pages, each of which is more or less equivalent to hundreds of others, movies lock the viewer into a plot and can’t be abandoned mid-session very easily; we want to see the crime solved once we’ve seen it performed.

The Content Delivery Network

Providers of streaming media programs deal with the issues of quality and reliability by giving their programs priority over most Web traffic. The simple means of obtaining priority treatment from the Internet is to:

1. Locate content close to the user.
2. Moderate the rate at which packets are released onto the Internet.

Locating the content close to the user is accomplished by purchasing a Content Delivery Network (CDN) service. CDNs lease space at Internet Exchange Points (IXP,) colocation facilities (colos,) or at nearby private locations with high-speed connections. Reducing the distance between the CDN and the end user enables the CDN server to out-compete far away web sites (or content distributors) that are required to engage in the stop-and-wait behavior of Jacobson's Algorithm previously described. All other things being equal, the performance of TCP streams is determined by speed, packet loss rate, and round-trip time between sender and receiver.²¹ Using a CDN optimizes all three factors; in fact, CDNs have a multiplier effect over traditional Internet delivery in terms of quality.²²

TCP lacks the ability to use all available bandwidth on a given pipe because it engages in a never-ending hunt for more capacity; per Jacobson's Algorithm, it increases the transmit rate until it loses a packet, cuts the rate in half, and then begins again. This behavior is unproductive for media streaming, an application that seeks the lesser of the rate required by the content or the rate the pipe can reliably deliver; streaming is affected by packet loss and wants to minimize it. Therefore, media streaming does not typically employ TCP, relying instead on customized transport protocols over the Internet's standard User Datagram Protocol (UDP,) a sort of "un-protocol" that's carried by IP directly (Netflix is the exception, as it uses TCP in combination with the web protocol, HTTP.)

Telepresence is a class of application that features two-way interaction at a distance.

Media streaming protocols moderate the load they present to the network by altering the degree of compression in the programs they transmit. In the case of Netflix, this is accomplished by encoding programs at multiple levels of compression and selecting the best encoding to current network conditions. Netflix encodings vary in their bandwidth requirements from 375 kilobits/sec to 3.8 megabits/sec.²³ Netflix measures the pipe's capacity, chooses an encoding that requires no more than 60% of measured capacity, and then makes adjustments as the program plays. The Netflix behavior is dynamic, but less so than TCP's behavior.

The Rise of Telepresence

While video streaming puts significantly more traffic on the Internet, it's not a radically new way of using the Internet. Like web browsing, video streaming is a one-way application: Web surfers download much more data than they upload, and so do Netflix users; web sites can be accelerated by CDNs, and so can video streaming. CDNs prevent the web and video streaming services from overloading the Internet backbone, so both applications can grow indefinitely without major new investments in the Internet backbone. Telepresence is very different.

Telepresence is the name for a class of application that features two-way interaction at a distance. The classical example of this kind of interaction is video conferencing, but there are other examples as well: Telerobotics in general and telesurgery in particular are telepresence applications, as is remote interactive gaming. These applications go beyond conferencing as they aren't limited to speaking, listening, and viewing; they involve concrete action at a distance.

Telepresence applications are the hallmark of the Next Generation Broadband network, the most important and compelling argument for provisioning high speed, 50+ megabit/second networks everywhere and steadily increasing the performance of all network connections.

Telepresence has significant effects on the Internet backbone and can't be accelerated by CDNs. The telepresence load on backbone networks is already significant, but it often goes unnoticed because of the way it's handled. Firms that use telepresence extensively within their own walls – such as Cisco, one of the leading suppliers of advanced video conferencing equipment – tend to build specialized backbones dedicated to telepresence that are segregated from the backbones that support general Internet use. The neutralists' fear of a fragmented Internet is nowhere more real than it is among heavy telepresence users.

Handling the Telepresence Load

It's reasonable to ask why telepresence users don't just increase their general-purpose backbone capacity instead of creating specialized backbones for telepresence. After all, higher capacity backbones should allow greater Internet performance and telepresence at the same time, or something close to the same time. Why can't the telepresence problem be solved by fatter Dumb Pipes?

To understand why Dumb Pipes don't solve the telepresence problem we need to explain the claim made above to the effect that *Dumb Pipes Can Never Be Fat Enough*. The principal problem has to do with the way that the dumb networks allocate bandwidth. When there's no centralized (or semi-centralized) bandwidth manager, it's up to network endpoints – applications – to compete with each other for bandwidth. The demand for Internet bandwidth at the endpoints always exceeds the supply. The Internet consists of 530 million computers today, most of which have the ability to transmit and receive at 1 Gbps.²⁴ While only a small fraction of the computers that could be active on a given part of the Internet are active at any given time, it wouldn't take many 1 Gbps transmitters to overload the fattest pipe in the Internet backbone, which only has the capacity to carry 100 Gbps today. The most common “fat pipes” in use today are capable of providing only 10 Gbps, enough to serve no more than 10 computers doing peer-to-peer file transfers at their peak rate.

Internet applications have two styles of bandwidth utilization: A self-limited style and a network-limited style. Network-limited file transfer applications (such as traditional ftp and the newer P2P apps such as BitTorrent) are designed to use as much bandwidth as the network has available in order to complete their business rapidly. P2P applications seek bandwidth aggressively. Self-limited applications, such as VoIP, have a distinct upper limit on the bandwidth they can use at any given time; for VoIP applications like Vonage that connect to the Public Switched Telephone Network (PSTN,) that upper limit is 64 Kbps.

In a perfectly-functioning dumb network, each application gets roughly the same proportion or fraction of the bandwidth that it seeks: If one application seeks 1 Gbps (because it's network-limited instead of self-limited) and another seeks 64 Kbps, these networks are judged “fair” if each application gets 50% of its sought-after allocation, for

Telepresence has significant effects on the Internet backbone and can't be accelerated by CDNs.

Application diversity is a vital and necessary goal that can only be achieved by application-centric management and pricing practices.

example. So the dumb network is doing its job if it provides a bandwidth-hungry P2P application with 500 Mbps and a self-limiting VoIP application with 32 Kbps. While some argue that this state of affairs is ideal, it doesn't advance the cause of application diversity.

WHY DUMB PIPES GETS IT WRONG

The model of network management proposed by net neutrality advocates is both internally inconsistent and factually deficient. Proponents of this view argue that it's permissible for Internet users to actively shape traffic or otherwise manage their own network use according to any criteria they wish to employ, but it's decidedly not permissible for network operators to perform similar tasks on their behalf.²⁵ Application agnosticism (AA) is actually harmful to innovation: Robbing the Internet of the ability to serve emerging applications effectively makes it less "open," not more. The Internet's openness, its value to innovators in particular and to liberal democracy generally, is greatly improved by the deployment of refined systems of management and economics operating under appropriate, technically-and economically-aware, regulatory oversight.

Application diversity is a vital and necessary goal that can only be achieved by application-centric (AC) management and pricing practices. Ideally, these practices should be end user-controlled, but taking into account the fact that most Internet users have limited technical knowledge, the option for the user to voluntarily cede management of application management to network operators is a beneficial alternative that must be allowed.

AA proponents²⁶ argue that the Internet's capacity to stimulate innovation is a consequence of information hiding. According to their analysis, the Internet partitions information about the network from information about applications according to an engineering principle known as the *end-to-end principle* or E2E. E2E originally pertained to the placement of functions in distributed computer systems, and may be applied by engineers (if it is applied at all) with varying degrees of dogmatism. Lawrence Lessig asserts that E2E dictates that network operators must behave like "day-dreaming postal workers" moving mail in a sorting room with no regard for necessary class of service, content, or source.²⁷

In this view, ignorance is bliss: When networks are blind to applications, network operators can't interfere with novel developments hostile to their economic interests. Proponents of this view fear that if networks partake of the fruit of the tree of application knowledge, operators will be tempted to stifle innovation in order to preserve privileged positions in the Internet ecosystem and market forces are powerless to stop them. These ills can only be avoided by hiding information about application behavior from network operators. Therefore, neutralists urge regulators to cloak Internet applications in cyber-hijabs so as not to over-stimulate inherently intemperate operators. Just as the Taliban believe that "the face of a woman is a source of corruption,"²⁸ neutralists believe that network operators are corrupted by application knowledge.

Lessig's "Code is Law" formulation holds that the inventors of the Internet foresaw the possibility of network operator interference and designed it out of the Internet, essentially installing a wall of separation between networks and applications. The original architecture created an Eden free of tempting serpents, so retaining it precludes the Fall of the Internet.

Neutralists admit that application knowledge has potential benefits, but don't consider them sufficient to justify its intrinsic risk of degrading user freedom. They believe that the Internet has done magnificently well without application knowledge, achieving a condition in which users and applications harmonize their needs in a perfectly efficient and subtly coordinated "emergent order" brought about by no means other than the withering away of centralized control. Another notable neutralist, David Isenberg, declares the Internet's "stupidity" to be its greatest virtue.

Lessig's "Code is Law" program is part of a larger critique of the inability of law to shape human behavior before the fact. Lessig would rather not limit the law to punishing bad behavior, he seeks to "baby-proof the world" against the temptation to misbehave. A strong Utopian impulse leads him to read things into the Internet's design that have only rarely been seen by skilled practitioners of network engineering: There is no real barrier between the application information in Internet Protocol datagrams and network information; this is why Deep Packet Inspection is possible in the first place.

Lessig's Internet pronouncements have a fervent, religious character that makes engineers uncomfortable and fails to satisfy the religious. Lessig has been described as "a prophet for the Internet age," a fact that he proudly proclaims on a personal web site dedicated to reviews of one of his books.²⁹ He's also given to describing those he regards as kindred spirits in religious terms:³⁰

...as with Moses, it was another leader, Linus Torvalds, who finally carried the [Open Source] movement into the promised land by facilitating the development of the final part of the OS puzzle. Like Moses, too, [Richard] Stallman is both respected and reviled by allies within the movement. He is [an] unforgiving, and hence for many inspiring, leader of a critically important aspect of modern culture. I have deep respect for the principle and commitment of this extraordinary individual, though I also have great respect for those who are courageous enough to question his thinking and then sustain his wrath.

Even Lessig's admirers concede that his writing employs less of the sober style and substance of policy analysis than of the fire and brimstone of revivalist preaching:³¹

*Lessig's skilled melding of these evocative modes of writing ought, at the very least, to give us pause. Arguments cannot be separated from the way they are made: the form of Lessig's treatise is inseparable from its substance. Part manifesto, part jeremiad, *The Future of Ideas* delivers the formulaic punch of both.*

The manifesto and the jeremiad are emotion-driven genres that get much of their power and sweep from rhetorical pyrotechnics. Built on overstatement, oversimplification, and a blithe refusal to acknowledge that there are always alternative points of view, neither the manifesto nor the jeremiad has room for ambiguity. One-sidedness is their nature: Marx's manifesto would not have been a manifesto if he had given capitalism its due; the Puritan ministers could not have put the fear of God in their congregations if they confessed to doubt, or admitted that there was more than one way to read the Bible. On the basis of this one-sidedness, manifestos and jeremiads predict the future. Amid

scathing indictments of our moral slackness and gloomy forecasts of our impending doom, they show us how, if only we change our ways – If we overthrow capitalism, say, or devote ourselves to God, or wrest control of the Internet away from the powers that be – we will not only avert disaster but will create an ideal world, a communist utopia for example, or an eternal paradise, or a thriving, truly democratic culture.

Americans have always loved fire and brimstone preaching.

This kind of advocacy seems to have more to do with inflaming an audience than with reaching the policy formulation most likely to stimulate innovation and democratic values. It's not a serious technical or economic analysis in any case.

GETTING INTERNET REGULATION RIGHT

While AA advocates tend to employ appeals to emotion, myth, and anecdote to make their case, AC advocates typically employ much less titillating technical analysis. While myth is extremely accessible, it's often wrong. Technical analysis is more likely to be correct, but it's difficult for the average citizen to follow and much less entertaining than hyperbolic revivalism, so the AA position tends to be portrayed in the media much more sympathetically than it should be. The debate over Internet regulation has not always served to increase public understanding of the Internet and of networks generally.

The first step for Internet regulators is to read and understand the RFCs, and the second is to defer to them in the initial formulation of any system of regulation. Creating the correct regulatory framework for emerging technologies is always a challenge because regulation depends on enforcing norms of behavior, while technical innovation seeks to alter norms and enable new forms of behavior. The Internet is an open, standards-based system, however, that consists of systems and protocols that conform to published agreements about network behavior and operator conduct. The Internet's design documents, known as Requests for Comment (RFCs) are published on-line by the RFC Editor, a function supported by the Internet Society.³²

The Requests for Comments (RFCs) form a series of notes, started in 1969, about the Internet (originally the ARPANET). The notes discuss many aspects of computer communication, focusing on networking protocols, procedures, programs, and concepts but also including meeting notes, opinion, and sometimes humor. For more information on the history of the RFC series, see ["30 Years of RFCs"](#). The early RFCs include a trove of history about the early development of computer communication protocols, from which modern Internet technology was derived.

The first step for Internet regulators is to read and understand the RFCs, and the second is to defer to them in the initial formulation of any system of regulation. There is a great deal of disagreement among policy advocates regarding the Internet's fundamental principles, and no source is more definitive regarding not only the fundamentals but the details than the RFCs.

The first step for Internet regulators is to read and understand the RFCs, and the second is to defer to them in the initial formulation of any system of regulation.

The recent debate about a practice that's been called "paid prioritization" by its enemies can largely be settled by checking the numerous Internet RFCs about Quality of Service such as RFC 2475, *An Architecture for Differentiated Services*, which says:³³

This document defines an architecture for implementing scalable service differentiation in the Internet. A "Service" defines some significant characteristics of packet transmission in one direction across a set of one or more paths within a network. These characteristics may be specified in quantitative or statistical terms of throughput, delay, jitter, and/or loss, or may otherwise be specified in terms of some relative priority of access to network resources. Service differentiation is desired to accommodate heterogeneous application requirements and user expectations, and to permit differentiated pricing of Internet service.

Given that the RFC says in plain language that there is a consensus to the effect that Differentiated Services (a form of Quality of Service) is desirable in order *to accommodate heterogeneous application requirements* and *to permit differentiated pricing of Internet service*, regulators who wish to impose an alternate view upon the Internet have a steep hill to climb. They should at least elaborate detailed reasoning to support personal views clearly at odds with the norms published as consensus by the Internet engineering community. Such deference is not absolute, of course. Technical standards are neither meant to have force of law nor to represent inviolate standards of conduct. In fact, technical standards are subject to obsolescence and replacement, and aren't always consistent with each other.

Technical standards are also subject to mediation by economics and other policy considerations. The RFC process doesn't set prices for example, nor does it dictate systems for billing and measuring economic events. Consequently, regulatory bodies that impose price controls on the Internet don't violate Internet standards, even though such actions may be dubious for other reasons.

Internet regulators must also defer to national purposes regarding the Internet. This is especially important when questions arise about funding and the allocation of such common resources as spectrum and rights of way. Internet standards are not self-executing; just as the Internet Protocol cannot send packets without the help of a physical network, the Internet's infrastructure neither builds itself nor finances itself. If the taxpayers are called to help finance networks for educational purposes, for example, it's reasonable for policy makers to give preference to the technical standards that promote educational purposes, whatever they may be.

Finally, regulators must realize that the Internet is only a portion of a national or international system of digital communication, not the entire story. The Internet depends on physical networks of various kinds (such as Ethernet, DSL, Passive Optical Networking, and 3GPP mobile broadband,) each with its own engineering facts and limitations. The inclusion of "reasonable network management" language in Internet regulation frameworks recognizes these facts.

But in the first instance, regulators should recognize that they depart from Internet norms at their peril.

CONCLUSION

The Internet emerged at a historical moment when network engineering was much less advanced than it is today. It has remained relevant (and become dominant) by incorporating technical advances as they've been developed, largely through an RFC process that records the evolving consensus about the Internet's structure and operation.

The Internet's dynamic process of self-modification is the key to its longevity and utility. It would be a supreme error for any regulatory body to insist that the Internet should be declared a finished system forbidden from further improvement. The first generation of Internet engineers had remarkable insight, but the current generation has a greater store of accumulated knowledge and better design tools. Consequently, the Internet standards of today are better than those produced in the 1970s when global digital networking was in its infancy.

The spirit of the Internet is the drive for continuous improvement. This is the means by which networks become faster and cheaper, new applications are enabled, and digital quality of life is improved. Today's Internet is more complex and refined than was the Internet of the 1970s, and with any luck, the Internet will continue to improve. Regulatory oversight is valuable, but it should not be overly prescriptive, rooted too firmly in the past, or shackled to myth and dogma.

The Internet's large caretaker and stakeholder community should always be the first line of defense against any effort on the part of commercial or governmental interests to set it on the wrong course. Government regulators should always take a back seat to the Internet's technical community, taking action only as and when genuine problems present themselves that can't be resolved through the Internet's organic, consensus processes. Such instances are extremely rare.

ENDNOTES

- ¹ Federal Communications Commission, *Preserving the Open Internet*, 2010, http://www.fcc.gov/Daily_Releases/Daily_Business/2010/db1223/FCC-10-201A1.pdf.
- ² James Frazer, *The Golden Bough: A Study in Magic and Religion*, 1st ed. (New York: Simon & Schuster, 1996); Lawrence Lessig, *Code and Other Laws of Cyberspace* (New York: Basic Books, 1999).
- ³ Senator Maria Cantwell and Al Franken, *Internet Freedom, Broadband Promotion, and Consumer Protection Act of 2011*, 2011.
- ⁴ Tim Wu, “Network Neutrality, Broadband Discrimination,” *SSRN Electronic Journal* (2003), <http://www.ssrn.com/abstract=388863>.
- ⁵ David Isenberg, “The Rise of the Stupid Network,” *Computer Telephony* (August 1997): 16-24.
- ⁶ Lawrence Lessig, *The Future of Ideas : the Fate of the Commons in a Connected World*, 1st ed. (New York: Random House, 2001).
- ⁷ Barbara van Schewick, *Internet Architecture and Innovation* (Cambridge MA: The MIT Press, 2010).
- ⁸ “ALOHA.net,” in *Wikipedia*, n.d., <http://en.wikipedia.org/wiki/ALOHA.net>.
- ⁹ “IEEE 802.3,” in *Wikipedia*, n.d., http://en.wikipedia.org/wiki/IEEE_802.3.
- ¹⁰ Tim Wu, *The Master Switch: The Rise and Fall of Information Empires*, 1st ed. (New York: Alfred A. Knopf, 2010); van Schewick, *Internet Architecture and Innovation*.
- ¹¹ For present purposes, we will assume that bandwidth represents all network resources even though it doesn’t.
- ¹² David Price, “How Much Bandwidth Is Used For Online Piracy?” (presented at the Information Technology & Innovation Foundation, Washington, DC, January 31, 2011), <http://itif.org/events/how-much-bandwidth-used-online-piracy>.
- ¹³ Hubert Zimmerman, “OSI Reference Model - The ISO Model of Architecture for Open Systems Interconnection,” *IEEE Transactions on Communications* Com-28, no. 4 (April 1980): 425-432.
- ¹⁴ Richard Bennett, *Designed for Change: End-to-End Arguments, Internet Innovation, and the Net Neutrality Debate* (Washington, DC: Information Technology and Innovation Foundation, September 2009), <http://www.itif.org/index.php?id=294>.
- ¹⁵ Mark Handley, “Why the Internet only just works,” *BT Technology Journal* 24, no. 3 (July 2006): 119-129.
- ¹⁶ Erlang models may be used to allocate resources of many kinds; see: Mateo Restrepo, “Erlang Loss Models for the Static Deployment of Ambulances” (Cornell University, April 15, 2007), http://people.orie.cornell.edu/~huseyin/publications/erlang_ems.pdf.
- ¹⁷ “P2P as a percentage of consumer Internet traffic will drop to 17 percent of consumer Internet traffic by 2014, down from 39 percent at the end of 2009,” “Cisco Visual Networking Index: Forecast and Methodology, 2009–2014” (Cisco Systems, June 2, 2010), http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360.pdf; Price, “How Much Bandwidth Is Used For Online Piracy?”
- ¹⁸ Van Jacobson, “Congestion Avoidance and Control,” *Computer Communication Review* 25, no. 1, ACM Special Interest Group on Data Communication (1995): 157.
- ¹⁹ “Cisco Visual Networking Index: Forecast and Methodology, 2009–2014.”
- ²⁰ Chris Anderson and Michael Wolf, “The Web Is Dead. Long Live the Internet,” *Wired Magazine*, August 17, 2010, http://www.wired.com/magazine/2010/08/ff_webrip/all/1.
- ²¹ Jitendra Padhye et al., “Modeling TCP Throughput: A Simple Model and its Empirical Validation” (University of Massachusetts, May 30, 1998), <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.143.9137&rep=rep1&type=pdf>.
- ²² Prasad Bagal, Shivkumar Kalyanaraman, and Bob Packer, “Comparative Study of RED, ECN, and TCP Rate Control” (Global Internet '99 Symposium, 1999), <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.143.5526&rep=rep1&type=pdf>.
- ²³ Neil Hunt, “Encoding for Streaming,” company blog, *The Netflix Blog*, November 6, 2008, <http://blog.netflix.com/2008/11/encoding-for-streaming.html>.
- ²⁴ *The State of the Internet* (Akamai, Inc., Third Quarter 2010), page 5.
- ²⁵ van Schewick, *Internet Architecture and Innovation*.

-
- ²⁶ The principal supporters of application-agnostic network management include Harvard law Professor Lawrence Lessig, Lessig disciples Tim Wu and Barbara van Schewick, legal scholars affiliated with the Harvard Berkman Center for Internet and Society, much of Washington's self-styled public interest lobby, and venture capitalists heavily invested in traditional Internet properties such as Fred Wilson and Brad Burnham, although Lessig and Wu have made contradictory statements in support of application-oriented management practices on occasion.
- ²⁷ Lessig, *Code and Other Laws of Cyberspace*.
- ²⁸ M. J. Gohan, *The Taliban: Ascent to Power* (Oxford University Press, 2000), pp. 108-110.
- ²⁹ Lawrence Lessig, "Reviews - Remix: Making Art and Commerce Thrive in the Hybrid Economy," personal website, n.d., <http://remix.lessig.org/reviews.php>.
- ³⁰ Lessig, *The Future of Ideas : the Fate of the Commons in a Connected World*.
- ³¹ "Lawrence Lessig's Messianic Manifesto: A Doomsday Look at Cyberspace," Law and Public Policy, *Knowledge@Wharton*, March 27, 2002, <http://knowledge.wharton.upenn.edu/article.cfm?articleid=533>.
- ³² "RFC Editor Overview," *RFC Editor*, n.d., <http://www.rfc-editor.org/overview.html>.
- ³³ S. Blake et al., "RFC 2475 - An Architecture for Differentiated Services," Internet RFC, December 1998, <http://tools.ietf.org/rfc/rfc2475.txt>.