1 of 2 DOCUMENTS

Harvard Law Review

May, 2006

119 Harv. L. Rev. 1974

LENGTH: 31268 words

ARTICLE: THE **GENERATIVE INTERNET**

NAME: Jonathan L. Zittrain*

BIO: * Professor of Internet Governance and Regulation, Oxford University; Jack N. & Lillian R. Berkman Visiting Professor of Entrepreneurial Legal Studies, Harvard Law School. I received helpful insights on earlier drafts from Ben Adida, David Barron, Stefan Bechtold, Yochai Benkler, John Bracken, Tom Brown, David Clark, Julie Cohen, Einer Elhauge, Joan Feigenbaum, William Fisher, Charles Fried, Rainer Gawlick, Jack Goldsmith, David Isenberg, David Johnson, Lawrence Lessig, Betsy Masiello, Daniel Meltzer, Frank Michelman, Martha Minow, Thomas Nachbar, Charles Nesson, Andy Oram, John Palfrey, Randy Picker, David Post, Ira Rubinstein, Pamela Samuelson, William Stuntz, K.A. Taipale, and participants in the Georgetown Law Faculty Workshop Series, the Penn/Temple/Wharton Colloquium, the Berkman/Berkeley Colloquium, and the Harvard Law School summer research program. I thank Erin Ashwell, Bryan Choi, Michael Finney, Janna Fishman, Yukyan Lam, Bethany Lobo, Jacob Mermelstein, Jessie Mishkin, K. Joon Oh, Christina Olson, Erika Reinders, Joshua Salzman, Greg Skidmore, and Elizabeth Stark for outstanding research assistance.

SUMMARY:
... The generative capacity for unrelated and unaccredited audiences to build and distribute code and content through the Internet to its tens of millions of attached personal computers has ignited growth and innovation in information technology and has facilitated new creative endeavors. ... First, the early Internet consisted of nodes primarily at university computer science departments, U.S. government research units, and select technology companies with an interest in cutting-edge network research. ... It also opens PCs to the prospect of mass infection by a computer virus that exploits either user ignorance or a security vulnerability that allows code from the network to run on a PC without approval by its owner. ... As discussed earlier, TiVo's recorded shows may only be saved to a standard VCR or DVD, or in a copy-protected PC format, rather than, as would be trivial for the manufacturer to allow, digitally to a consumer's PC hard drive in an unprotected format or over the Internet to an arbitrary destination. ... As it does with an information appliance like the Xbox, Microsoft could decide to charge a fee to approve third-party software were it to revisit something akin to a brief, informally stated, and quickly disavowed possibility of collecting a "vig," or small tax, on "every transaction over the Internet that uses Microsoft's technology. ...

HIGHLIGHT:

The generative capacity for unrelated and unaccredited audiences to build and distribute code and content through the Internet to its tens of millions of attached personal computers has ignited growth and innovation in information technology and has facilitated new creative endeavors. It has also given rise to regulatory and entrepreneurial backlashes. A further backlash among consumers is developing in response to security threats that exploit the openness of the Internet and of PCs to third-party contribution. A shift in consumer priorities from generativity to stability will compel undesirable responses from regulators and markets and, if unaddressed, could prove decisive in closing today's open computing environments. This Article explains why PC openness is as important as network openness, as well as why today's open network might give rise to unduly closed endpoints. It argues that the Internet is better conceptualized as a generative grid that includes both PCs and networks rather than as an open network indifferent to the configuration of its endpoints. Applying this framework, the Article explores ways - some of them bound to be unpopular among advocates of an open Internet represented by uncompromising end-to-end neutrality - in which the Internet can be made to satisfy genuine and pressing security concerns while retaining the most important generative aspects of today's networked technology.

TEXT:
[*1975]

I. Introduction

From the moment of its inception in 1969, [n1] the Internet has been designed to serve both as a means of establishing a logical network and as a means of subsuming existing heterogeneous networks while allowing those networks to function independently - that is, both as a set of building blocks and as the glue holding the blocks together. [n2] **[*1976]** The Internet was also built to be open to any sort of device: any computer or other information processor could be part of the new network so long as it was properly interfaced, an exercise requiring minimal technical effort. [n3] These principles and their implementing protocols have remained essentially unchanged since the Internet comprised exactly two network subscribers, [n4] even as the network has grown to reach hundreds of millions of people in the developed and developing worlds. [n5]

Both mirroring and contributing to the Internet's explosive growth has been the growth of the personal computer (PC): an affordable, multifunctional device that is readily available at retail outlets and easily reconfigurable by its users for any number of purposes. The audience writing software for PCs, including software that is Internet capable, is itself massive and varied. [n6] This diverse audience has driven the variety of applications powering the rapid technological innovation to which we have become accustomed, in turn driving innovation in expressive endeavors such as politics, entertainment, journalism, and education. [n7]

Though the openness of the Internet and the PC to third-party contribution is now so persistent and universal as to seem inevitable, these **[*1977]** technologies need not have been configured to allow instant contribution from a broad range of third parties. This Article sketches the generative features of the Internet and PCs, along with the alternative configurations that these instrumentalities defeated. This Article then describes the ways in which the technologies' openness to third-party innovation, though possessing a powerful inertia that so far has resisted regulatory and commercial attempts at control, is increasingly at odds with itself.

Internet technologists have for too long ignored Internet and PC security threats that might precipitate an unprecedented intervention into the way today's Internet and PCs work. These security threats are more pervasive than the particular vulnerabilities arising from bugs in a particular operating system (OS): so long as OSs permit consumers to run third-party code - the sine qua non of a PC OS - users can execute malicious code and thereby endanger their own work and that of others on the network. A drumbeat of trust-related concerns plausibly could fuel a gradual but fundamental shift in consumer demand toward increased stability in computing platforms. Such a shift would interact with existing regulatory and commercial pressures - each grounded in advancing economically rational and often legally protected interests - to create solutions that make it not merely possible, but likely, that the personal computing and networking environment of tomorrow will be sadly hobbled, bearing little resemblance to the one that most of the world enjoys today.

The most plausible path along which the Internet might develop is one that finds greater stability by imposing greater constraint on, if not outright elimination of, the capacity of upstart innovators to demonstrate and deploy their genius to large audiences. Financial transactions over such an Internet will be more trustworthy, but the range of its users' business models will be narrow. This Internet's attached consumer hardware will be easier to understand and will be easier to use for purposes that the hardware manufacturers preconceive, but it will be off limits to amateur tinkering. Had such a state of affairs evolved by the Internet's twenty-fifth anniversary in 1994, many of its formerly unusual and now central uses would never have developed because the software underlying those uses would have lacked a platform for exposure to, and acceptance by, a critical mass of users.

Those who treasure the Internet's generative features must assess the prospects of either satisfying or frustrating the growing forces against those features so that a radically different technology configuration need not come about. Precisely because the future is uncertain, those who care about openness and the innovation that today's Internet and PC facilitate should not sacrifice the good to the perfect - or the future to the present - by seeking simply to maintain a tenuous technological status quo in the face of inexorable pressure to change. Rather, we should establish the principles that will blunt the most unappealing **[*1978]** features of a more locked-down technological future while acknowledging that unprecedented and, to many who work with information technology, genuinely unthinkable boundaries could likely become the rules from which we must negotiate exceptions.

Although these matters are of central importance to cyberlaw, they have generally remained out of focus in our field's evolving literature, which has struggled to identify precisely what is so valuable about today's Internet and which has focused on network openness to the exclusion of endpoint openness. Recognition of the importance and precariousness of the **generative Internet** has significant implications for, among other issues, the case that Internet service providers (ISPs) should unswervingly observe end-to-end neutrality, the proper focus of efforts under the rubric of "Internet governance," the multifaceted debate over digital rights management (DRM), and prospects for Internet censorship and filtering by authoritarian regimes.

Scholars such as Professors Yochai Benkler, Mark Lemley, and Lawrence Lessig have crystallized concern about keeping the Internet "open," translating a decades-old technical end-to-end argument concerning simplicity in network protocol design into a claim that ISPs should refrain from discriminating against particular sources or types of data. [n8] There is much merit to an open Internet, but this argument is really a proxy for something deeper: a generative networked grid. Those who make paramount "network neutrality" derived from end-to-end theory confuse means and ends, focusing on "network" without regard to a particular network policy's influence on the design of network endpoints such as PCs. As a result of this focus, political advocates of end-to-end are too often myopic; many writers seem to presume current PC and OS architecture to be fixed in an "open" position. If they can be persuaded to see a larger picture, they may agree to some compromises at the network level. If complete fidelity to end-to-end network neutrality persists, our PCs may be replaced by information appliances or may undergo a transformation from open platforms to gated communities to prisons, creating a consumer information environment that betrays the very principles that animate end-to-end theory.

The much-touted differences between free and proprietary PC OSs may not capture what is most important to the Internet's future. Proprietary systems can remain "open," as many do, by permitting unaffiliated third parties to write superseding programs and permitting PC **[*1979]** owners to install these programs without requiring any gatekeeping by the OS provider. In this sense, debates about the future of our PC experience should focus less on such common battles as Linux versus Microsoft Windows, as both are "open" under this definition, and more on generative versus nongenerative: understanding which platforms will remain open to third-party innovation and which will not.

The focus on the management of domain names among those participating in dialogues about Internet governance is equally unfortunate. Too much scholarly effort has been devoted to the question of institutional governance of this small and shrinking aspect of the Internet landscape. [n9] When juxtaposed with the generativity problem, domain names matter little. Cyberlaw's challenge ought to be to find ways of regulating - though not necessarily through direct state action - which code can and cannot be readily disseminated and run upon the generative grid of Internet and PCs, lest consumer sentiment and preexisting regulatory pressures prematurely and tragically terminate the grand experiment that is the Internet today.

New software tools might enable collective regulation if they can inform Internet users about the nature of new code by drawing on knowledge from the Internet itself, generated by the experiences of others who have run such code. Collective regulation might also entail new OS designs that chart a middle path between a locked-down PC on the one hand and an utterly open one on the other, such as an OS that permits users to shift their computers between "safe" and "experimental" modes, classifying new or otherwise suspect code as suited for only the experimental zone of the machine.

In other words, to Professor Lessig and those more categorically libertarian than he, this Article seeks to explain why drawing a bright line against nearly any form of increased Internet regulability is no longer tenable. Those concerned about preserving flexibility in Internet behavior and coding should participate meaningfully in debates about changes to network and PC architecture, helping to ameliorate rather than ignore the problems caused by such flexibility that, if left unchecked, will result in unfortunate, hamhanded changes to the way mainstream consumers experience cyberspace.

An understanding of the value of generativity, the risks of its excesses, and the possible means to reconcile the two, stands to reinvigorate the debate over DRM systems. In particular, this understanding promises to show how, despite no discernable movement toward the **[*1980]** locked-down dystopias Professors Julie Cohen and Pamela Samuelson predicted in the late 1990s, [n10] changing conditions are swinging consumer sentiment in favor of architectures that promote effective DRM. This realignment calls for more than general opposition to laws that prohibit DRM circumvention or that mandate particular DRM schemes: it also calls for a vision of affirmative technology policy - one that includes some measure of added regulability of intellectual property - that has so far proven elusive.

Finally, some methods that could be plausibly employed in the Western world to make PCs more tame for stability's sake can greatly affect the prospects for Internet censorship and surveillance by authoritarian regimes in other parts of the world. If PC technology is inaptly refined or replaced to create new points of control in the distribution of new code, such technology is horizontally portable to these regimes - regimes that currently effect their censorship through leaky network blockages rather than PC lockdown or replacement.

II. A Mapping of Generative Technologies

To emphasize the features and virtues of an open Internet and PC, and to help assess what features to give up (or blunt) to control some of the excesses wrought by placing such a powerfully modifiable technology into the mainstream, it is helpful to be clear about the meaning of generativity, the essential quality animating the trajectory of information technology innovation. Generativity denotes a technology's overall capacity to produce unprompted change driven by large, varied, and uncoordinated audiences. The grid of PCs connected by the Internet has developed in such a way that it is consummately generative. From the beginning, the PC has been designed to run almost any program created by the manufacturer, the user, or a remote third party and to make the creation of such programs a relatively easy task. When these highly adaptable machines are connected to a network with little centralized control, the result is a grid that is nearly completely open to the creation and rapid distribution of the innovations of technology-savvy users to a mass audience that can enjoy those innovations without having to know how they work.

[*1981]

A. Generative Technologies Defined

Generativity is a function of a technology's capacity for leverage across a range of tasks, adaptability to a range of different tasks, ease of mastery, and accessibility.

1. Capacity for Leverage. - Our world teems with useful objects, natural and artificial, tangible and intangible; leverage describes the extent to which these objects enable valuable accomplishments that otherwise would be either impossible or not worth the effort to achieve. Examples of devices or systems that have a high capacity for leverage

include a lever itself (with respect to lifting physical objects), a band saw (cutting them), an airplane (getting from one place to another), a piece of paper (hosting written language), or an alphabet (constructing words). A generative technology makes difficult jobs easier. The more effort a device or technology saves - compare a sharp knife to a dull one - the more generative it is. The greater the variety of accomplishments it enables - compare a sharp Swiss Army knife to a sharp regular knife - the more generative it is. [n11]

2. Adaptability. - Adaptability refers to both the breadth of a technology's use without change and the readiness with which it might be modified to broaden its range of uses. A given instrumentality may be highly leveraging, yet suited only to a limited range of applications. For example, although a plowshare can enable the planting of a variety of seeds, planting is its essential purpose. Its comparative leverage quickly vanishes when devoted to other tasks, such as holding doors open, and it is not readily modifiable for new purposes. The same goes for swords (presuming they are indeed difficult to beat into plowshares), guns, chairs, band saws, and even airplanes. In contrast, paper obtained for writing can instead (or additionally) be used to wrap fish. A technology that offers hundreds of different additional kinds of uses is more adaptable and, all else equal, more generative than a technology that offers fewer kinds of uses. Adaptability in a tool better permits leverage for previously unforeseen purposes.

3. Ease of Mastery. - A technology's ease of mastery reflects how easy it is for broad audiences both to adopt and to adapt it: how much skill is necessary to make use of its leverage for tasks they care about, regardless of whether the technology was designed with those tasks in mind. An airplane is neither easy to fly nor easy to modify for new purposes, not only because it is not inherently adaptable to purposes other than transportation, but also because of the skill required to make whatever transformative modifications might be possible and **[*1982]** because of the risk of physical injury if one poorly executes such modifications. Paper, in contrast, is readily mastered: children can learn how to use it the moment they enter preschool, whether to draw on or to fold into a paper airplane (itself much easier to fly and modify than a real one). Ease of mastery also refers to the ease with which people might deploy and adapt a given technology without necessarily mastering all possible uses. Handling a pencil takes a mere moment to understand and put to many uses even though it might require innate artistic talent and a lifetime of practice to achieve da Vincian levels of mastery with it. That is, much of the pencil's generativity stems from how useful it is both to the neophyte and to the master.

4. Accessibility. - The more readily people can come to use and control a technology, along with what information might be required to master it, the more accessible the technology is. Barriers to accessibility can include the sheer expense of producing (and therefore consuming) the technology, taxes and regulations imposed on its adoption or use (for example, to serve a government interest directly or to help private parties monopolize the technology through intellectual property law), and the use of secrecy and obfuscation by its producers to maintain scarcity without necessarily relying upon an affirmative intellectual property interest. [n12]

By this reckoning of accessibility, paper and plowshares are highly accessible, planes hardly at all, and cars somewhere in the middle. It might be easy to learn how to drive a car, but cars are expensive, and the privilege of driving, once earned by demonstrating driving skill, is revocable by the government. Moreover, given the nature of cars and driving, such revocation is not prohibitively expensive to enforce effectively.

5. Generativity Revisited. - As defined by these four criteria, generativity increases with the ability of users to generate new, valuable uses that are easy to distribute and are in turn sources of further innovation. It is difficult to identify in 2006 a technology bundle more generative than the PC and the Internet to which it attaches.

B. The Generative PC

The modern PC's generativity flows from a separation of hardware from software. [n13] From the earliest days of computing, this separation has been sensible, but it is not necessary. Both hardware and software comprise sets of instructions that operate upon informational inputs to **[*1983]** create informational outputs; the essential difference is that hardware is not easy to change once it leaves the factory. If the manufacturer knows enough about the computational task a machine will be asked to perform, the instructions to perform the task could be "bolted in" as hardware. Indeed, "bolting" is exactly what is done with an analog adding machine, digital calculator, "smart" typewriter, or the firmware within a coffeemaker that enables it to begin brewing at a user-selected time. These devices are all hardware and no software. Or, as some might say, their software is inside their hardware.

The essence - and genius - of software standing alone is that it allows unmodified hardware to execute a new algorithm, obviating the difficult and expensive process of embedding that algorithm in hardware. [n14] PCs carry on research computing's tradition of separating hardware and software by allowing the user to input new code. Such code can be loaded and run even once the machine is solely in the consumer's hands. Thus, the manufacturer can write new software after the computer leaves the factory, and a consumer needs to know merely how to load the cassette, diskette, or cartridge containing the software to enjoy its benefits. (In today's networked environment, the consumer need not take any action at all for such reprogramming to take place.) Further, software is comparatively easy for the manufacturer to develop because PCs carry on another sensible tradition of their institutional forbears: they make use of OSs. [n15] OSs provide a higher level of abstraction at which the programmer can implement his or her algorithms, allowing a programmer to take shortcuts when creating software. [n16] The ability to reprogram using external inputs provides adaptability; OSs provide leverage and ease of mastery.

Most significant, PCs were and are accessible. They were designed to run software not written by the PC manufacturer or OS publisher, including software written by those with whom these manufacturers had no special arrangements. [n17] Early PC manufacturers not only published the documentation necessary to write software that would run on their OSs, but also bundled high-level programming languages **[*1984]** along with their computers so that users themselves could learn to write software for use on their (and potentially others') computers. [n18] High-level programming languages are like automatic rather than manual transmissions for cars: they do not make PCs more leveraged, but they do make PCs accessible to a wider audience of programmers. [n19] This increased accessibility enables programmers to produce software that might not otherwise be written.

That the resulting PC was one that its own users could - and did - program is significant. The PC's truly transformative potential is fully understood, however, as a product of its overall generativity.

PCs were genuinely adaptable to any number of undertakings by people with very different backgrounds and goals. The early OSs lacked capabilities we take for granted today, such as multitasking, but they made it possible for programmers to write useful software with modest amounts of effort and understanding. [n20]

Users who wrote their own software and thought it suited for general use could hang out the shingle in the software business or simply share the software with others. A market in third-party software developed and saw a range of players, from commercial software publishers employing thousands of people, to collaborative software projects that made their outputs freely available, to hobbyists who showcased and sold their wares by advertising in computer user magazines or through local computer user groups. [n21] Such a range of developers enhanced the variety of applications that were written not only because accessibility arguably increased the sheer number of people **[*1985]** coding, but also because people coded for different reasons. While hobbyists might code for fun, others might code out of necessity, desiring an application but not having enough money to hire a commercial software development firm. And, of course, commercial firms could provide customized programming services to individual users or could develop packaged software aimed at broad markets.

This variety is important because the usual mechanisms that firms use to gauge and serve market demand for particular applications may not provide enough insight to generate every valuable application. It might seem obvious, for example, that a spreadsheet would be of great use to someone stuck with an adding machine, but the differences between the new application and the old method may prevent firms from recognizing the former as better serving other tasks. The PC, however, allowed multiple paths for the creation of new applications: Firms could attempt to gauge market need and then write and market an application to meet that need. Alternatively, people with needs could commission firms to write software. Finally, some could simply write the software themselves.

This configuration - a varied audience of software writers, most unaffiliated with the manufacturers of the PCs for which they wrote software - turned out to benefit the manufacturers, too, because greater availability of software enhanced the value of PCs and their OSs. Although many users were tinkerers, many more users sought PCs to accomplish particular purposes nearly out of the box; PC manufacturers could tout to the latter the instant uses of their machines provided by the growing body of available software.

Adapting a metaphor from Internet architecture, PC architecture can be understood as shaped like an hourglass. [n22] PC OSs sit in the narrow middle of the hourglass; most evolved slowly because proprietary OSs were closed to significant outside innovation. At the broad bottom of the PC hourglass lies cheap, commoditized hardware. At the broad top of the hourglass sits the application layer, in which most PC innovation takes place. Indeed, the computer and its OS are products, not services, and although the product life might be extended through upgrades, such as the addition of new physical memory chips or OS upgrades, the true value of the PC lies in its availability and stability as a platform for further innovation - running applications from a variety of sources, including the users themselves, and thus leveraging **[*1986]** the built-in power of the OS for a breathtaking range of tasks. [n23]

Despite the scholarly and commercial attention paid to the debate between free and proprietary OSs, [n24] the generativity of the PC depends little on whether its OS is free and open or proprietary and closed - that is, whether the OS may be modified by end users and third-party software developers. A closed OS like Windows can be and has been a highly generative technology. Windows is generative, for instance, because its application programming interfaces enable a programmer to rework nearly any part of the PC's functionality and give external developers ready access to the PC's hardware inputs and outputs, including scanners, cameras, printers, monitors, and mass storage. Therefore, nearly any desired change to the way Windows works can be imitated by code placed at the application level. [n25] The qualities of an OS that are most important have to do with making computing power, and the myriad tasks it can accomplish, available for varied audiences to adapt, share, and use through the ready creation and exchange of application software. If this metric is used, debates over whether the OS source code itself should be modifiable are secondary.

Closed but generative OSs - largely Microsoft DOS and then Microsoft Windows - dominate the latter half of the story of the PC's rise, and their early applications outperformed their appliance-based counterparts. For example, despite the head start from IBM's production of a paper tape-based word processing unit in World War II, **[*1987]** dedicated word processing appliances [n26] have been trounced by PCs running word processing software. PC spreadsheet software has never known a closed, appliance-based consumer product as a competitor; [n27] PCs have proven viable substitutes for video game consoles as platforms for gaming; [n28] and even cash registers have been given a run for their money by PCs configured to record sales and customer information, accept credit card numbers, and exchange cash at retail points of sale. [n29]

The technology and market structures that account for the highly generative PC have endured despite the roller coaster of hardware and software producer fragmentation and consolidation that make PC industry history otherwise heterogeneous. [n30] These structures and their implications can, and plausibly will, change in the near future. To understand these potential changes, their implications, and why they should be viewed with concern and addressed by action designed to minimize their untoward effects, it is important first to understand roughly analogous changes to the structure of the Internet that are presently afoot. As the remainder of this Part explains, the line between the PC and the Internet has become vanishingly thin, and it is no longer helpful to consider them separately when thinking about the future of consumer information technology and its relation to innovation and creativity.

C. The **Generative Internet**

The Internet today is exceptionally generative. It can be leveraged: its protocols solve difficult problems of data distribution, making it much cheaper to implement network-aware services. [n31] It is adaptable **[*1988]** in the sense that its basic framework for the interconnection of nodes is amenable to a large number of applications, from e-mail and instant messaging to telephony and streaming video. [n32] This adaptability exists in large part because Internet protocol relies on few assumptions about the purposes for which it will be used and because it efficiently scales to accommodate large amounts of data and large numbers of users. [n33] It is easy to master because it is structured to allow users to design new applications without having to know or worry about the intricacies of packet routing. [n34] And it is accessible because, at the functional level, there is no central gatekeeper with which to negotiate access and because its protocols are publicly available and not subject to intellectual property restrictions. [n35] Thus, programmers independent of the Internet's architects and service providers can offer, and consumers can accept, new software or services.

How did this state of affairs come to pass? Unlike, say, FedEx, whose wildly successful offline package delivery network depended initially on the financial support of venture capitalists to create an efficient physical infrastructure, those individuals thinking about the Internet in the 1960s and 1970s planned a network that would cobble together existing networks and then wring as much use as possible from them. [n36]

The network's design is perhaps best crystallized in a seminal 1984 paper entitled End-to-End Arguments in System Design. [n37] As this paper describes, the Internet's framers intended an hourglass design, with a simple set of narrow and slow-changing protocols in the middle, resting on an open stable of physical carriers at the bottom and any number of applications written by third parties on the top. The Internet hourglass, despite having been conceived by an utterly different group of architects from those who designed or structured the market **[*1989]** for PCs, thus mirrors PC architecture in key respects. The network is indifferent to both the physical media on which it operates and the nature of the data it passes, just as a PC OS is open to running upon a variety of physical hardware "below" and to supporting any number of applications from various sources "above."

The authors of End-to-End Arguments describe, as an engineering matter, why it is better to keep the basic network operating protocols simple - and thus to implement desired features at, or at least near, the endpoints of the networks. n38 Such features as error correction in data transmission are best executed by client-side applications that check whether data has reached its destination intact rather than by the routers in the middle of the chain that pass data along. n39 Ideas that entail changing the way the routers on the Internet work - ideas that try to make them "smarter" and more discerning about the data they choose to pass - challenge end-to-end philosophy.

The design of the Internet also reflects both the resource limitations and intellectual interests of its creators, who were primarily academic researchers and moonlighting corporate engineers. These individuals did not command the vast resources needed to implement a global network and had little interest in exercising control over the network or its users' behavior. [n40] Energy spent running the network was seen as a burden; the engineers preferred to design elegant and efficient protocols whose success was measured precisely by their ability to run without effort. Keeping options open for growth and future development was seen as sensible, [n41] and abuse of the network was of little worry because the people using it were the very people designing it - a culturally homogenous set of people bound by their desire to see the network work.

Internet designers recognized the absence of persistent connections between any two nodes of the network, [n42] and they wanted to allow additions to the network that neither taxed a central hub nor required centrally managed adjustments to overall network topography. [n43] These constraints inspired the development of a stateless protocol - one that did not presume continuous connections or centralized knowledge of those connections - and packet switching, [n44] which broke all **[*1990]** data into small, discrete packets and permitted multiple users to share a connection, with each sending a snippet of data and then allowing someone else's data to pass. The constraints also inspired the fundamental protocols of packet routing, by which any number of routers in a network - each owned and managed by a different entity - maintain tables indicating in which rough direction data should travel without knowing the exact path to the data's final destination. [n45]

A flexible, robust platform for innovation from all corners sprang from these technical and social underpinnings. Two further historical developments assured that an easy-to-master Internet would also be extraordinarily accessible. First, the early Internet consisted of nodes primarily at university computer science departments, U.S. government research units, and select technology companies with an interest in cutting-edge network research. [n46] These institutions collaborated on advances in bandwidth management and tools for researchers to use for communication and discussion. n47 But consumer applications were nowhere to be found until the Internet began accepting commercial interconnections without requiring academic or government research justifications, and the population at large was solicited to join. n48 This historical development - the withering away of the norms against commercial use and broad interconnection that had been reflected in a National Science Foundation admonishment that its contribution to the functioning Internet backbone be used for noncommercial purposes n49 - greatly increased the Internet's generativity. It opened development of networked technologies to a broad, commercially driven audience that individual companies running proprietary services did not think to invite and that the original designers of the Internet would not have thought to include in the design process.

A second historical development is easily overlooked because it may in retrospect seem inevitable: the dominance of the Internet as the network to which PCs connected, rather than the emergence of proprietary networks analogous to the information appliances that PCs themselves beat. The first large-scale networking of consumer PCs **[*1991]** took place through self-contained "walled garden" networks like CompuServe, The Source, and Prodigy. n50 Each network connected its members only to other subscribing members and to content managed and cleared through the network proprietor. For example, as early as 1983, a home computer user with a CompuServe subscription was able to engage in a variety of activities - reading news feeds, sending e-mail, posting messages on public bulletin boards, and participating in rudimentary multiplayer games (again, only with other CompuServe subscribers). n51 But each of these activities was coded by CompuServe or its formal partners, making the network much less generatively accessible than the Internet would be. Although CompuServe entered into some development agreements with outside software programmers and content providers, n52 even as the service grew to almost two million subscribers by 1994, its core functionalities remained largely unchanged. n53

The proprietary services could be leveraged for certain tasks, and their technologies were adaptable to many purposes and easy to master, but consumers' and outsiders' inability to tinker easily with the services limited their generativity. n54 They were more like early video game consoles n55 than PCs: capable of multiple uses, through the development of individual "cartridges" approved by central management, yet slow to evolve because potential audiences of developers were slowed or shut out by centralized control over the network's services.

The computers first attached to the Internet were mainframes and minicomputers of the sort typically found within university computer science departments, n56 and early desktop access to the Internet came through specialized nonconsumer workstations, many running variants **[*1992]** of the UNIX OS. n57 As the Internet expanded and came to appeal to nonexpert participants, the millions of PCs in consumer hands were a natural source of Internet growth. n58 Despite the potential market, however, no major OS producer or software development firm quickly moved to design Internet protocol compatibility into PC OSs. PCs could access "walled garden" proprietary services, but their ability to run Internet-aware applications locally was limited.

A single hobbyist took advantage of PC generativity and produced and distributed the missing technological link. Peter Tattam, an employee in the psychology department of the University of Tasmania, wrote Trumpet Winsock, a program that allowed owners of PCs running Microsoft Windows to forge a point-to-point Internet connection with the servers run by the nascent ISP industry. n59 Ready consumer access to Internet-enabled applications such as Winsock, coupled with the development of graphical World Wide Web protocols and the PC browsers to support them - all initially noncommercial ventures - marked the beginning of the end of proprietary information services and peer-to-peer telephone-networked environments like electronic bulletin boards. After recognizing the popularity of Tattam's software, Microsoft bundled the functionality of Winsock with later versions of Windows 95. n60

As PC users found themselves increasingly able to access the Internet, proprietary network operators cum content providers scrambled to reorient their business models away from corralled content and toward accessibility to the wider Internet. n61 These online service providers quickly became mere ISPs, with their users branching out to the thriving Internet for programs and services. n62 Services like CompuServe's "Electronic Mall" - an e-commerce service allowing outside vendors, through arrangements with CompuServe, to sell products to subscribers n63 **[*1993]** - were lost amidst an avalanche of individual websites selling goods to anyone with Internet access.

The greater generativity of the Internet compared to that of proprietary networked content providers created a network externality: as more information consumers found their way to the Internet, there was more reason for would-be sources of information to set up shop there, in turn attracting more consumers. Dispersed third parties could and did write clients and servers for instant messaging, [n64] web browsing, [n65] e-mail exchange, [n66] and Internet searching. [n67] Furthermore, the Internet remained broadly accessible: anyone could cheaply sign on, immediately becoming a node on the Internet equal to all others in information exchange capacity, limited only by bandwidth. Today, due largely to the happenstance of default settings on consumer wireless routers, Internet access might be free simply because one is sitting on a park bench near the apartment of a broadband subscriber who has "gone wireless." [n68]

The resulting Internet is a network that no one in particular owns and that anyone can join. Of course, joining requires the acquiescence of at least one current Internet participant, but if one is turned away at one place, there are innumerable others to court, and commercial ISPs provide service at commoditized rates. [n69] Those who want to offer services on the Internet or connect a formerly self-contained local network - such as a school that wants to link its networked computers **[*1994]** to the Internet - can find fast and reliable broadband Internet access for several hundred dollars per month. [n70] Quality of service (QoS) - consistent bandwidth between two points [n71] - is difficult to achieve because the Internet, unlike a physical distribution network run by one party from end to end such as FedEx, comprises so many disparate individual networks. Nevertheless, as the backbone has grown and as technical innovations have both reduced bandwidth requirements and staged content "near" those who request it, [n72] the network has proven remarkably effective even in areas - like person-to-person video and audio transmission - in which it at first fell short.

D. The Generative Grid

Both noncommercial and commercial enterprises have taken advantage of open PC and Internet technology, developing a variety of Internet-enabled applications and services, many going from implementation to popularity (or notoriety) in a matter of months or even days. Yahoo!, Amazon.com, eBay, flickr, the Drudge Report, CNN.com, Wikipedia, MySpace: the list of available services and activities could go into the millions, even as a small handful of Web sites and applications account for a large proportion of online user activity. [n73] Some sites, like CNN.com, are online instantiations of existing institutions; others, from PayPal to Geocities, represent new ventures by formerly unestablished market participants. Although many of the offerings created during the dot-com boom years - roughly 1995 to 2000 **[*1995]** - proved premature at best and flatly ill-advised at worst, the fact remains that many large companies, including technology-oriented ones, ignored the Internet's potential for too long. [n74]

Significantly, the last several years have witnessed a proliferation of PCs hosting broadband Internet connections. [n75] The generative PC has become intertwined with the **generative Internet,** and the whole is now greater than the sum of its parts. A critical mass of always-on computers means that processing power for many tasks, ranging from difficult mathematical computations to rapid transmission of otherwise prohibitively large files, can be distributed among hundreds, thousands, or millions of PCs. [n76] Similarly, it means that much of the information that once needed to reside on a user's PC to remain conveniently accessible - documents, e-mail, photos, and the like - can instead be stored somewhere on the Internet. [n77] So, too, can the programs that a user might care to run.

This still-emerging "generative grid" expands the boundaries of leverage, adaptability, and accessibility for information technology. It also raises the ante for the project of cyberlaw because the slope of this innovative curve may nonetheless soon be constrained by some of the very factors that have made it so steep. Such constraints may arise because generativity is vulnerability in the current order: the fact that tens of millions of machines are connected to networks that can convey reprogramming in a matter of seconds means that those computers stand exposed to near-instantaneous change. This kind of generativity keeps publishers vulnerable to the latest tools of intellectual property infringement, crafted ever more cleverly to evade monitoring and control, and available for installation within moments everywhere. It also opens PCs to the prospect of mass infection by a computer virus that exploits either user ignorance or a security vulnerability that allows code from the network to run on a PC without approval by its owner.

Shoring up these vulnerabilities will require substantial changes in some aspects of the grid, and such changes are sure to affect the current level of generativity. Faced with the prospect of such changes, we **[*1996]** must not fight an overly narrow if well-intentioned battle simply to preserve end-to-end network neutrality or to focus on relative trivialities like domain names and cookies. Recognizing the true value of the grid - its hypergenerativity - along with the magnitude of its vulnerabilities and the implications of eliminating those vulnerabilities, leads to the realization that we should instead carefully tailor reforms to address those vulnerabilities with minimal impact on generativity.

III. Generative Discontent

To appreciate the power of the new and growing Internet backlash - a backlash that augurs a dramatically different, managed Internet of the sort that content providers and others have unsuccessfully strived to bring about - one must first identify three powerful groups that may find common cause in seeking a less generative grid: regulators (in part driven by threatened economic interests, including those of content providers), mature technology industry players, and consumers. These groups are not natural allies in technology policy, and only recently have significant constituencies from all three sectors gained momentum in promoting a refashioning of Internet and PC architecture that would severely restrict the Internet's generativity .

A. Generative Equilibrium

Cyberlaw scholarship has developed in three primary strands. First, early work examined legal conflicts, usually described in reported judicial opinions, arising from the use of computers and digital networks, including the Internet. [n78] Scholars worked to apply statutory or common law doctrine to such conflicts, at times stepping back to dwell on whether new circumstances called for entirely new legal approaches. [n79] Usually the answer was that they did not: these scholars **[*1997]** mostly believed that creative application of existing doctrine, directly or by analogy, could resolve most questions. Professors such as David Johnson and David Post disagreed with this view, maintaining in a second strand of scholarship that cyberspace is different and therefore best regulated by its own sets of rules rather than the laws of territorial governments. [n80]

In the late 1990s, Professor Lessig and others argued that the debate was too narrow, pioneering yet a third strand of scholarship. Professor Lessig argued that a fundamental insight justifying cyberlaw as a distinct field is the way in which technology - as much as the law itself - can subtly but profoundly affect people's behavior: "Code is law." [n81] He maintained that the real projects of cyberlaw are both to correct the common misperception that the Internet is permanently unregulable and to reframe doctrinal debates as broader policy-oriented ones, asking what level of regulability would be appropriate to build into digital architectures. For his part, Professor Lessig argued that policymakers should typically refrain from using the powers of regulation through code - powers that they have often failed, in the first instance, to realize they possess. [n82]

The notion that code is law undergirds a powerful theory, but some of its most troubling empirical predictions about the use of code to restrict individual freedom have not yet come to pass. Work by professors such as Julie Cohen and Pamela Samuelson has echoed Professor Lessig's fears about the ways technology can unduly constrain individual **[*1998]** behavior - fears that include a prominent subset of worries about socially undesirable digital rights management and "trusted systems." [n83]

Trusted systems are systems that can be trusted by outsiders against the people who use them. [n84] In the consumer information technology context, such systems are typically described as "copyright management" or "rights management" systems, although such terminology is loaded. As critics have been quick to point out, the protections afforded by these systems need not bear any particular relationship to the rights granted under, say, U.S. copyright law. [n85] Rather, the possible technological restrictions on what a user may do are determined by the architects themselves and thus may (and often do) prohibit many otherwise legal uses. An electronic book accessed through a rights management system might, for example, have a limitation on the number of times it can be printed out, and should the user figure out how to print it without regard to the limitation, no fair use defense would be available. Similarly, libraries that subscribe to electronic material delivered through copyright management systems may find themselves technologically incapable of lending out that material the way a traditional library lends out a book, even though the act of lending is a privilege - a defense to copyright infringement for unlawful "distribution" - under the first sale doctrine. n86

The debate over technologically enforced "private" copyright schemes in the United States grew after the Digital Millennium Copyright **[*1999]** Act n87 (DMCA) was signed into law with a raft of complex anticircumvention provisions. n88 These provisions are intended to support private rights management schemes with government sanctions for their violation. They provide both criminal and civil penalties for those who use, market, or traffic in technologies designed to circumvent technological barriers to copyright infringement. They also penalize those who gain mere "access" to such materials, a right not reserved to the copyright holder. n89 For example, if a book has a coin slot on the side and requires a would-be reader to insert a coin to read it, one who figures out how to open and read it without inserting a coin might be violating the anticircumvention provisions - even though one would not be directly infringing the exclusive rights of copyright.

These are the central fears of the "code is law" theory as applied to the Internet and PCs: technical barriers prevent people from making use of intellectual works, and nearly any destruction of those barriers, even to enable lawful use, is unlawful. But these fears have largely remained hypothetical. To be sure, since the DMCA's passage a number of unobtrusive rights management schemes have entered wide circulation. n90 Yet only a handful of cases have been brought against those who have cracked such schemes, n91 and there has been little if any decrease in consumers' capacity to copy digital works, whether for fair use or for wholesale piracy. Programmers have historically been able to crack nearly every PC protection scheme, and those less technically inclined can go to the trouble of generating a fresh copy of a digital work without having to crack the scheme, exploiting the "analog **[*2000]** hole" by, for example, filming a television broadcast or recording a playback of a song. n92 Once an unprotected copy is generated, it can then be shared freely on Internet file-sharing networks that run on generative PCs at the behest of their content-hungry users. n93

Thus, the mid-1990s' fears of digital lockdown through trusted systems may seem premature or unfounded. As a practical matter, any scheme designed to protect content finds itself rapidly hacked, and the hack (or the content protected) in turn finds itself shared with technically unsophisticated PC owners. Alternatively, the analog hole can be used to create a new, unprotected master copy of protected content. The fact remains that so long as code can be freely written by anyone and easily distributed to run on PC platforms, trusted systems can serve as no more than speed bumps or velvet ropes - barriers that the public circumvents should they become more than mere inconveniences. Apple's iTunes Music Store is a good example of this phenomenon: music tracks purchased through iTunes are encrypted with Apple's proprietary scheme, n94 and there are some limitations on their use that, although unnoticeable to most consumers, are designed to prevent the tracks from immediately circulating on peer-to-peer networks. n95 But the scheme is easily circumvented by taking music purchased from the store, burning it onto a standard audio CD, and then re-ripping the CD into an unprotected format, such as MP3. n96

An important claim endures from the third "code is law" strand of cyberlaw scholarship. Professors Samuelson, Lessig, and Cohen were right to raise alarm about such possibilities as losing the ability to read anonymously, to lend a copyrighted work to a friend, and to make fair use of materials that are encrypted for mere viewing. However, the second strand of scholarship, which includes the empirical claims **[*2001]** about Internet separatism advanced by Professor Johnson and David Post, appears to explain why the third strand's concerns are premature. Despite worries about regulation and closure, the Internet and the PC have largely remained free of regulation for mainstream users who experience regulation in the offline world but wish to avoid it in the online world, whether to gamble, obtain illegal pornography, or reproduce copyrighted material.

The first strand, which advocates incremental doctrinal adaptation to new technologies, explains how Internet regulators have indeed acted with a light touch even when they might have had more heavy-handed options. As I discuss elsewhere, the history of Internet regulation in the Western world has been tentative and restrained, with a focus by agile Internet regulators on gatekeeping regimes, effected through discrete third parties within a sovereign's jurisdiction or through revisions to technical architectures. [n97] Although Internet regulators are powerful in that they can shape, apply, and enforce the authority of major governments, many have nevertheless proven willing to abstain from major intervention.

This lack of intervention has persisted even as the mainstream adoption of the Internet has increased the scale of interests that Internet uses threaten. Indeed, until 2001, the din of awe and celebration surrounding the Internet's success, including the run-up in stock market valuations led by dot-coms, drowned out many objections to and discussion about Internet use and reform - who would want to disturb a goose laying golden eggs? Further, many viewed the Internet as immutable and thought the problems it generated should be dealt with piecemeal, without considering changes to the way the network functioned - a form of "is-ism" that Professor Lessig's third strand of cyberlaw scholarship famously challenged in the late 1990s even as he advocated that the Internet's technology remain essentially as is, unfettered by regulation. [n98]

By 2002 the dot-com hangover was in full sway, serving as a backdrop to a more jaundiced mainstream view of the value of the Internet. The uptake of consumer broadband, coupled with innovative but disruptive applications like the file-sharing service Napster - launched by amateurs but funded to the tune of millions of dollars during the boom - inspired fear and skepticism rather than envy and **[*2002]** mimicry among content providers. And a number of lawsuits against such services, initiated during the crest of the Internet wave, had by this time resulted in judgments against their creators. [n99] The woes of content providers have made for a natural starting point in understanding the slowly building backlash to the **generative Internet,** and they appropriately continue to inform a large swath of cyberlaw scholarship. But the persistence of the publishers' problems and the dot-com bust have not alone persuaded regulators or courts to tinker fundamentally with the Internet's generative capacity. The duties of Western-world ISPs to police or preempt undesirable activities are still comparatively light, even after such perceived milestones as the Digital Millennium Copyright Act and the Supreme Court's decision against file-sharing promoters in Metro-Goldwyn-Mayer Studios Inc. v. Grokster, Ltd. [n100] Although there is more regulation in such countries as China and Saudi Arabia, where ISPs are co-opted to censor content and services that governments find objectionable, [n101] regulatory efforts are still framed as exceptions to the rule that where the Internet goes, freedom of action by its subscribers follows.

By incorporating the trajectory of disruption by the **generative Internet** followed by comparatively mild reaction by regulators, the three formerly competing strands of cyberlaw scholarship can be reconciled to explain the path of the Internet: Locked-down PCs are possible but undesirable. Regulators have been unwilling to take intrusive measures to lock down the generative PC and have not tried to trim the Internet's corresponding generative features to any significant degree. The Internet's growth has made the underlying problems of concern to regulators more acute. Despite their unwillingness to act to bring about change, regulators would welcome and even encourage a PC/Internet grid that is less exceptional and more regulable.

[*2003]

B. Generativity as Vulnerability: The Cybersecurity Fulcrum

Mainstream firms and individuals now use the Internet. Some use it primarily to add to the generative grid, whether to create new technology or new forms of art and other expression facilitated by that technology. Others consume intellectual work over the Internet, whether developed generatively or simply ported from traditional distribution channels for news and entertainment such as CDs, DVDs, and broadcast radio and television. Still others, who use it primarily for nonexpressive tasks like shopping or selling, embrace simplicity and stability in the workings of the technology.

Consumers hold the key to the balance of power for tomorrow's Internet. They are powerful because they drive markets and because many vote. If they remain satisfied with the Internet's generative characteristics - continuing to buy the hardware and software that make it possible and to subscribe to ISPs that offer unfiltered access to the Internet at large - then the regulatory and industry forces that might otherwise wish to constrain the Internet will remain in check, maintaining a generative equilibrium. If, in contrast, consumers change their view - not simply tolerating a locked-down Internet but embracing it outright - then the balance among competing interests that has favored generative innovation will tilt away from it. Such a shift is and will continue to be propelled by an underappreciated phenomenon: the security vulnerabilities inherent in a **generative Internet** populated by mainstream consumers possessing powerful PCs.

The openness, decentralization, and parity of Internet nodes described in Part II are conventionally understood to contribute greatly to network security because a network without a center will have no central point of breakage. But these characteristics also create vulnerabilities, especially when united with both the flaws in the machines connecting to the network and the users' lack of technical sophistication. These vulnerabilities have been individually noted but collectively ignored. A decentralized Internet does not lend itself well to collective action, and its vulnerabilities have too readily been viewed as important but not urgent. Dealing with such vulnerabilities collectively can no longer be forestalled, even though a locus of responsibility for action is difficult to find within such a diffuse configuration.

1. A Threat Unanswered and Unrealized. - On the evening of November 2, 1988, a Cornell University graduate student named Robert Tappan Morris transmitted a small piece of software over the Internet from Cornell to MIT. [n102] At the time, few PCs were attached to the Internet; rather, it was the province of mainframes, minicomputers, **[*2004]** and professional workstations in institutional hands. [n103] Morris's software, when run on the MIT machine, sought out other nearby computers on the Internet, and it then behaved as if it were a person wanting to log onto those machines. [n104]

The sole purpose of the software - called variously a "virus" or, because it could transmit itself independently, a "worm" - was to propagate, attempting to crack each newly discovered computer by guessing user passwords or exploiting a known security flaw. [n105] It succeeded. By the following morning, a large number of machines on the Internet had ground to a near halt. Each had been running the same flavor of vulnerable Unix software that the University of California at Berkeley had made freely available. [n106] Estimates of the number of computers affected ranged from 1000 to 6000. [n107]

Within a day or two, puzzled network administrators reverse engineered the worm. [n108] Its methods of propagation were discovered, and the Berkeley computer science department coordinated patches for vulnerable Unix systems. [n109] These patches were made available over the Internet, at least to those users who had not elected to disconnect their machines to prevent reinfection. [n110] The worm was traced to Morris, who apologized; a later criminal prosecution for the act resulted in his receiving three years' probation, 400 hours of community service, and a $ 10,050 fine. [n111]

The Morris worm was a watershed event because it was the first virus spread within an always-on network linking always-on computers. Despite the dramatically smaller and demographically dissimilar **[*2005]** information technology landscape that existed when the Morris worm was released, nearly everything one needs to know about the risks to Internet and PC security today resides within this story. Responses to security breaches today have not significantly improved since the response in 1988 to the Morris worm, even though today's computing and networking are categorically larger in scale and less tightly controlled.

The computers of the 1988 Internet could be compromised because they were general-purpose machines, running OSs with which outsiders could become familiar and for which these outsiders could write executable code. They were powerful enough to run multiple programs and host multiple users simultaneously. [n112] They were generative. When first compromised, they were able to meet the demands of both the worm and their regular users, thereby buying time for the worm's propagation before human intervention was made necessary. [n113] The OS and other software running on the machines were not perfect: they contained flaws that rendered them more generatively accessible than their designers intended. [n114] More important, even without such flaws, the machines were designed to be operated at a distance and to receive and run software sent from a distance. They were powered and attached to a network continuously, even when not in use by their owners. And the users of these machines were lackadaisical about implementing available fixes to known software vulnerabilities - and often utterly, mechanistically predictable in the passwords they chose to protect entry into their computer accounts. n115

Each of these facts contributing to the vulnerabilities of networked life remains true today, and the striking feature is how few recurrences of truly disruptive security incidents there have been since 1988. Remarkably, a network designed for communication among academic and government researchers scaled beautifully as hundreds of millions of new users signed on during the 1990s, a feat all the more impressive when one considers how technologically unsavvy most new users were in comparison to the 1988 crowd. However heedless network administrators in 1988 were of good security hygiene, mainstream consumers of the 1990s were far worse. Few knew how to manage or code their PCs, much less how to apply patches rigorously or observe good password security.

[*2006] Network engineers and government officials did not take any significant preventive actions to forestall another Morris-type worm, despite a brief period of soul searching. Although many who reflected on the Morris worm grasped the dimensions of the problem it heralded, the problem defied easy solution because management of both the Internet and the computers attached to it was so decentralized and because any truly effective solution would cauterize the very purpose of the Internet. n116

The worm did not literally "infect" the Internet itself because it did not reprogram the Internet's various distributed routing mechanisms and corresponding links. Instead, the burden on the network was simply increased traffic, as the worm impelled computers to communicate with each other while it attempted to spread ad infinitum. n117 The computers actually infected were managed by disparate groups and individuals who answered to no particular authority for their use or for failing to secure them against compromise. n118 These features led those individuals overseeing network protocols and operation to conclude that the worm was not their problem.

Engineers concerned with the Internet protocols instead responded to the Morris worm incident by suggesting that the appropriate solution was more professionalism among coders. The Internet Engineering Task Force - the far-flung, unincorporated group of engineers who work on Internet standards and who have defined its protocols through a series of formal "request for comments" documents, or RFCs - published informational RFC 1135 as a postmortem on the worm incident. n119 After describing the technical details of the worm, the RFC focused on "computer ethics" and the need to instill and enforce ethical standards as new people - mostly young computer scientists such as Morris - signed on to the Internet. n120

The state of play in 1988, then, was to acknowledge the gravity of the worm incident while avoiding structural overreaction to it. The most concrete outcomes were a criminal prosecution of Morris, an apparent consensus within the Internet technical community regarding individual responsibility for the problem, and the Department of Defense's creation of the CERT Coordination Center to monitor the overall health of the Internet and dispatch news of security threats n121 - **[*2007]** leaving preemptive or remedial action in the hands of individual computer operators.

Compared to the computers connected to the 1988 Internet, the computers connected to the proprietary consumer networks of the 1980s described in Part II were not as fundamentally vulnerable because those networks could be much more readily purged of particular security flaws without sacrificing the very foundations of their existence. n122 For example, although it would have been resource intensive, CompuServe could have attempted to scan its network traffic at designated gateways to prevent the spread of a worm. More important, no worm could have spread through CompuServe in the manner of Morris's because the computers attached to CompuServe were configured as mere "dumb terminals": they exchanged data, not executable files, with CompuServe and therefore did not run software from the CompuServe network. Moreover, the mainframe computers at CompuServe with which those dumb terminals communicated were designed to ensure that the line between users and programmers was observed and digitally enforced. n123

Like CompuServe, the U.S. long-distance telephone network of the 1970s was intended to convey data - in the form of telephone conversations - rather than code between consumers. In the early 1970s, several consumers who were curious about the workings of the telephone network discovered that telephone lines used a tone at a frequency of 2600

Hertz to indicate that they were idle. n124 As fortune would have it, a toy whistle packaged as a prize in boxes of Cap'n Crunch cereal could, when one hole was covered, generate exactly that tone. n125 People in the know could dial toll-free numbers from their home phones, blow the whistle to clear but not disconnect the line, and then dial a new, non-toll free number, which would be connected without incurring a charge. n126 When this imaginative scheme came to light, AT&T reconfigured the network so that the 2600-Hertz tone no longer controlled it. n127 Indeed, the entire notion of in-band signaling was eliminated so that controlling the network required something other than generating sound into the telephone mouthpiece. To accomplish this reconfiguration, AT&T separated the data and executable **[*2008]** code in the network - a natural solution in light of the network's centralized control structure and its purpose of carrying data, not hosting user programming.

An understanding of AT&T's long-distance telephone network and its proprietary information service counterparts, then, reveals both the benefit and the bane of yesterday's Internet and today's generative grid: the Internet's very channels of communication are also channels of control. What makes the Internet so generative is that it can transmit both executable code to PCs and data for PC users. This duality allows one user to access her news feeds while another user creates a program that aggregates hundreds of news feeds and then provides that program to millions of other users. If one separated data from the executable code as AT&T or CompuServe did, it would incapacitate the generative power of the Internet.

CompuServe's mainframes were the servers, and the consumer computers were the clients. On AT&T's network, both the caller and the call's recipient were "clients," with their link orchestrated by a centralized system of switches - the servers. On the Internet of both yesterday and today, a server is definable only circularly: it is a computer that has attached itself to the Internet and made itself available as a host to others - no more, no less. n128 A generative PC can be used as a client or it can be turned into a website that others access. Lines between consumer and programmer are self-drawn and self-enforced.

Given this configuration, it is not surprising that there was little momentum for collective action after the Morris worm scare. The decentralized, nonproprietary ownership of the Internet and its computers made it difficult to implement any structural revisions to the way they worked. And more important, it was simply not clear what curative steps would not entail drastic and purpose-altering changes to the very fabric of the Internet: the notion was so wildly out of proportion to the level of the perceived threat that it was not even broached.

2. The PC/Network Grid and a Now-Realized Threat. - In the absence of significant reform, then, how did the Internet largely dodge further bullets? A thorough answer draws on a number of factors, each instructive for the situation in which the grid finds itself today.

First, the computer scientists were right that the ethos of the time frowned upon destructive hacking. n129 Even Morris's worm arguably **[*2009]** did more damage than its author had intended, and for all the damage it did cause, the worm had no payload other than itself. Once the worm compromised a system, it would have been trivial for Morris to have directed the worm, say, to delete as many files as possible. Like Morris's worm, the overwhelming majority of viruses that followed in the 1990s reflected similar authorial restraint n130: they infected simply for the purpose of spreading further. Their damage was measured both by the effort required to eliminate them and by the burden placed upon network traffic as they spread, rather than by the cost of reconstructing lost files or accounting for sensitive information that had been compromised. n131

Second, network operations centers at universities and other institutions became full-time facilities, with professional administrators who more consistently heeded the admonitions to update their patches regularly and to scout for security breaches. Administrators carried beepers and were prepared to intervene quickly in the case of a system intrusion. And even though mainstream consumers began connecting unsecured PCs to the Internet in earnest by the mid-1990s, their machines generally used temporary dial-up connections, greatly reducing the amount of time per day during which their computers were exposed to security threats and might contribute to the problem.

Last, there was no commercial incentive to write viruses - they were generally written for fun. Thus, there was no reason that substantial resources would be invested in creating viruses or in making them especially virulent. [n132]

Each of these factors that created a natural security bulwark has been attenuating as powerful networked PCs reach consumers' hands. Universal ethics have become infeasible as the network has become ubiquitous: truly anyone who can find a computer and a connection is allowed online. Additionally, consumers have been transitioning to **[*2010]** always-on broadband, [n133] their computers are ever more powerful and therefore capable of more mischief should they be compromised, and their OSs boast numerous well-known flaws. Furthermore, many viruses and worms now have purposes other than simply to spread, including purposes that reflect an underlying business incentive. What seemed truly remarkable when discovered is now commonplace: viruses can compromise a PC and leave it open to later instructions, such as commanding it to become its own Internet mail server, sending spam to e-mail addresses harvested from the hard disk, or conducting web searches to engage in advertising fraud, with the entire process typically unnoticeable by the PC's owner. In one notable experiment conducted in the fall of 2003, a researcher connected to the Internet a PC that simulated running an "open proxy," a condition unintentionally common among Internet mail servers. [n134] Within ten hours, spammers had found the computer and had begun attempting to send mail through it. [n135] Sixty-six hours later, the computer had recorded an attempted 229,468 distinct messages directed at 3,360,181 would-be recipients. [n136] (The researcher's computer pretended to forward the spam but in fact threw it away. [n137])

CERT Coordination Center statistics reflect this sea change. The organization began documenting the number of attacks against Internet-connected systems - called "incidents" - at its founding in 1988. As Figure 1 shows, the increase in incidents since 1997 has been exponential, roughly doubling each year through 2003.

[*2011]

Figure 1. Number of Security Incidents Reported to CERT/CC, 1988-2003 [n138]

[SEE FIGURE 1 IN ORIGINAL]

CERT announced in 2004 that it would no longer keep track of this figure because attacks have become so commonplace and widespread as to be indistinguishable from one another. [n139]

At the time of the Morris worm, there were an estimated 60,000 distinct computers, or "hosts," on the Internet. [n140] In 2005, the count was over 353 million, [n141] and worldwide PCs in use were estimated at almost 900 million. [n142] This massive set of always-on, powerful PCs, many with high-speed Internet connections and run by unskilled users, is a phenomenon new to the twenty-first century. Today's viruses are highly and nearly instantly communicable, capable of sweeping through a substantial worldwide population in a matter of hours. [n143] **[*2012]** The symptoms may reveal themselves to users instantly or they may lie in remission, at the choice of the virus author. Even uninfected systems can fall prey to a widespread infection because the spread of a virus can disrupt network connectivity and because viruses can be programmed to attack a specific network destination by simply accessing it repeatedly. Summed across all infected machines, such a distributed denial of service attack can disrupt connectivity to even the most well-connected and well-defended server.

Well-crafted worms and viruses routinely infect swaths of Internet-connected PCs. In 2004, for example, the Sasser worm infected more than half a million computers in three days. [n144] The Slammer worm in January 2003 infected 90% of a particular kind of Microsoft server - hindering hundreds of thousands of computers - within fifteen minutes. [n145] The SoBig.F virus was released in August 2003 and quickly accounted for over 70% of all e-mail in the world, causing 23.2 million virus-laden e-mails to arrive on America Online's doorstep alone. [n146] SoBig.F receded as a threat, in part because its author designed it to expire a few weeks later. [n147] If any of these pieces of malware [n148] had been truly "mal" - for example, programmed to propagate indefinitely, erasing hard drives, transposing numbers randomly inside spreadsheets, or adding profanity at random intervals to Word documents - no security system would have stood in the way of major compromise. Although these examples all occurred on PCs running Microsoft Windows, the fundamental problem arises from generativity, not from a particular vendor's security flaw. Whether running MacOS, Windows, or a flavor of UNIX, a PC that is flexible enough to have its code rewritten with the uninformed assent of its user is a PC that can be compromised.

Combine one well-written worm of the sort that can evade firewalls and antivirus software with one truly malicious worm writer, and worrisome low-level annoyances could spike to more acute effects: check-in unavailable at some airline counters; no overnight deliveries or other forms of package and letter distribution; payroll software unable to generate paychecks for millions of workers; or vital records held in **[*2013]** medical offices, schools, town halls, and other data repositories eliminated, released, or nefariously altered.

Government regulators, network designers, and network operators have encouraged, or at least tolerated, a **generative Internet** and a generative PC. They have responded with a light touch to a rising number of problems occasioned by the many possible uses of information technology, proceeding carefully with secondary liability regimes or architectural mandates in an attempt to eliminate perceived harmful activities. As these activities increase - frustrating legitimate interests both on and off the Internet - such restraint will be tested at precisely the time that the emerging PC/Internet grid is amenable to behavior-controlling interventions. Part IV thus turns to that regulation-friendly grid and how it has come about after the consummately uncontrolled growth described in Part II.

IV. A Postdiluvian Internet

I have thus far described a highly generative PC and network combination - one whose components triumphed over their respective closed competitors in the 1980s and 1990s thanks to their power, flexibility, and accessibility to outside development. Tens of millions of people now have always-on Internet access, and that number is growing briskly. They can use that access to download new programs or participate in new online services, many of which are offered by developers who have no connection to or special arrangement with PC manufacturers, OS developers, or network service providers. This state of affairs is the emerging generative grid. And despite the interests of regulators in controlling individual misbehavior and of technologists in preventing a groundswell of service-compromising malware, this generative grid appears to be here to stay. But this appearance is deceiving: though the grid is here to stay, its generativity is under threat.

This Part describes fundamental changes to network and PC architecture that I believe are not only possible, but are also plausible, driven or enabled quite naturally by the collective, if unconcerted, pressures of the powerful regulators and soon-to-be security-conscious consumers described in Part III. Together, these changes stand to negate many of the generative features celebrated in Part II. After Part IV describes this emerging postdiluvian Internet, Part V describes what might be done about it.

A. Elements of a Postdiluvian Internet

Looking to the grid's future gives us a sense of what regulators and consumers want, as well as when these groups' interests and views will largely align, or at least will not paralyzingly conflict.

[*2014] 1. Information Appliances. - An "information appliance" is one that will run only those programs designated by the entity that built or sold it. In the taxonomy of generativity, an information appliance may have the leverage and adaptability of a PC, but its accessibility for further coding is strictly limited.

There are already several Internet-aware mainstream information appliances. A flagship example is TiVo, a digital video recorder that connects to a user's cable, satellite, or antenna feed to record television programs. [n149] TiVo also connects to a phone line or Internet connection to download program information daily. [n150] It is thus both a product and a service. Consumers who have TiVo nearly uniformly love it, and many say they would not want to watch television without it. [n151]

The designers of TiVo did not write its software from scratch; they implemented it on top of the highly generative GNU/Linux OS. n[152] Most TiVo owners do not realize that they have purchased a form of PC. There is no sign that an OS - in the generative sense of something open to third-party programs - is present. Users interact with TiVo using a remote control, n[153] and the limitations on TiVo's use are not easy to circumvent. n[154] Such limitations include the inability to create standardized digital output: TiVo's recorded programs cannot be copied over to a PC without TiVo's own DRM-restricted software. n[155]

TiVo works as reliably as a traditional appliance because its makers know, with much greater certainty than most PC manufacturers, the uses to which it can be put. This certainty is good for TiVo and its partner hardware manufacturers n[156] because it gives TiVo complete control over the combined product/service that it provides. It is good for consumers because they find TiVo useful and trustworthy. It is also satisfying to regulators because TiVo was designed to prevent contributions to the peer-to-peer problem. As discussed earlier, TiVo's recorded shows may only be saved to a standard VCR or DVD, or in a copy-protected PC format, rather than, as would be trivial for the **[*2015]** manufacturer to allow, digitally to a consumer's PC hard drive in an unprotected format or over the Internet to an arbitrary destination. n[157] Even though many consumers would no doubt like such a feature, TiVo has likely refrained from offering it in part because it has relationships with content providers n[158] and in part because of fears of secondary copyright liability. The general historical forbearance of Internet regulators notwithstanding, it would be entirely possible under Grokster to find distributor-like liability for digital video recorders that allow widespread copying and transmission of the programs they record, especially if such programs were tagged by their originators with a flag indicating that they did not want them to be widely shared. n[159] The manufacturer of TiVo competitor ReplayTV was sued for permitting this type of retransmission - to any other ReplayTV located on the Internet - and for including an automatic commercial skipping feature that TiVo lacks. n[160] ReplayTV filed for bankruptcy before the case concluded. n161 ReplayTVs made by the successor manufacturer lack both of these features, n[162] and TiVo is in the process of implementing a new feature that will allow content distributors to insist that a TiVo not save a recorded program for very long, allowing only time-shifting of the content by the consumer, not long-term librarying. n[163]

TiVo heralds growth in Internet-aware appliances that exploit the generativity of OSs and networks but that are not themselves generative to consumers or other end users. This lack of generativity does not alone bespeak a major shift in the fundamentals of the Internet/PC grid, though it does perhaps make for some lost collective potential. Consumers who might otherwise buy a television-aware PC - and who might find themselves pleasantly surprised that it can be adapted to uses that they did not contemplate at the time of purchase - instead will buy TiVos. In turn, there is less incentive for some coders to **[*2016]** create new uses because the audience of people who might benefit from those uses has decreased. Indeed, to the extent those uses depend on large uptake for their value - as, for example, online auctions do - they will be less likely to arise. The makers of TiVo might decide to make TiVo a hub for non-television-related activities - after all, it is a PC underneath, and it increasingly has always-on Internet access - but such a development would have to take place the way that development of new software took place on the proprietary CompuServe service: on the initiative of the company itself or through negotiated deals with third-party software developers. Of course, the makers of TiVo could choose to update the machine to be able to run third-party applications. In recent months TiVo has taken a very limited step in this direction, releasing a software developers' kit that allows a TiVo user to connect to a PC or server and interact in a limited way - essentially adding a web browser to TiVo's functionality without allowing third parties to reprogram the machine. n[164]

Similar patterns may be found for other new information appliances. For example, smartphones are mobile cellular telephones that are designed to surf the Internet and handle nontelephonic tasks like taking pictures and maintaining calendars and to-do lists. n[165] Some smartphones like the Palm Treo are based on general-purpose handheld computers and then add telephonic functionality; n[166] they can run applications from third parties, possibly to the chagrin of cellular operators whose service is subject to disruption should Treos be compromised by malicious code. Others, including some phones by Cingular, run a version of Windows, n[167] but are configured by the cellular carriers who sell them to run only specially "signed" software: customers cannot simply double-click their way to running software not approved by the cellular carrier. Beyond smartphones, some information appliances are more closely related to the PC.

For example, the Xbox is a powerful video game console produced by Microsoft. As a general-purpose device it has capacity for non-gaming applications, but, unlike a PC running Windows, it is generatively far less accessible: third-party hardware and software add-ons must be licensed by Microsoft, **[*2017]** and some portion of profits from their sale must be shared with Microsoft as royalties. [n168]

2. The Appliancized PC. - The PC is heading in the direction of these information appliances. The first step that OS makers have taken as a natural response to threats from viruses and worms is to caution users before they run unfamiliar code for the first time. Users have found that they can compromise their PCs simply by visiting the wrong webpage, by clicking on executable e-mail attachments, or by downloading malicious software. Microsoft Windows presents a security warning when a user tries to run "unknown" software, defined as software without a digital certificate recognized by Microsoft. [n169] In the most recent version of Windows (Windows XP), when a user attempts to run such an unknown program, or when one tries to execute automatically - perhaps when the user visits a webpage - the user is presented with the warning: "The publisher could not be verified. Are you sure you want to run this software?"

Users in many situations will not know how to answer this question reasonably. Unless the user was not expecting to run any software at that instant, she knows only that she wants to run the software so long as it works. How will the consumer know that it works unless she tries it? Frequently, she will assume that the computer ought to know the answer better than she does and find herself saying no to perfectly good software.

The consumer-choice solution is thus no panacea for the unintentional downloading of harmful applications. Consumers confused enough to click on a virus-laden e-mail will not likely be deterred by a warning box, especially if that warning box appears frequently when innocuous but unsigned software is run. Further, a signature alone means nothing: one could sign software with the equivalent of "Donald Duck." [n170] The user is simply not in the best position to determine what software is good and what software is bad. While not so effective at solving the fundamental generativity-as-vulnerability problem, **[*2018]** an architecture for digital software signing makes it easier for custodians of computers to appliancize them. Businesses can readily configure employees' computers to run only approved applications; so, too, can libraries, schools, and parents. This not only screens out some undesirable content, but also locks down the PC for uses that could be quite positive, even if not explicitly approved by the machines' custodians.

For those who buy their own computers and wish to operate them without a constant infusion of viruses, the next step of security, beyond a generic and ineffective warning, requires specific advice: does the OS maker think this software should run? Software publishers can readily offer an architecture through which to answer such a question. Microsoft has long maintained a digital signature program for such software as device drivers, which govern the interaction between the OS and certain external PC devices like printers, scanners, and cameras. [n171] Third-party vendors can write their own drivers and leave them unsigned, they can sign them on their own authority, or they can submit them to Microsoft for approval. In the third case, Microsoft tests the drivers and, if they pass Microsoft's test, signs them as approved for use. [n172]

In some respects, this sort of testing and approval is a positive development for consumers. Too often, PC users find their machines consumed by viruses and are baffled that such infection is not covered by the PC manufacturer's or OS vendor's warranties. The OS vendor can address this concern by promising some form of assistance with OS problems, so long as the consumer sets the computer not to run unsigned or unapproved software. An OS maker like Microsoft can also benefit because it is uniquely positioned to offer this value-added service, one that gives it first-order gatekeeping ability over every piece of software running on its machines. As it does with an information appliance like the Xbox, [n173] Microsoft could decide to charge a fee to approve third-party software were it to revisit something akin to a brief, informally stated, and quickly disavowed possibility of collecting a "vig," or small tax, on "every transaction over the Internet that uses Microsoft's technology." [n174] Alternatively, the OS maker could offer its approval for free, still benefiting as kingmaker and gaining helpful influence over the PC experiences a user is likely to have by assuring some minimum quality control. An independent software maker might chafe at having to obtain such approval, but it could always **[*2019]** choose to forgo approval and reduce the number of machines on which its software will run when users accept default settings designed to block unknown software.

To understand the full implications of these potential solutions - and why they are troubling - it helps to juxtapose this developing architecture with another new feature in OSs that is possible now that networking is so ubiquitous: automatic updating. [n175] This new feature, which appears in the latest Microsoft and Apple OSs and in many individual pieces of software, takes account of the fact that more and more PCs have always-on broadband Internet connections. From the moment the computer is first connected to the Internet, the feature is enabled for some software. For others, including Windows XP, the feature is off, but the computer prompts the user to turn it on. [n176] With automatic updating, the computer regularly checks - typically daily - for updates from the OS publisher and from the makers of any software installed on the PC. At first blush, this function is innocuous enough; it takes advantage of the networkability and adaptability of the PC to obtain the most recent security patches as they become available. Because it does not rely on consumer vigilance, this development solves some of the consumer maintenance problems noted as early as 1988, during the Morris worm incident, when many computer flaws went "unpatched."

So far Apple and Microsoft install automatically only security-related updates that they deem "critical"; updates to the "regular" functioning of the OS or auxiliary applications still require the consumer's approval. Many other makers of stand-alone software use automatic updating far more liberally, and there is no technical limit on what changes to a PC they can effect. They might not only update themselves, but also use the feature to download a small patch to other vendors' software, to install entirely new software, to upgrade the OS, or to eliminate installed software. Thus, the security benefits of automatic updating may well fail to justify the new vulnerabilities it creates, especially for producers of more obscure pieces of software whose update servers might be more easily compromised by third parties than the "bunkerized" versions run by OS makers. But whether or not **[*2020]** it addresses security concerns, automatic updating opens the door to an utter transformation of the way the Internet grid works.

With automatic updating, the OS and attendant applications become services rather than products. This transformation holds appeal for software makers, who can request or require consumers to subscribe to regular updates, much as those who purchase antivirus software are asked to subscribe annually to virus definition updates after a one-year grace period. Further, such updates help reduce software piracy: if a consumer does not validate his or her copy of the software or OS, the manufacturer can deactivate the software from a distance or can configure it to cease functioning if not properly renewed. [n177]

Automatic updating works in concert with appliancization, allowing manufacturers to see when their software has been hacked or altered - and to shut down or reinstall the original OS when they have. Exactly this happened with the Hughes DirecTV satellite receiver information appliance. Just before the Super Bowl in 2001, consumers who had hacked their DirecTV receivers to receive smartcard access found their receivers suddenly no longer working: the satellite had broadcast not only programming for people to watch, but programming for the receiver to obey. [n178] The receiver checked for particular hacks and, if they were found, self-destructed, rendering the affected cards entirely useless. By some reports, the last few computer bytes of the hacked smartcards were rewritten to read "Game Over." [n179]

Automatically updating software on PCs is becoming more common at the same time as the Internet itself becomes a host for highly controlled software. The emergence of the PC/Internet grid makes it easier for applications to be developed to run on remote servers rather than on the PC itself. A PC or information appliance equipped with only a web browser can now access a range of services - a development variously known as application streaming, web services, and Web 2.0. [n180] On one view, this development is generatively neutral, merely shifting the locus of generative software writing to a server on the Internet rather than on the PC itself - perhaps even avoiding the generative damage caused by PC lockdown. This shift, however, undermines **[*2021]** distributed PC processing power for novel peer-to-peer applications. [n181] It also carries many, but not all, of the drawbacks of automatic updating. From a security standpoint, a service updated at one location on the Internet may be much less likely to interfere with the user's enjoyment of a service offered elsewhere by another provider - unlike an automatic update by one PC application that can harm a concurrent PC application or disrupt the overall operation of the PC. However, this very isolation of services can also prevent generative building of software on other software. n182 Some Internet-hosted services maintain standardized interfaces to user data to permit outside development. For example, Google has so far allowed independent software developers to create "mash-ups" with their own virtual pushpins superimposed on Google's maps. n183 Google can withdraw this permission at any time and shut off the underlying service facilitating the mash-ups, keeping dependent generative applications in the realm of whimsy because any long-term or commercial development on such terms would be foolhardy. n184

B. Implications of a Postdiluvian Internet: More Regulability, Less Generativity

Consumers deciding between security-flawed generative PCs and safer but more limited information appliances (or appliancized PCs) may consistently undervalue the benefits of future innovation (and therefore of generative PCs). The benefits of future innovation are difficult to perceive in present-value terms, and few consumers are likely to factor into their purchasing decisions the history of unexpected information technology innovation that promises so much more just around the corner.

From the regulators' point of view, automatic updating presents new gatekeeping opportunities. Updates can be and have been used by manufacturers not only to add functionality, but also to take it away, at times apparently because of legal concerns. For example, an earlier version of Apple's iTunes software permitted users to stream **[*2022]** their music on the Internet, without permitting them to copy each others' music permanently. n185 Apple subsequently thought better of the feature, and in a later automatic update trimmed iTunes to permit streaming only to those on one's local network. n186 This capability has considerable implications for the content management and infringement problems discussed in Part III. At the time of Sony Corp. of America v. Universal City Studios, Inc., n187 it was no doubt difficult to imagine impounding consumers' VCRs as a remedy, should VCRs have been found instruments of contributory copyright infringement. But if Sony could have reprogrammed consumers' VCRs at a distance, at the direction of content owners and regulators, such a remedy might have been very tempting. Similar inspiration prompted a California district court to shape the way the Napster service functioned by ordering the company to make efforts to filter out unauthorized copyrighted files from its centralized directory of user-supplied offerings. n188

Professor Randal Picker sees automatic updating as transforming the information technology landscape and suggests that regulators should indeed exploit it. n189 For example, in the copyright context, he believes that a software author who builds automatic updating into a product that could facilitate infringement ought to have "a duty of ongoing design to reduce noninfringing use." n190 For those who fail to build in automatic updating in the first instance, Professor Picker suggests a "hard use test" designed to make it legally risky to release potentially infringing software without retaining programming control. n191 Professor Picker's argument is straightforward enough: once it becomes easy to revise distributed products to make them less harmful (in the eyes of regulators), why not encourage such revisions? But Professor Picker fails to take into account the generative loss from compelling software originators to retain exclusive control.

A current example is illustrative: MP3 players, including the iPod, are increasingly being used for radio-like broadcasts. Through so-called "podcasting," an owner of an MP3 player can lawfully **[*2023]** download, for listening purposes, a number of selected programs from the Internet at large. n192 The iTunes streaming feature could have been a significant contributor to the popular uptake of podcasting because it could have allowed people to share their favorite broadcasts widely. But because Apple withdrew the feature, its potential impact cannot be known. Although Apple's withdrawal was voluntary, many more generative developments might be lost as a result of legally compelled restrictions on such features.

Worse, Professor Picker's solution would be difficult to apply to group-written open-source software. n193 A regulatory judgment in favor of software as service would, if not carefully crafted, punish decentralized development processes, which in turn might reduce or eliminate entire categories of information technology innovation.

Furthermore, the logic of Professor Picker's argument for imposing gatekeeping responsibilities need not stop at a software author's own products. Consider the consequences if OS makers were held responsible for all applications running on their systems. For example, DeCSS, a program that decrypts DVDs, has been found to be an illegal circumvention tool under section 1201 of the DMCA. n194 Under threat of liability or direct government regulation, it would take little technical effort for Microsoft, using exactly the technologies that antivirus vendors use to screen for and eliminate malware, to send an update to its OS customers that would prevent DeCSS from running. Indeed, any vendor of antivirus software with automatically updating definitions could be so pressured.

To be sure, the potential for new uses does not always militate against trying to eliminate real infringements. But the advent of software as service shifts generative power away from the broad audiences at the edges of information technology deployment and toward the center - toward professional software authors constrained by vendors, regulators, and private litigants who can influence them. n195 This shift will be unfortunate if it tends to frustrate the vast creativity and energy of "edge" contributors of the sort described in Part II. n196

[*2024] Software as service may be inevitable for some markets as the networked grid becomes reliable and fast enough to warrant moving code away from the productized desktop. However, consumers may consistently undervalue the unanticipated future uses of open PCs, creating a degree of market failure. Further, the generative history of the Internet and the PC suggests that we should hesitate before we pressure software developers to include automatic updating and use it for regulatory purposes.

The "third strand" cyberlaw scholars, who first raised concerns about an era of trusted systems in the mid-1990s, foresaw software that would refuse to abide by a user's desires if those desires exceeded the permissions built into the technology. Such software exists, but as Part III explains, its controls are structurally weak when implemented on generative PCs. So long as the user can run unrestricted software and can use an open network to obtain cracked copies of the locked-down content, trusted systems provide thin protection. For instance, Microsoft's Windows Media Player contains a powerful rights management system, but it will still play content stripped of protective tags. Even if it should cease to do so - instead playing only certified content - users could still install third-party players that ignore such restrictions.

United States lawmakers recognized this loophole in 2002, proposing the Consumer Broadband and Digital Television Promotion Act n197 (CBDTPA). The Act would have set in motion a process by which technology makers would, among themselves, develop standards for flagging digital content restrictions and then be compelled to write software that would respect those restrictions. n198 It would apply to any "digital media device," meaning:

> Any hardware or software that -
>
> (A) reproduces copyrighted works in digital form;
>
> (B) converts copyrighted works in digital form into a form whereby the images and sounds are visible or audible; or
>
> (C) retrieves or accesses copyrighted works in digital form and transfers or makes available for transfer such works to hardware or software described in subparagraph (B). n199

If technology makers could not agree on a standard, the FCC was to set one for them. n200

[*2025] Had the CBDTPA passed, it would have been nothing short of a breathtaking intervention. The proposal was in fact so drastic that it appeared to be a mere warning shot - not actually intended to become law - by the legislators who proposed it. But the substantive insight represented by the sweep of the CBDTPA rings true: if trusted systems are truly to restrict access to content, the open and the closed cannot coexist. The idea behind the CBDTPA was not simply to regulate the way that software restricted what people could do, but to regulate the way that OSs restricted what software could do and in turn the way that hardware could restrict what OSs could do.

The security-and market-driven phenomena that this Part describes point to a technology configuration that already accomplishes many of the goals of the CBDTPA. Unlike the CBDTPA, these phenomena are not fiats to be grafted onto a vibrant, resistant marketplace. They are the marketplace, representing a sum across the technology and publishing industries, governments, and consumers. In essence, they point to a license to code that is issued by mainstream software makers but can be shaped by governments.

Such a license may not be hard to obtain. Like a driver's license or a cosmetologist's license that vouches for the basic training and identity of the holder, a license to code would exist, at first, simply to require a verifiable identity behind the production of software. It could be held by a software author to indicate permission to make new works and could also apply to each item of software itself so that regulators could revoke the license of individual works. Further, the creator could be identified easily and held accountable for creating a virus or for negligently allowing his or her identity to be stolen.

New software writers might find users skeptical of running their software at first. Like new traders on the auction site eBay, new producers would have to prove their worth among daring consumers, slowly building a reputation for trustworthiness. Alternatively, new producers could submit the software for testing, such as the sort of testing that Microsoft requires for those who want their device drivers to be signed or that some smartphone makers require before allowing programs to run on their devices.

Should approved software later turn out to enable undesirable behavior, a government could demand that an OS maker treat it as a virus and revoke its license - or even the license of its author. Different governments might make different judgments about the software and thus could ask OS makers to block the offending software only on PCs in their respective jurisdictions. In a country like China, a movement **[*2026]** toward free OSs like Linux [n201] - to save money and to avoid a sense of control by American OS firms - need not produce generativity if it is thought to interfere with government content-filtering objectives. The tools to lock down PCs might be implemented using free or proprietary code, and a consumer market of appliancized PCs or information appliances would make it much harder to circumvent censorship because third-party code giving rise to new, unfiltered overlay networks could not be run readily. A focus on generativity, rather than on free versus proprietary software, [n202] illuminates the implications for political censorship in one place flowing from seemingly unrelated security measures taken in another.

So far, this Part has discussed reasons why consumers, OS makers, and regulators might appreciate a world in which the PC is more locked down in certain ways - ways that strike at the heart of its generativity. To be sure, there is no basis on which to insist flatly that any tradeoff between regulability and generativity should favor the latter. But this is a false dichotomy if we can make the grid more secure without sacrificing its essential generative characteristics. Making progress on the security problem is difficult because the distributed nature of the Internet and individual ownership of PCs do not induce participants to internalize their negative security externalities. As Part V discusses, ISPs are not held economically accountable when their subscribers' computers fall victim to viruses. Similarly, individual users may not care if their compromised machines cause trouble for other, faraway users. Locking down the PC, although attractive from a regulatory point of view, is undesirable because of its effect on innovation: technical innovation will slow as third parties are squeezed out of the development cycle, and intellectual and artistic innovation will slow as some of the technical innovations forestalled are quite possibly ones that would enhance valuable expressive activities.

The existence of a widely distributed, multipurpose PC inviting third-party contribution permits innovation arbitrage, whereby firms that are slow to invent or improve information appliances find themselves leapfrogged by entrepreneurial PC software programmers with less invested in the status quo. For example, the free Internet telephony services offered by such PC applications as Skype and audio-enabled instant messenger programs serve as benchmarks for appliancized telephone services that connect to the Internet through wi-fi **[*2027]** networks to complete calls when possible, saving money by avoiding traditional telephone networks. Without the ability to distribute their software to PCs, the innovators behind Skype and its siblings would find it costly to establish themselves in the marketplace, competitive pressures on incumbents would ease, and innovation would slow. Thus, generative PCs and information appliances can complement each other - the former providing a fertile soil for innovation, the latter providing a stable, user-friendly instantiation of innovation. Even a firm like Skype can start small and then, due to its success on PCs, secure funding to jump into the appliance business. [n203] Without generative PCs, the engine of innovation would be greatly weakened. [n204]

V. How To Temper a Postdiluvian Internet

Part II of this Article describes ways in which the Internet and PC developed among hobbyists and amateurs as much as through traditional firms, thanks to the technologies' generativity. The technologies' adaptability, ease of mastery, and accessibility meant that any number of communities, responding to a variety of needs with a variety of business models, could produce new applications and distribute them freely. Many of these applications themselves enable creative expression and exchange, resulting in recursive generativity: open, adaptable OSs and networks provide the basis for new tools that are themselves open and adaptable for creative expression.

The emergence of broadband has meant that these applications can run either locally or remotely and still be instantly accessible. Part III of this Article describes how disruptive uses of Internet and PC generativity, in particular cybersecurity problems, stand to raise alarm and ignite a significant reaction. Part IV sketches how, paradoxically, the **[*2028]** very generative avenues opened by the merging of the PC and Internet into a grid include the means by which much greater control can and likely will be asserted.

The worst aspects of this future ought to be blunted even though they will benefit important stakeholders: The postdiluvian Internet creates new opportunities for regulators. It accords with, or is identical to, the plans of many technology industry firms. In addition, it is genuinely premised on new conveniences for consumers, allowing their desired activities to take place more reliably, though at the expense of unquantifiable future opportunities.

To evaluate the different paths information technology might take, we must bear in mind key contributors to its success: those who are creative and are inspired to express that creativity, whether through producing new code or code-enabled art. Amateurs, who produced applications that others overlooked, played a vital role in the rise of the Internet and the PC that Part II chronicles. In this sense, of course, "amateurs" are those who do what they do because they love to do it. The availability of tools through which amateurs could express creativity meant that code was written by parties other than those who had chosen lives of professional codewriting. Today, thanks to networked information technology and the recursively generative code produced in large part by amateurs, art can be produced and shared by people other than professional artists, citizens can engage in far-ranging dialogues with others whom they would not otherwise encounter, and people can work together from the four corners of the globe to produce intellectual projects of social and economic significance. [n205]

The most important opportunities for such creativity ought to be retained as the Internet evolves. But this will require those who support creative communities to make an important concession. They will have to face the reality that a free and open Internet, including open PCs distributed among tens of millions of consumers, is simply not possible unless the most pressing demands of countervailing regulatory forces are satisfied. It is now an opportune time for thoughtful **[*2029]** interventions in law and code. Matters are still in flux, and no stakeholder is too invested in any of the most locked-down versions of the postdiluvian Internet. Intervention can preserve and maybe even enhance generativity while making necessary progress toward stability.

This Part sketches ways in which current thinking by cyberlaw scholars on these issues is perhaps too constrained. It also describes some specific projects that could help solve some of the Internet's most pressing problems with as little constriction of its generative capacities as possible.

A. Refining Principles of Internet Design and Governance

1. Superseding the End-to-End Argument. - Saltzer, Reed, and Clark's 1984 paper on end-to-end design purported only

to stipulate a good heuristic for keeping networks simple. n206 Since then, the notion of end-to-end neutrality has been offered as a normative ideal of an Internet free from internal filtering. n207 Many cyberlaw scholars have taken up end-to-end as a battle cry for Internet freedom, n208 invoking it to buttress arguments about the ideological impropriety of filtering Internet traffic. Although these arguments are powerful, and although end-to-end neutrality in both its technical and political incarnations **[*2030]** has been a crucial touchstone for Internet development, end-to-end does not fully capture the overall project of maintaining generativity, which more fundamentally expresses the values that attracted cyberlaw scholars to end-to-end in the first place.

According to end-to-end theory, placing control and intelligence at the edges of a network maximizes network flexibility and user choice. n209 The political implication of this view - that end-to-end design preserves user freedom - depends on an increasingly unreliable presumption: whoever runs a machine at a given network endpoint can readily choose how the machine will work. For example, in response to a network teeming with viruses and spam, network engineers suggest more bandwidth (to make invisible the transmission of "deadweights" like viruses and spam) and better protection at user endpoints, rather than interventions by ISPs closer to the middle of the network. n210 But consumers are not well positioned to maintain their machines painstakingly against attack, leading them to prefer the locked-down PCs described in Part IV. Those who favor end-to-end principles because they favor generativity must realize that failure to take action at the network level may close some parts of the grid because consumers may demand, and PC manufacturers may provide, locked-down endpoint environments that promise security and stability with minimum user upkeep. Some may embrace a categorical end-to-end approach anyway: even in a world of locked-down PCs, there will no doubt remain non-mainstream generative computing platforms for professional technical audiences. But this view is too narrow. We ought to see the possibilities and benefits of PC generativity made available to everyone, including the millions of people who obtain PCs for current rather than future uses, but who end up delighted at the new uses to which they can put their machines.

Put simply, complete fidelity to end-to-end may cause users to embrace the digital equivalent of gated communities. Gated communities offer safety and stability to residents and a manager to complain to when something goes wrong. But from a generative standpoint, digital gated communities are prisons. Their confinement is less than obvious because what they block is generative possibility: the ability of outsiders to offer code and services to users, giving users and producers an **[*2031]** opportunity to influence the future without a regulator's permission. If digital gated communities become the norm, highly skilled Internet users of the sort who predominated in the mid-1980s will still be able to enjoy generative computing on platforms that are not locked down, but the rest of the public will not be brought along for the ride. For those using locked-down endpoints, the freedom in the middle of the network is meaningless.

Thus, strict loyalty to end-to-end neutrality should give way to a new generativity principle, a rule that asks that modifications to the PC/Internet grid be made when they will do the least harm to its generative possibilities. Under such a principle, for example, it may be preferable in the medium term to screen out viruses through ISP-operated network gateways rather than through constantly updated PCs. n211 Although such network screening theoretically opens the door to additional filtering that may be undesirable, this risk should be balanced against the very real risks to generativity inherent in PCs operated as services rather than products.

This generativity principle suggests at least two ways in which we might fundamentally reconceptualize the map of cyberspace. First, we must bridge the divide between those concerned with network connectivity and protocols, on the one hand, and those concerned with PC design, on the other - a divide that end-to-end unfortunately encourages. Such modularity in stakeholder competence and purview was originally a useful and natural extension of the Internet's hourglass architecture: it meant that network experts did not have to be PC experts and vice versa, just as the OS-application divide in the corresponding PC hourglass means that application developers need not know the ins and outs of PC peripherals and networking. But this division of responsibilities, which works so well for technical design, is crippling our ability to think through the trajectory of applied information technology. Now that the PC and the Internet are so inextricably intertwined, it is not enough for network engineers to worry only about network openness and assume that the endpoints can take care of themselves. It is abundantly clear that endpoints simply cannot.

Second, "middle" and "endpoint" are no longer subtle enough to capture the important features of the Internet/PC landscape. It remains correct that from a network standpoint, protocol designs and the ISPs that implement them are the "middle" of the network, as distinct from PC "endpoints." But the true import of a vernacular of "middle" and "endpoint" for policy purposes relates to individuals' power to control their experiences on the network. If consumers can no longer exercise meaningful control over their PC endpoints, instead **[*2032]** ceding such control to government or corporate authority such as an OS maker or a handful of security vendors, then the PCs become driven by a "middle" and their identities as endpoints diminish. Even today, consumers might not want or have the ability to fine-tune their PCs, and the taxonomy of generativity would say that such fine-tuning is not possible because the PCs are not easy for a mass audience to master even though they remain leveraged and adaptable. But there are a variety of methods by which PCs can compensate for the difficulty of mastery, only some of which require centralized control. For example, users might be able to choose from an array of proxies - perhaps Microsoft, Ralph Nader, or a public interest organization - for guidance on decisions about PC configuration. Now that the network's endpoints are controllable by faraway entities, abandoning the end-to-end debate's simplistic divide between middle and endpoint will enable us to identify and respond better to the emerging threats to the Internet's generativity.

2. Reframing the Internet Governance Debate. - Since the mid-1990s, an intense but misguided debate has raged over Internet governance. Those who care about the Internet's future are unduly occupied with domain names and the Internet Corporation for Assigned Names and Numbers (ICANN), the nonprofit organization chartered to administer top-level domain name policy. The existence of ICANN has proved a focus for debates about Internet governance for the circular reason that ICANN is an organization administering a particular - and quite limited - part of Internet functionality. The issues at stake in domain name assignment are real, but the focus on such disputes and on the governance of ICANN is myopic in relation to the threats posed by undue restriction of the Internet's generativity.

At the other extreme is the overbroad focus of the World Summit on the Information Society, a United Nations project that took place from 2003 to 2005. [n212] The World Summit included narrow debates about domain name management, but it also covered the larger dilemma of the global digital divide: how to ensure Internet access to as many people and peoples as possible. [n213] This cause is truly worthy, **[*2033]** but it makes Internet governance seem like merely a facet of international development policy.

Just as Internet architects should be encouraged to apply network design principles to PCs as PCs develop, groups engaged with issues of Internet governance should take note of the ways in which the Internet/PC grid is developing and grapple directly with how to maintain generativity. To proceed assuming that the primary challenge for Internet governance is narrowly one of managing administrative functions or is broadly one of deploying the network to additional people is to overlook the most important questions facing the Internet's future.

Part IV argues that OS makers or security firms may block the deployment of individual PC applications on behalf of PC users who crave security, creating broader bottlenecks to application deployment by anyone other than centralized kingmakers. The puzzle, then, is how to avoid these bottlenecks, whether coming from government or from private code-filtering schemes, while conceding that PC users can no longer be expected to exercise meaningful choice about code without help. A worthy Internet governance project to retain consumer choice without creating a new bottleneck could take the form of a grassroots campaign or public interest organization with participation from Internet architects. This project could set up a technical architecture to label applications and fragments of executable code, coupled with an organization to apply such labels nondiscriminatorily. Alternatively, the project could establish a distributed architecture by which the individual decisions about whether to run a given application, and the subsequent results, could serve as advice to others contemplating whether to run such code. The history of the Internet is seasoned with successful organizations devoted to such ends even though ideological views and regulatory agendas are often embedded in technical decisions. [n214] Public interest "underwriters' laboratories" for the Internet would reduce consumer demand for evaluations by OS makers or ISPs. Precisely because the lines separating viruses, spyware, poorly written software, and flat rent extraction by software authors are so blurry, they are best adjudicated using the sorts of quasi-public mechanisms that have served Internet development in the past. The alternative is to see such power accrete in a handful of private firms with incentives to become gatekeepers for purposes other than security.

[*2034] Perhaps most promising, the Internet grid itself can support the immediate collection and collation of useful information, which would then be passed from one PC to another, permitting PC users to make informed decisions about code. Tools can be developed to provide members of the general Internet public with simple but powerful information about the code they encounter. A tool of this sort could be a dashboard displaying information such as how many other computers in the world were running a candidate piece of software and whether their users were on average more or less satisfied with their computers than those who did not run it. A gauge that showed that a piece of software was nonexistent last week but is now unusually popular might signal to a cautious PC user to wait before running it. Professor Jean Camp and others have sketched the beginnings of a system collecting explicit user judgments about code. [n215] Such explicit user judgments - perhaps unreliable because many users do not know how their PCs work - could be augmented with automatically generated demographics, such as how often a PC reboots or generates pop-up windows, or implicit judgments that inexpert users might generate more reliably, such as how satisfied users are with their machines. By aggregating across thousands or millions of users, the dashboard can isolate and display the effects of a single piece of code.

3. Recognizing Interests in Tension with Generativity. - Those who have made the broad case for Internet freedom - who believe that nearly any form of control should be resisted - ought to be prepared to make concessions. Not only are many of the interests that greater control seeks to protect indeed legitimate, but an Internet and PCs entirely open to new and potentially dangerous applications at the click of a mouse are also simply not suited to widespread consumer use. If the inevitable reaction to such threats is to be stopped, its underlying policy concerns must in part be met.

There was a time in the Internet's development during which it made sense to eschew modalities of control because the network was experimental and because the harm that could be brought about by its misuse was limited. The network, after all, only carried bits. One might compare it to a hypothetical highly generative children's chemistry set, which would be adaptable and could be leveraged: it would contain chemicals that could accomplish a variety of tasks, with small **[*2035]** quantities adding up to big results if the user so desired. It would also be highly accessible: children would be able to learn how to use it. But such generativity would have a manifest downside risk: a chemical accident could be dangerous to the child or even to the entire neighborhood. [n216] A malicious child - or adult, for that matter - could wreak greater havoc as the set's generativity grew. The same principle, of course, applies to gene splicing kits, atom smashers, and many of the power tools at a local hardware store.

At an abstract level, then, one might ask: for a given technology, how much generativity can be maintained in the face of concessions to its threats? Knowing only that one technology is in the field of computer or information science and another is in the field of, say, physics, might be enough to imply that there ought to be room for more generativity for the former than for the latter because the risks of harm - particularly physical harm - from misuse or abuse of the former are structurally likely to be lower. The worst examples of the harm caused by wayward uses of applications and networks - including uses for invasion of privacy, financial scams, defamation, and copyright infringement - are less physical than that which could be caused by wayward uses of fertilizer, explosives, and petri dish cultures. [n217] With generative technologies in the physical sciences, on the one hand, the good applications may be much more identifiable ex ante and the harmful ones more diffuse, suggesting there is less to be gained from maintaining universal generativity. With the Internet, on the other hand, truly harmful applications are the known exceptions to be carved out from the universe of potential applications the technology enables. At the very least, then, if individual harms can be largely identified and then prevented or rectified by narrow interventions, these interventions would be preferable to undermining the generative nature of the PC and the Internet.

Now that the Internet is no longer experimental, an intuition that "it's only bits" is not so compelling. The degree of openness suited to, say, 1988 - when an inculcation of "professional ethics" among Internet users was thought to be the appropriate way to deal with virus writers [n218] - does not suit the present day.

[*2036] Some of the parties best suited to solve the problems of viruses, privacy invasion, spam, and copyright infringement may be sitting on the sidelines, unmotivated to help, and preferring an Internet that is entirely open because they can defend themselves from the Internet's worst harms thanks to their technological sophistication. If they do not contribute their creativity and effort to these problems, however, solutions less sensitive to generativity are all too likely to come about.

4. "Dual Machine" Solutions. - Part IV suggests that consumers, rightly fearful of security vulnerabilities latent in the **generative Internet**/PC grid, will demand a future in which locked-down information appliances predominate over generative PCs. One may seek the best of both worlds, however, by creating both generativity and security within a single device. To accomplish this compromise, we might build PCs with physical switches on the keyboard - switching between "red" and "green." n219 A PC switched to red mode would be akin to today's PCs: it would be capable of running any software it encountered. This mode would maximize user choice, allowing participation in unanticipated applications, such as PC-to-PC telephony, whose value in part depends on uptake by other users. Such a configuration would retain a structural vulnerability to worms and viruses, however. Hence the availability of green mode, by which the computer's processor would be directed to a different OS and different data within the same hardware. In green mode, the computer might run only approved or vetted software - less interesting, but much more reliable. The consumer could then switch between the two modes, attempting to ensure that valuable or sensitive data is created and stored in green mode and leaving red mode for experimentation. A crude division such as this has the benefit of being eminently understandable to the consumer - just as a driver can understand putting a sport utility vehicle into all-wheel drive for off-roading - while retaining much of the leverage and adaptability of today's PC.

But such PCs give rise to new problems. For example, ISPs might offer a lower rate for connecting a green PC and a higher rate for a red one - presuming the green to be less burdensome for customer service and less amenable to network abuse. Corporate environments might offer only green PCs and thus limit the audience for available innovation. Or the green PC might be so restrictively conceived that most users would find it unpalatable and would thus continue to choose between traditional PCs and vendor-specific information appliances. Even to hypothesize a green PC is to ask that some way be found to determine which software is suitable for use on an open PC and which **[*2037]** is not. A PC running multiple virtual machines n220 is a promising avenue, but it raises many of the same sorts of problems that a locked-down PC would encounter, although solutions to these problems might lie in a distributed, public-interest effort to sort red applications from green ones.

B. Policy Interventions

1. Enabling Greater Opportunity for Direct Liability. - As the capacity to inflict damage increases with the Internet's reach and with the number of valuable activities reliant upon it, the imperatives to take action will also increase. Intermediaries will be called to supervise because they provide a service capable of filtering user behavior. Preemptive reductions in PC or Internet generativity may also arise as it becomes easier to implement such changes over the grid.

One way to reduce pressure on institutional and technological gatekeepers is to make direct responsibility more feasible. Forthcoming piecemeal solutions to problems such as spam take this approach. ISPs are working with makers of major PC e-mail applications to provide for forms of sender authentication. n221 A given domain can, using public key encryption tools, authenticate that it is indeed the source of e-mail attributed to it. With Microsoft's Sender ID or something like it, e-mail purporting - but not proving - to be from a user at yahoo.com can be filtered as spam so easily that it will no longer be worthwhile to send. This regime will hold ISPs more accountable for the e-mail they permit their networks to originate because they will find themselves shunned by other ISPs if they permit excessive anonymous spam. This opportunity for more direct liability reduces the pressure on those processing incoming e-mail - both the designated recipients and their ISPs - to resort to spam filtration heuristics that may unintentionally block legitimate e-mail. n222

The same principle can be applied to individuals' uses of the Internet that are said to harm legally protected interests. From the point of view of generativity, music industry lawsuits against individual file sharers inflict little damage on the network and the PC themselves, even if they are bad policy because the underlying substantive law **[*2038]** demarcating the protected interest is itself ill-advised - as I believe it is. The Internet's future may be brighter if technical processes are refined to permit easier identification of Internet users, alongside legal processes - and perhaps technical limitations - to ensure that such identification is only made with good cause.

As discussed in section II.C, many Internet consumers have embraced wi-fi, and the wireless routers default to sharing the connection with anyone nearby who has a PC configured with a wireless antenna. Consumers may not intend to open their networks, but they carry generative benefits for those nearby without their own Internet access. n223 Usage by others does not typically impede the original consumer's enjoyment of broadband, but should outsiders use that connection, say, to send viruses or to pirate copyrighted files, the original consumer could be blamed when the Internet connection is traced. n224 As such examples arise and become well-known, consumers will seek to cut off others' access to their surplus network resources, and the manufacturers of wireless routers will change the default to "closed." If, however, genuine individual identity can be confirmed in appropriate circumstances, wi-fi sharing need not be impeded: each user will be held responsible for his or her own actions and no more. But insofar as technologically guaranteed anonymity is retained, more drastic means to eliminate individual wrongdoing through gatekeeper intervention wait in the wings.

2. Establishing a Choice Between Generativity and Responsibility. - To the extent that those who use generative platforms to invade legally protected interests can be held directly liable, maintainers of technology platforms - ISPs and newly service-like OS makers - should be encouraged to keep their platforms generative, rather than narrowing their offerings to facilitate regulatory control as Professor Picker suggests. n225

In turn, the more service-oriented and less generative the platform, the more legal responsibility we should impose on the technology provider to guarantee a functioning system. If a TiVo unit were not to operate as promised - suppose it simply crashed and failed to record any television programs - the law of warranty would quickly come into play. If the TiVo unit were new enough, the company would make good on a repair or replacement. Yet this simple exchange rarely takes place after the purchase of a new computer. Suppose a **[*2039]** new PC stops functioning: after a week of using it to surf the Internet and write articles, the consumer turns it on and sees only a blue error screen. n226 Unless smoke pours out of the PC to indicate a genuine hardware problem, the hardware manufacturer is likely to diagnose the problem as software-related. Because the user installed software after purchasing the machine, pinpointing the problem is not easy, and in particularly difficult cases, the OS maker will simply suggest a laborious and complete reinstallation of the OS, wiping clean all the changes that the consumer has made.

Hardware and OS makers are right that they ought to bear very little responsibility for this all-too-common problem because it is not clear that either the hardware or the OS is at fault. The mishmash of software found on even a two-week-old Internet-exposed PC thus precludes any obvious responsibility of a particular hardware or software manufacturer when problems arise.

Part IV argues that TiVo and the PC are converging. To the extent that PC OSs seek to screen what programs can run on them, the law should hold OS makers responsible for problems that arise just as TiVo and cellular phone manufacturers take responsibility for issues that arise with their controlled technologies. If the OS remains open to new applications by third parties, the maker's responsibility should be duly lessened. Such a regime permits technology vendors to produce closed platforms but encourages them to produce generative platforms by scaling liabilities accordingly. This tracks the intuition behind secondary theories of liability: technology makers may shape their technologies largely as they please, but the configurations they choose then inform their duties and liabilities.

This Part sketches a modest route by which the operation of law might appropriately further generativity. There are no doubt others, such as shaping consumer protection law to ensure that a shift from product to service does not permit a technology vendor to upset settled consumer expectations through a supposedly routine automatic product update that in fact substantially changes the benefit of the consumer's bargain. Each of these kinds of interventions is grounded in recognition that the law already influences the development of technology in manifold ways and thus can be adjusted to take generativity into account as an important end among other policy priorities.

VI. Conclusion

The modern Internet is at a point of inflection. This Article argues that its generativity, and that of the PC, has produced extraordinary **[*2040]** progress in information technology, which in turn has led to extraordinary progress in the development of forms of artistic and political expression. Internet architects and regulatory authorities have applauded this progress, but they are increasingly concerned by its excesses. The experimentalist spirit that fostered maximum generativity is out of place now that we rely upon the Internet and PCs for applications that we deem vital.

The challenge facing those interested in a vibrant global Internet is to maintain the core of that experimentalist spirit in the face of growing pressures. One path leads to a velvet divorce, creating two separate Internets with distinct audiences: a return to the quiet backwater for the original experimentalist Internet that would restart the generative cycle among researchers and hackers distinct from consumers who live with a new, controlled Internet experience. Two Internets would consign the existing grid to an appliancized fate, in which little new happens as existing technology players incrementally modify existing applications without the competitive pressure of grid-enabled innovation arbitrage.

The alternative paths that this Article advocates try to maintain the fundamental generativity of the existing grid while taking seriously the problems that fuel enemies of the Internet free-for-all. It requires charting an intermediate course to make the grid more secure - and to make some activities to which regulators object more regulable - in order to continue to enable the rapid deployment of the sort of amateur programming that has made the Internet such a stunning success.

Crippling generative accessibility and adaptability by transforming the PC into an information appliance is undesirable. So, too, are hamfisted clamps by ISPs upon network traffic in an effort to beat back viruses and other PC security threats, even as complete fidelity to end-to-end neutrality may on balance harm the generative information technology environment. Some limits are inevitable, and this Article attempts to point to ways in which these limits might be most judiciously applied. The key is to set such limits through thoughtful adjustments to accessibility that do not themselves spawn new centralized gatekeepers. The right interventions will preserve the public's ability to adopt new technologies from all corners, creating rough accountability for those who wish to introduce software to the world and for individuals who put that software to certain uses, while enabling those who maintain generative technologies - the Internet architects, ISPs, and OS publishers - to keep those technologies open and to ensure that those who wish to contribute to the global information grid can do so without having to occupy the privileged perches of established firms or powerful governments.

Legal Topics:

For related research and practice materials, see the following legal topics:
Civil ProcedureJudicial OfficersGeneral OverviewComputer & Internet LawCopyright ProtectionCivil Infringement ActionsOwner RightsAdaptationGovernmentsAgriculture & FoodProcessing, Storage & Distribution

FOOTNOTES:

n1. See Neil Randall, The Soul of the Internet 25-26 (1997); Living Internet, ARPANET - The First Internet, http://livinginternet.com/i/ii arpanet.htm (last visited Apr. 9, 2006). The first message sent over the system was intended to be "log"; the transmission crashed after the second letter, making "lo" the first Internet message. Randall, supra, at 26.

n2. See Randall, supra note 1, at 60 (describing efforts by Vinton Cerf and Robert Kahn to develop a protocol by which networks could connect to one another); Wikipedia, History of the Internet, http://en.wikipedia.org/wiki/History of the Internet (last visited Apr. 9, 2006) (same); see also Barry M. Leiner et al., The Past and Future History of the Internet, Comm. ACM, Feb. 1997, at 102, 104 (remarking that "the Internet was not designed for just one application but as a general infrastructure on which new applications could be conceived").

n3. See Brian E. Carpenter, Architectural Principles of the Internet (1996), http://www.ietf.org/ rfc/rfc1958.txt (describing the need for design to accommodate heterogeneous hardware); see also Leander Kahney, Jacking into Home Networking, Wired News, May 4, 2000, http:// www.wired.com/news/technology/0,1282,36078,00.html (describing a "residential gateway" device that allows consumers to establish home networks); Leander Kahney, Your Car: The Next Net Appliance, Wired News, Mar. 5, 2001, http://www.wired.com/news/technology/0,1282,42104,00. html (describing an embedded operating system that could enable various consumer appliances to connect to the Internet).

n4. For additional background on the history of the Internet, see Living Internet, supra note 1.

n5. See Internet World Stats, Internet Usage Statistics - The Big Picture, http://www. internetworldstats.com/stats.htm (last visited Apr. 9, 2006) (estimating that over one billion people used the Internet in 2005).

n6. See Steve Lohr, Microsoft To Offer Streamlined Products Aimed at Programmers, N.Y. Times, June 29, 2004, at C2 (estimating that there are approximately eighteen million amateur programmers worldwide, about three times the number of professional programmers).

n7. See Paul Freiberger & Michael Swaine, Fire in the Valley 78-79, 118-24 (2d ed. 2000) (describing the role of hobbyists and enthusiasts in establishing the market for PCs); Howard Rheingold, Innovation and the Amateur Spirit (Dec. 23, 1999), http://www.hrea.org/lists/ huridocs-tech/markup/msg00383.html (noting the role of amateurs in "creating a platform that had never existed before - the personal computer linked to a global network - before professionals could build industries on that platform"); cf. Robert Horvitz, Program Manager, Global Internet Policy Inst., ICT Applications, UNDESA-UNITAR "E-Government for Development" Seminar 2 (June 23-28, 2003), available at http://unpan1.un.org/intradoc/groups/public/ documents/un/unpan012242.pdf (noting that "the first PC manufacturers encouraged their customers to create new applications" and "saw amateur programmers and electronics hobbyists as the primary market for PCs").

n8. See, e.g., Mark A. Lemley & Lawrence Lessig, The End of End-to-End: Preserving the Architecture of the Internet in the Broadband Era, 48 UCLA L. Rev. 925 (2001); cf. Yochai Benkler, Freedom in the Commons: Towards a Political Economy of Information, 52 Duke L.J. 1245, 1266-67 (2003) (decrying the ability of ISPs to control content as an "affront to [individual] autonomy").

n9. See, e.g., Robert A. Badgley, Internet Domain Names and ICANN Arbitration: The Emerging "Law" of Domain Name Custody Disputes, 5 Tex. Rev. L. & Pol. 343 (2001); Tamar Frankel, Governing by Negotiation: The Internet Naming System, 12 Cardozo J. Int'l & Comp. L. 449 (2004); A. Michael Froomkin, Wrong Turn in Cyberspace: Using ICANN To Route Around the APA and the Constitution, 50 Duke L.J. 17 (2000).

n10. See Julie E. Cohen, Some Reflections on Copyright Management Systems and Laws Designed To Protect Them, 12 Berkeley Tech. L.J. 161 (1997) (advocating caution in adoption of laws protecting DRM); Pamela Samuelson, Intellectual Property and the Digital Economy: Why the Anti-Circumvention Regulations Need To Be Revised, 14 Berkeley Tech. L.J. 519 (1999) (arguing that the United States's DRM anticircumvention laws are overbroad and unclear).

n11. Cf. Freiberger & Swaine, supra note 7, at 200 (describing a combined word processor, spreadsheet, database, and programming language called Framework as a "remarkably powerful and advanced product" that "represented a "Swiss army knife' approach" to software design).

n12. See Wikipedia, Obfuscated Code, http://en.wikipedia.org/wiki/Obfuscated code (last visited Apr. 9, 2006) (describing the practical utility of obfuscated source code).

n13. See Paul E. Ceruzzi, A History of Modern Computing 81-84 (2d ed. 2003) (describing the insights that led to code stored on removable media).

n14. See Winn L. Rosch, Winn L. Rosch Hardware Bible 35-38 (6th ed. 2003).

n15. See Martin Campbell-Kelly, From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry 206-07 (2003); Wikipedia, History of Operating Systems, http://en.wikipedia.org/wiki/History of operating systems (last visited Apr. 9, 2006).

n16. These shortcuts can take the form of new functions. For example, a computer can be told simply to average a set of numbers, relieving the programmer of the more tedious job of explicitly asking it to sum the numbers and then divide by the number of numbers. Shortcuts can also be specific to the functions that a computer's nonprocessing hardware can perform. For example, rather than having to communicate directly with a printer and know exactly how it expects to receive data, the programmer simply can tell an OS to print, and the OS will take care of the rest.

n17. See Freiberger & Swaine, supra note 7, at 199.

n18. See Wikipedia, QBasic, http://en.wikipedia.org/wiki/QBasic (last visited Apr. 9, 2006) (noting that QBasic was shipped with MS-DOS 5.0 and later versions).

n19. See Rosch, supra note 14, at 45-49; Wikipedia, High-Level Programming Language, http:// en.wikipedia.org/wiki/High-level programming language (last visited Apr. 9, 2006).

n20. " Programmers" was a term best understood circularly as "those who programmed" rather than those who chose it as a career: many were autodidacts who programmed as a hobby or second job, rather than professionals trained in vocational environments. See, e.g., Freiberger & Swaine, supra note 7, at 164-65.

n21. Small software operations continue to populate the American software industry. As of 2004, the median number of employees per software establishment was a mere four and the average was thirty-nine. Furthermore, 54.5% of the 23,311 software companies in the United States had between two and four employees, and only 2.2% employed more than 100. At the same time, the 0.2% of software producers with over 5000 employees brought in around two-thirds of total industry revenues. See Software & Info. Indus. Ass'n, Software Industry Profile June 2004, at 1 (2004), available at http://www.siia.net/software/pubs/profile 0604.pdf. User expenditures on software applications (that is, software other than OSs) grew more than tenfold from $ 100 million in 1970 to $ 1.3 billion in 1980. The industry topped $ 17 billion in 1990 and maintained an average annual growth rate of nearly 15% over the course of the decade, reaching $ 63 billion in 2000. See Campbell-Kelly, supra note 15, at 14-15 tbl.1.1. The packaged software industry as a whole grew 12% per year over the 1990s, with growth closely tied to new PC sales. Software & Info. Indus. Ass'n, Packaged Software Industry Revenue and Growth 2 (2004), available at http://www.siia.net/software/pubs/growth software04.pdf.

n22. Cf. Comm. on the Internet in the Evolving Info. Infrastructure et al., The Internet's Coming of Age 36-38, 126-30 (2001) [hereinafter Coming of Age] (analogizing the architecture of the Internet to an hourglass because "the minimal required elements appear at the narrowest point, and an ever-increasing set of choices fills the wider top and bottom").

n23. Maintaining compatibility with a range of existing applications can be so important to a traditional OS publisher that it impedes the rollout of updated versions of the OS. See Steve Lohr & John Markoff, Windows Is So Slow, but Why?, N.Y. Times, Mar. 27, 2006, at C1.

n24. See, e.g., Jonathan Zittrain, Normative Principles for Evaluating Free and Proprietary Software, 71 U. Chi. L. Rev. 265 (2004) (describing and proposing a framework for evaluating the differences between free software and proprietary software); Henry Chu et al., Developing Nations See Linux as a Savior from Microsoft's Grip, L.A. Times, Aug. 9, 2004, at A4 (highlighting the recent adoption of Linux by governmental agencies in China and Brazil and providing an international perspective on the struggle for OS dominance between Microsoft and Linux); Lee Gomes, Linux Campaign Is an Uphill Battle for Microsoft Corp., Wall St. J., June 14, 2001, at B10 (describing Microsoft's response to the threat of Linux to its core OS business).

n25. At the extreme, one could even port GNU/Linux to Windows, just as Windows functionality has been ported to GNU/Linux. See Press Release, Linspire, Michael Robertson, CEO of Lindows.com, To Offer a PC Operating System To Run Both Linux and Windows Software (Oct. 23, 2001), available at http://www.linspire.com/lindows news pressreleases archives.php?id=1 (describing Lindows, now called Linspire, a product that ports Windows functionality to Linux); see also Andy Patrizio, Lindows: Linux Meets Windows, Wired News, Oct. 25, 2001, http:// www.wired.com/news/linux/0,1411,47888,00.html (evaluating the potential of a Linux-based alternative to Windows).

n26. The successors to IBM's word processor included the Friden Flexowriter, see Wikipedia, Friden Flexowriter, http://en.wikipedia.org/wiki/Friden Flexowriter (last visited Apr. 9, 2006), and "smart" typewriters found in office environments of the 1970s and 1980s.

n27. See Freiberger & Swaine, supra note 7, at 338-39.

n28. In 2003, console and portable game sales amounted to $ 5.8 billion, while PC game sales were $ 1.2 billion. See Press Release, NPD Group, Inc., The NPD Group Reports Annual 2003 U.S. Video Game Industry Driven by Console Software Sales (Jan. 26, 2004), available at http://npd.com/press/releases/press 040126a.htm. This statistic may not be indicative of the actual usage of PC games versus console games because of a presumably higher rate of PC game piracy.

n29. See NCR Corp., 2004 Annual Report 8-9 (2005), available at http://investor.ncr.com/ downloads/ncr2004ar.pdf (observing that recent growth in retail store automation technologies has been driven by self-service checkout systems and point-of-sale workstations).

n30. For commentary on recent consolidation in the information technology industry, see Andrew Ross Sorkin & Barnaby Feder, A Sector Where "Merger' Can Mean the Start of Something Ugly, N.Y. Times, Feb. 10, 2005, at C1.

n31. See Barry M. Leiner et al., A Brief History of the Internet (2003), http:// www.isoc.org/internet/history/brief.shtml (discussing the role of Internet protocols in addressing issues of data distribution capability, cost of distribution, and design flexibility in the development of the

Internet).

n32. See id. ("A key concept of the Internet is that it was not designed for just one application, but as a general infrastructure on which new applications could be conceived, as illustrated later by the emergence of the World Wide Web.").

n33. See Coming of Age, supra note 22, at 101 ("Increasing bandwidth in the Internet will provide adequate performance in many if not most circumstances... . [Increasing bandwidth] will enable more and more applications to run safely over the Internet, without requiring specific treatment, in the same way that a rising tide as it fills a harbor can lift ever-larger boats.").

n34. See id. at 36-38, 126-32 (describing how open standards and an "hourglass" architecture allow applications to use the Internet without requiring their authors to possess expertise on the network technology underlying the Internet).

n35. See Leiner et al., supra note 2, at 106 ("A key to the rapid growth of the Internet has been free and open access to the basic documents, especially the specifications of the protocols."); see also Coming of Age, supra note 22, at 124-26 (discussing the role of open standards in the growth of the Internet).

n36. See Leiner et al., supra note 2, at 103-04.

n37. J.H. Saltzer et al., End-to-End Arguments in System Design, 2 ACM Transactions on Computer Sys. 277 (1984).

n38. See id. at 277-86.

n39. Jonathan Zittrain, Internet Points of Control, 44 B.C. L. Rev. 653, 685-86 (2003).

n40. See Leiner et al., supra note 2, at 102-04.

n41. See Coming of Age, supra note 22, at 34-41 (discussing the design principles underlying the Internet that allowed for scalable, distributed, and adaptive design).

n42. See id. at 9-10, 98-106 (discussing quality of service goals such as reliability and robustness).

n43. See Leiner et al., supra note 2, at 103-04.

n44. See Wikipedia, Packet Switching, http://en.wikipedia.org/wiki/Packet switching (last visited Apr. 9, 2006).

n45. See Wikipedia, Routing, http://en.wikipedia.org/wiki/Routing (last visited Apr. 9, 2006); see also Leiner et al., supra note 31 (listing "gateway functions to allow [the network] to forward packets appropriately" as one of the Internet's design goals).

n46. See Freiberger & Swaine, supra note 7, at 208-09 (describing ARPANET as a network that "interconnected computers at Defense Department research and academic sites"); Christos J.P. Moschovitis et al., History of the Internet 126 (1999); Wikipedia, supra note 2.

n47. See Moschovitis et al., supra note 46, at 98-99, 102-03 (recounting the creation of Usenet, which resulted in "the emergence of newsgroups, in which people share ideas and information on specific topics").

n48. See Leiner et al., supra note 2, at 107-08.

n49. See id. at 105 ("NSF enforced an acceptable-use policy, prohibiting Backbone use for purposes "not in support of research and education."').

n50. See Wikipedia, CompuServe, http://en.wikipedia.org/wiki/Compuserve (last visited Apr. 9, 2006); Wikipedia, Prodigy (ISP), http://en.wikipedia.org/wiki/Prodigy %28ISP%29 (last visited Apr. 9, 2006).

n51. See Peter H. Lewis, A Boom for On-line Services, N.Y. Times, July 12, 1994, at D1.

n52. See William Glaberson, Press Notes: As On-Line "Circulation' Expands, More Newspapers Are Making the Plunge into Electronic Publishing, N.Y. Times, Oct. 10, 1994, at D5 (discussing partnerships between daily newspapers and online services).

n53. See Amy Harmon, Loyal Subscribers of Compuserve Are Fearing a Culture Clash in Its Takeover, N.Y. Times, Feb. 16, 1998, at D8.

n54. See Robert X. Cringely, That Does Not Compute!, PBS, Sept. 17, 1997, http://www.pbs.org/ cringely/pulpit/pulpit19970917.html (discussing the challenges facing proprietary services due to their technological inflexibility).

n55. See, e.g., Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1514-16 (9th Cir. 1992) (describing the security and licensing mechanisms used to control development of software for the Sega Genesis console).

n56. See Freiberger & Swaine, supra note 7, at 24-25 (noting that mainframes and minicomputers were the only types of computers that existed in the early 1970s).

n57. See NetAction, The Origins and Future of Open Source Software, http://www.netaction.org/ opensrc/future/unix.html (last visited Apr. 9, 2006) (stating that workstation manufacturers began shipping systems with "built-in" Internet protocols in 1987 and 1988).

n58. See Leiner et al., supra note 2 ("Widespread development of local-area networks (LANs), PCs, and workstations in the 1980s allowed the nascent Internet to flourish.").

n59. See Trumpet Software International, History, http://www.trumpet.com.au/history.html (last visited Apr. 9, 2006).

n60. See Wikipedia, Winsock, http://en.wikipedia.org/wiki/Winsock (last visited Apr. 9, 2006).

n61. See Walled Gardens - A Brick Wall?, Shosteck Email Briefing (Herschel Shosteck Assocs., Ltd.), Mar. 2000, http://www.shosteck.com/news/mar00.htm ("No matter how good the [America Online] proprietary content and services were, users demanded access to the millions of websites available on the world wide web, and Internet email."); see also Harmon, supra note 53 ("Compuserve's era as the home of choice for the technological elite really ended ... when the service failed to quickly offer subscribers a path to the World Wide Web.").

n62. See Walled Gardens - A Brick Wall?, supra note 61 ("While [America Online] continues to gain revenue from its proprietary e-commerce services and advertising relationships, the firm's majority appeal is as an easy on-ramp to the Internet - in essence, an access provider with much less emphasis on specific content and services.").

n63. See Stephen C. Miller, Point, Click, Shop Till You Drop, N.Y. Times, Apr. 20, 1995, at C2.

n64. See, e.g., ICQ, The ICQ Story, http://company.icq.com/info/icqstory.html (last visited Apr. 9, 2006).

n65. See Moschovitis et al., supra note 46, at 153-54, 164-65; Living Internet, Tim Berners-Lee, Robert Cailliau, and the World Wide Web, http://livinginternet.com/w/wi lee.htm (last visited Apr. 9, 2006).

n66. See Coming of Age, supra note 22, at 146; Randall, supra note 1, at 89-95; Living Internet, Email History, http://livinginternet.com/e/ei.htm (last visited Apr. 9, 2006).

n67. See Moschovitis et al., supra note 46, at 181.

n68. See Associated Press, Security-Free Wireless Networks, Wired News, May 30, 2004, http://www.wired.com/news/wireless/0,1382,63667,00.html. Indeed, in 2006 a firm called FON began facilitating the swapping of Internet connections among people, asking that they open their home wireless networks to members of FON in exchange for being able to access other FON members' wireless connections when traveling themselves. See John Markoff, Venture for Sharing Wi-Fi Draws Big-Name Backers, N.Y. Times, Feb. 6, 2006, at C3; What is FON?, http:// en.fon.com/info/what-is-fon.php (last visited Apr. 9, 2006).

n69. See Ken Belson, Yahoo To Offer Portal Service to BellSouth, N.Y. Times, Oct. 18, 2005, at C5 (detailing price competition among providers of broadband Internet access); see also Coming of Age, supra note 22, at 46 (describing the growth of the ISP market). In addition, several cities have implemented or announced efforts to provide their residents with free or subsidized Internet access. See, e.g., Moschovitis et al., supra note 46, at 126 (discussing a free public Internet access program in Cleveland); Bob Tedeschi, E-Commerce Report: What Would Benjamin Franklin Say? Philadelphia Plans Citywide Free Wi-Fi Internet Access for Computer Users, N.Y. Times, Sept. 27, 2004, at C8 (describing Philadelphia's plan to offer free wireless access in public areas).

n70. See, e.g., Motorola, Inc., A Business Case for T1 Voice and High-Speed Ethernet Commercial Services 4 (2005), available at http://broadband.motorola.com/ips/ pdf/CommSvcs.pdf (finding that T1 service costs range from $ 250 to $ 1200 per month across the United States); Ross Wehner, Out of the Shadows, Denver Post, Sept. 30, 2005, at 1C (noting that T1 lines typically cost $ 400 to $ 600 per month).

n71. See About.com, QoS, http://compnetworking.about.com/od/networkdesign/l/bldef qos.htm (last visited Apr. 9, 2006) (stating that the goal of QoS is to "guarantee[] ... the ability of a network to deliver predictable results" and stating that "elements of network performance within the scope of QoS often include availability (uptime), bandwidth (throughput), latency (delay), and error rate"); see also Coming of Age, supra note 22, at 9-10, 98-106 (discussing QoS goals); Wikipedia, Quality of Service, http://en.wikipedia.org/wiki/Quality of service (last visited Apr. 9, 2006) (discussing the elements of QoS and mechanisms by which networks provide QoS).

n72. Akamai is a leading provider of such edge-caching services. See Akamai Technologies, Inc., Annual Report (Form 10-K), at 3-5 (Mar. 16, 2006), available at http://www.akamai.com/en/ resources/pdf/investors/10k 2005.pdf (describing Akamai's service offerings). Internap is another leading provider of optimized throughput services for Internet publishers. Internap Network Services, Inc., Annual Report (Form 10-K), at 1-3 (Mar. 10, 2006).

n73. See, e.g., James Fallows, A Journey to the Center of Yahoo, N.Y. Times, Nov. 6, 2005, 3, at 3 (noting that Yahoo! estimates that its websites account for thirteen percent of all page views). For a current list of the parent companies of the ten websites most visited by users in the United States, see Nielsen//NetRatings, NetView Usage Metrics, http://www.nielsen-netratings.com/ news.jsp?section=dat to&country=us (follow "Weekly Top 10 Parent Companies" hyperlinks under "Home Panel" and "Work Panel" headings) (last visited Apr. 9, 2006).

n74. See, e.g., Randall, supra note 1, at 324 (noting Microsoft's delayed entry into the web browser market); Joshua Quittner, Billions Registered, Wired, Oct. 1994, at 50, available at http://www.wired.com/wired/archive/2.10/mcdonalds.html (noting the failure of several large companies to register their namesake Internet domain names).

n75. See FCC, Wireline Competition Bureau, High-Speed Services for Internet Access: Status as of December 31, 2004, at 6 (2005), available at http://www.fcc.gov/ Bureaus/Common Carrier/Reports/FCC-State Link/IAD/hspd0705.pdf.

n76. See George Johnson, Supercomputing "@Home' Is Paying Off, N.Y. Times, Apr. 23, 2002, at F1 (describing distributed supercomputing projects such as SETI@home, distributed.net, and Genome@home).

n77. See Jonathan Zittrain, Searches and Seizures in a Networked World, 119 Harv. L. Rev. F. 83, 85 (2006), http://www.harvardlawreview.org/forum/issues/119/dec05/zittrainfor05.pdf.

n78. See, e.g., Ann Bartow, Electrifying Copyright Norms and Making Cyberspace More Like a Book, 48 Vill. L. Rev. 13 (2003) (arguing that legislation is needed to make digital copyright follow the rules of analog copyright because existing rules do not appropriately constrain Internet users); Dan L. Burk, Trademarks Along the Infobahn: A First Look at the Emerging Law of Cybermarks, 1 Rich. J.L. & Tech. 1 (1995), http://www.law.richmond.edu/jolt/v1i1/burk.html (arguing for the use of trademark law to resolve disputes concerning domain names and similar issues); Austan Goolsbee & Jonathan Zittrain, Evaluating the Costs and Benefits of Taxing Internet Commerce, 52 Nat'l Tax. J. 413 (1999) (noting the difficulty that localities encounter in taxing Internet commerce and presenting empirical evidence suggesting that current legislation is not effective); I. Trotter Hardy, The Ancient Doctrine of Trespass to Web Sites, 1996 J. Online L. art. 7, http://www.wm.edu/law/publications/jol/95 96/hardy.html (arguing for the application of trespass law to the Internet to protect websites as property); Donald J. Karl, Comment, State Regulation of Anonymous Internet Use After ACLU of Georgia v. Miller, 30 Ariz. St. L.J. 513 (1998) (noting repercussions for Internet freedom in light of a 1997 court decision).

n79. See, e.g., Goolsbee & Zittrain, supra note 78, at 424-25 (suggesting that Internet commerce ought to be taxed under the same regime as other commerce); I. Trotter Hardy, The Proper Legal Regime for "Cyberspace," 55 U. Pitt. L. Rev. 993 (1994) (highlighting mismatches between existing doctrine and activities in cyberspace and suggesting that new rules may be necessary).

n80. See David R. Johnson & David G. Post, And How Shall the Net Be Governed?: A Meditation on the Relative Virtues of Decentralized, Emergent Law, in Coordinating the Internet 62 (Brian Kahin & James H. Keller eds., 1997) (arguing that the Internet should be governed by new mechanisms because it cannot be controlled by an existing sovereign); David R. Johnson & David Post, Law and Borders - The Rise of Law in Cyberspace, 48 Stan. L. Rev. 1367 (1996) (arguing that the Internet's cross-jurisdictional functioning makes regulation by small jurisdictional units impossible). Several scholars then argued against Johnson and Post's exceptionalist view of the Internet. See, e.g., Jack L. Goldsmith, Against Cyberanarchy, 65 U. Chi. L. Rev. 1199 (1998); see also A. Michael Froomkin, Time To Hug a Bureaucrat, 35 Loy. U. Chi. L.J. 139, 144 (2003) (suggesting skepticism regarding Internet self-regulation and arguing that "for most e-commerce, there really isn't an "Internet' in any useful sense any more than there is "telephone space'; rather, the Internet is just another quicker, better way of passing information between machines and people"); Allan R. Stein, The Unexceptional Problem of Jurisdiction in Cyberspace, 32 Int'l Law. 1167, 1191 (1998) (arguing that the problem of Internet jurisdiction arises from the quantity of transactions, not the quality, and that Internet jurisdiction is therefore "not uniquely problematic"). Professor Post later responded to these arguments. See David G. Post, Against "Against Cyberanarchy," 17 Berkeley Tech. L.J. 1365 (2002).

n81. See Lawrence Lessig, Code and Other Laws of Cyberspace 19-20 (1999).

n82. See id. at 43-60.

n83. See id. at 127-30, 135-38 (remarking that "when intellectual property is protected by code, ... nothing requires the owner to grant the right of fair use" and that "fair use becomes subject to private gain"); Cohen, supra note 10; Pamela Samuelson, DRM [and, or, vs.] the Law, Comm. ACM, Apr. 2003, at 41, 42 (highlighting the control over copyrighted works available to users of DRM); see also P. Bernt Hugenholtz, Code as Code, or the End of Intellectual Property as We Know It, 6 Maastricht J. Eur. & Comp. L. 308, 308 (1999) (remarking that "contract and "code' combined have the capability of making copyright and its set of statutory limitations largely redundant, and may require an entire new body of information law to safeguard the public domain").

n84. For a general discussion of trusted systems, see Mark Stefik, The Internet Edge 55-78 (2000); and Jonathan L. Zittrain, Technological Complements to Copyright (2005).

n85. See Dan L. Burk, Anticircumvention Misuse, 50 UCLA L. Rev. 1095, 1106-07 (2003) (stating that "the DRM anticircumvention provisions ... enable a new form of exclusive right" that "[is] entirely separate from the exclusive rights under copyright"); Julie E. Cohen, Pervasively Distributed Copyright Enforcement, 95 Geo. L.J. (forthcoming Nov. 2006) (manuscript at 2, on file with the Harvard Law School Library) (stating that "the emerging regime of pervasively distributed copyright enforcement is not simply aimed at defining the boundaries of legal entitlements, nor at creating and rationalizing information flows within markets" and that "it seeks to produce not only willing vendors and consumers, but also tractable ones, and it seeks these changes not merely at the behavioral level, but at the infrastructural level as well"); Cohen, supra note 10, at 177 (noting that DRM systems "adopted to protect digital works will prevent some actions that copyright law allows"); Samuelson, supra note 83, at 42 (stating that "DRM permits content owners to exercise far more control over uses of copyrighted works than copyright law provides").

n86. 17 U.S.C. 109(a)-(b) (2000).

n87. Pub. L. No. 105-304, 112 Stat. 2860 (1998) (codified as amended in scattered sections of 17 U.S.C.).

n88. See generally Elec. Frontier Found., Unintended Consequences: Five Years Under the DMCA 1-2 (2003), available at http://www.eff.org/IP/DMCA/unintended consequences.pdf (arguing that the DMCA chills free expression); Burk, supra note 85, at 1102-10 (explaining the legislative history and purposes of the DMCA's anticircumvention protections); Samuelson, supra note 83, at 42 (providing a brief overview of the DMCA).

n89. 17 U.S.C. 1201, 1203, 1204 (2000). Outside the DRM context, however, the exclusive right to control "access" to works is not one that copyright holders enjoy. See 17 U.S.C. 106 (2000 & Supp. II 2002).

n90. See Bill Rosenblatt, 2004 Year in Review: DRM Technologies, DRM Watch, Dec. 29, 2004, http://www.drmwatch.com/drmtech/article.php/3453001; Bill Rosenblatt, 2003 in Review: DRM Technology, DRM Watch, Dec. 31, 2003, http://www.drmwatch.com/drmtech/article.php/ 3294391.

n91. See Elec. Frontier Found., DMCA Archive, http://www.eff.org/IP/DMCA (last visited Apr. 9, 2006) (listing recent litigation under the DMCA); see also DVD Copy Control Ass'n, Inc. v. Bunner, 10 Cal. Rptr. 3d 185 (Ct. App. 2004) (evaluating a claim under California trade secret law against an individual who used decryption software known as DeCSS to access a DVD equipped with anticircumvention technology); Elec. Frontier Found., Norway v. Johansen, http://www.eff. org/IP/Video/Johansen DeCSS case (last visited Apr. 9, 2006) (summarizing a case involving a Norwegian teenager who was prosecuted for using DeCSS to access a DVD equipped with anticircumvention technology).

n92. See Motion Picture Ass'n of Am., Content Protection Status Report (2002), available at http://judiciary.senate.gov/special/content protection.pdf.

n93. DRM will likely never be able to combat piracy effectively within the current generative information technology grid. See Stuart Haber et al., If Piracy Is the Problem, Is DRM the Answer?, in Digital Rights Management 224, 224 (Eberhard Becker et al. eds., 2003) (stating that "if even a small fraction of users are able to transform content from a protected to an unprotected form, then illegitimate distribution networks are likely to make that content available ubiquitously").

n94. See Wikipedia, FairPlay, http://en.wikipedia.org/wiki/FairPlay (last visited Apr. 9, 2006).

n95. See id. (describing the restrictions that FairPlay imposes).

n96. In late 2005, Sony BMG Music produced audio CDs that included software that could run if the CD were inserted into a PC for playback. See Tom Zeller Jr., The Ghost in the CD, N.Y. Times, Nov. 14, 2005, at C1. The software installed a DRM system onto the PC and then copied the CD's music in a protected format not compatible with the popular Apple iPod. Sony suggested that users employ the "indirect" method of burning and re-ripping in order to produce music that could be played on an iPod. See Sony BMG Music Entertainment, Frequently Asked Questions, http://cp.sonybmg.com/xcp/english/faq.html#ipod (last visited Apr. 9, 2006) (encouraging users to write to Sony BMG for instructions on this method); see also Zeller, supra (noting that Sony BMG provides instructions for how to load FairPlay-incompatible tracks onto an iPod).

n97. See Jonathan Zittrain, A History of Online Gatekeeping, 19 Harv. J.L. & Tech. (forthcoming Spring 2006) (manuscript at 2-5, on file with the Harvard Law School Library).

n98. See Lawrence Lessig, Open Code and Open Societies: Values of Internet Governance, 74 Chi.-Kent L. Rev. 1405, 1415 (1999) (stating that the Internet's bottom-up evolution through "open code" allows for rapid improvements in technology and allows the market, rather than the government, to determine which ideas are best); see also Lessig, supra note 81, at 24-42 (arguing that the Internet will adopt an architecture of control even in the absence of regulation).

n99. See, e.g., A & M Records, Inc. v. Napster, Inc., 114 F. Supp. 2d 896 (N.D. Cal. 2000), aff'd in part and rev'd in part, 239 F.3d 1004 (9th Cir. 2001); UMG Recordings, Inc. v. MP3.com, Inc., 92 F. Supp. 2d 349 (S.D.N.Y. 2000). In contrast, Third Voice, a controversial browser plug-in that allowed users to annotate - or vandalize, as some saw it - webpages with "stickies," shut down not because of lawsuits but because of financial difficulties. See Aparna Kumar, Third Voice Trails Off... ., Wired News, Apr. 4, 2001, http://www.wired.com/news/business/0,1367,42803,00. html.

n100. 125 S. Ct. 2764 (2005). I explore in detail the history and trajectory of regulatory interventions to prevent defamation and copyright infringement in Zittrain, supra note 97.

n101. See OpenNet Initiative, Documenting Internet Content Filtering Worldwide, http://www. opennet.net/modules.php?op=modload&name=Archive&file=index&req =viewarticle&artid=1 (last visited Apr. 9, 2006); Jonathan Zittrain & Benjamin Edelman, Documentation of Internet Filtering Worldwide, http://cyber.law.harvard.edu/filtering (last visited Apr. 9, 2006).

n102. Bob Sullivan, Remembering the Net Crash of '88, MSNBC, Nov. 2, 1998, http://www. msnbc.com/news/209745.asp?cp1=1.

n103. See id. (noting that only 60,000 nodes were connected to the Internet and that Morris's software affected primarily universities and research centers); see also Patricia Wallace, The Internet in the Workplace 35-37 (2004) (discussing changes in computer configuration in the workplace and contrasting PCs with a mainframe setup).

n104. See Joyce K. Reynolds, The Helminthiasis of the Internet 1-2 (1989), available at http://www.faqs.org/ftp/rfc/pdf/rfc1135.txt.pdf; Sullivan, supra note 102. For more on how worms behave, see Eugene H. Spafford, Crisis and Aftermath, 32 Comm. ACM 678, 678-84 (1989).

n105. See U.S. Gen. Accounting Office, GAO/IMTEC-89-57, Computer Security: Virus Highlights Need for Improved Internet Management 8 & n.1, 13-14 (1989) [hereinafter GAO Report], available at ftp://coast.cs.purdue.edu/pub/doc/morris worm/GAO-rpt.txt; Sullivan, supra note 102.

n106. See Reynolds, supra note 104, at 1; John Markoff, Computer Invasion: "Back Door' Ajar, N.Y. Times, Nov. 7, 1988, at B10.

n107. See GAO Report, supra note 105, at 8, 17; Markoff, supra note 106.

n108. Reynolds, supra note 104, at 3-4.

n109. Id.; Sullivan, supra note 102.

n110. Spafford, supra note 104, at 678; Sullivan, supra note 102.

n111. United States v. Morris, 928 F.2d 504, 505-06 (2d Cir. 1991). For a general discussion of United States v. Morris, see Susan M. Mello, Comment, Administering the Antidote to Computer Viruses: A Comment on United States v. Morris, 19 Rutgers Computer & Tech. L.J. 259 (1993).

n112. See Wallace, supra note 103, at 36.

n113. Morris reportedly considered writing a "worm killer" to remove worms from computers once he realized the damage that the worm was causing. He did not, however, because he felt that he had already caused enough damage. See Brief for Appellant at 18-19, Morris (No. 90-1336), 1990 WL 10029997.

n114. See Spafford, supra note 104, at 678-81.

n115. See id. at 680.

n116. See GAO Report, supra note 105, at 28.

n117. See Reynolds, supra note 104, at 2-3.

n118. See GAO Report, supra note 105, at 19-21 (noting that "each host site is responsible for establishing security measures adequate to meet its needs").

n119. See Reynolds, supra note 104.

n120. See id. at 5-8.

n121. On the origin of CERT, see Howard F. Lipson, CERT Coordination Ctr., Special Report CMU/SEI-2002-SR-009, Tracking and Tracing Cyber-Attacks: Technical Challenges and Global Policy Issues 5 (2002), available at http://www.cert.org/ archive/pdf/02sr009.pdf; and CERT, Frequently Asked Questions, http://www.cert.org/faq/ cert faq.html (last visited Apr. 9, 2006).

n122. See supra section II.C, pp. 1987-94.

n123. See Harmon, supra note 53 (describing CompuServe as "an information service").

n124. See Ron Rosenbaum, Secrets of the Little Blue Box, Esquire, Oct. 1971, at 116, 119. For an account about the individual who claimed to be the original "phone phreaker," see James Daly, John Draper, Forbes, June 3, 1996, at 138.

n125. See Rosenbaum, supra note 124, at 120.

n126. See id. at 119-20.

n127. See Amy Harmon, Defining the Ethics of Hacking, L.A. Times, Aug. 12, 1994, at A1.

n128. See Coming of Age, supra note 22, at 107-24 (describing the Internet as "a set of independent networks interlinked to provide the appearance of a single, uniformed network" and explaining the architecture of the Internet).

n129. Increases in computer crime have received attention from the hacker community and have influenced hackers' behavior. See Harmon, supra note 127 (chronicling one self-described hacker's efforts to "stay[] on the right side of the blurry line that separates hacking from criminal behavior").

n130. See Steve Lohr, A Virus Got You Down? Who You Gonna Call?, N.Y. Times, Aug. 12, 1996, at D1 (stating that most viruses are not deliberately destructive).

n131. Estimates of the cost of the Morris attack vary widely. Compare Kevin Commins, Insurers Plan Computer Virus Coverage, J. Com. & Com., June 8, 1989, at 1A ($ 1.2 billion), with Panel Speculates on Rogue Hacker's Motives, St. Louis Post-Dispatch, Apr. 2, 1989, at 10A ($ 96 million). The estimates likely vary because they are based on relatively soft variables such as estimates of staff time, the number of computers affected, and productivity loss.

n132. For an analysis of hackers, see Clifford Stoll, The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage (1989). The "hacker ethic" is commonly defined by "the belief that system-cracking for fun and exploration is ethically acceptable as long as the hacker commits no theft, vandalism, or breach of confidentiality." Wikipedia, Hacker Ethic, http://en.wikipedia.org/wiki/Hacker ethic (last visited Apr. 9, 2005); see also Pekka Himanen, The Hacker Ethic and the Spirit of the Information Age (2001) (providing an in-depth exploration of the hacker ethic).

n133. In fact, more U.S. households now connect to the Internet using broadband than using dial-up. See Press Release, FCC, FCC Report Shows Strongest Ever American Broadband Market 1 (Sept. 9, 2004), available at http://hraunfoss.fcc.gov/edocs public/attachmatch/DOC-251959A2. pdf.

n134. See Luke Dudney, SANS Inst., Internet Service Providers: The Little Man's Firewall (2004), available at http://www.sans.org/rr/whitepapers/casestudies/1340.php (discussing port blocking, packet blocking, and other methods that ISPs could employ to prevent the spread of computer viruses, and running an experiment to assess the extent of the "open proxy" problem).

n135. Id. at 4.

n136. Id. at 5.

n137. See id. at 4.

n138. See CERT Coordination Center, CERT/CC Statistics 1988-2005, http://www.cert.org/ stats#incidents (last visited Apr. 9, 2006).

n139. See id. Other studies have noted the exploding number of incidents of application attacks as a threat, as websites increasingly link webpages to company databases. See, e.g., Bee Ware SAS, The Risk of Application Attacks Securing Web Applications 1-2 (2005), available at http://www.securitydocs.com/pdf/2839.PDF.

n140. See Sullivan, supra note 102; Internet Sys. Consortium, ISC Domain Survey: Number of Internet Hosts, http://www.isc.org/index.pl?/ops/ds/host-count-history.php (last visited Apr. 9, 2006).

n141. See Internet Sys. Consortium, supra note 140.

n142. See Press Release, Computer Indus. Almanac Inc., Mobile PCs In-Use Surpass 200M (June 20, 2005), available at http://www.c-i-a.com/pr0605.htm.

n143. See, e.g., Associated Press, A New Computer Virus Is Making the Rounds, N.Y. Times, June 20, 2000, at C6 (describing the Love Bug virus that struck in May 2000).

n144. See Worm Brings Down PC's and Networks, N.Y. Times, May 4, 2004, at C10.

n145. John Schwartz, Rampant Epidemics of Powerful Malicious Software, N.Y. Times, Dec. 1, 2003, at C19; John Schwartz, Worm Hits Microsoft, Which Ignored Own Advice, N.Y. Times, Jan. 28, 2003, at C4.

n146. Brendan I. Koerner, In Computer Security, a Bigger Reason To Squirm, N.Y. Times, Sept. 7, 2003, 3, at 4; Sobig Is Biggest Virus of All, BBC News, Aug. 21, 2003, http://news.bbc.co.uk/ 2/hi/technology/3169573.stm.

n147. Amy Harmon, As Digital Vandals Disrupt the Internet, A Call for Oversight, N.Y. Times, Sept. 1, 2003, at A1; Koerner, supra note 146.

n148. See generally Wikipedia, Malware, http://en.wikipedia.org/wiki/Malware (last visited Apr. 9, 2006) (defining malware as "software designed to infiltrate or damage a computer system").

n149. See TiVo, What Is TiVo?, http://www.tivo.com/1.0.asp (last visited Apr. 9, 2006).

n150. See Wikipedia, TiVo, http://en.wikipedia.org/wiki/TiVo (last visited Apr. 9, 2006).

n151. See, e.g., Eric A. Taub, How Do I Love Thee, TiVo?, N.Y. Times, Mar. 18, 2004, at G1.

n152. See Peter Wayner, Whose Intellectual Property Is It Anyway? The Open Source War, N.Y. Times, Aug. 24, 2000, at G8; see also TiVo, TiVo - GNU/Linux Source Code, http:// www.tivo.com/linux/linux.asp (last visited Apr. 9, 2006) (making TiVo's source code publicly available).

n153. See Katie Hafner, Now Preening on the Coffee Table, N.Y. Times, Feb. 19, 2004, at G1.

n154. See, e.g., Paul Festa, TiVo Hacks Flourish, CNET News.com, Nov. 11, 2004, http:// news.com.com/TiVo+hacks+flourish/2100-1041 3-5447461.html.

n155. TiVoToGo allows users to transfer recordings from their TiVos very slowly in a copy-protected format to their own PCs and to copy them to PC DVDs. See David Pogue, TiVo Adds Portability to the Mix, N.Y. Times, Jan. 6, 2005, at G1.

n156. Philips, Sony, and others make TiVo-compliant boxes under licenses from TiVo. See Laurie J. Flynn, Networks See Threat in New Video Recorder, N.Y. Times, Nov. 5, 2001, at C4.

n157. TiVo can, however, share its programming with other TiVos in a household. See Katie Dean, TiVo Breaks into Home Networks, Wired News, June 10, 2004, http://www.wired.com/ news/digiwood/0,1412,63776,00.html.

n158. See, e.g., TiVo Inc., Definitive Proxy Statement (Form DEF 14A), at 7, 13-14 (May 31, 2005) (noting that NBC is entitled to nominate one of TiVo's directors and indicating that subsidiaries of NBC own more than four percent of TiVo's stock).

n159. See Digital Broad. Content Prot., Report and Order and Further Notice of Proposed Rulemaking, 30 Commc'ns Reg. (P & F) FCC 03-273, at 1189, 1194-97 (Nov. 4, 2003); see also Press Release, FCC, FCC Adopts Anti-Piracy Protection for Digital TV (Nov. 4, 2003), available at http://hraunfoss.fcc.gov/edocs public/attachmatch/DOC-240759A1.pdf.

n160. However, TiVo users have discovered a clandestine thirty-second skip feature embedded in their TiVos. See O'Reilly, The 30-Second Skip, http://hacks.oreilly.com/pub/h/491 (last visited Apr. 9, 2006).

n161. See Paramount Pictures Corp. v. Replay TV, 298 F. Supp. 2d 921, 923 (C.D. Cal. 2004).

n162. See id.

n163. See Dan Tynan, Winners and Losers 2005, PC World, Dec. 27, 2005, http://www.pcworld. com/news/article/0,aid,123923,00.asp (describing TiVo's compatibility with a DRM system provided by Macrovision).

n164. See TiVo Home Media Engine Software Development Kit, http://tivohme.sourceforge.net (last visited Apr. 9, 2006).

n165. See Simon Romero, Wireless Wanderer: A Field Study, N.Y. Times, May 24, 2001, at G1.

n166. See Press Release, palmOne, Inc., Treo 600 from palmOne Now Available for Cingular Wireless GSM/GPRS Customers (Nov. 13, 2003), available at http://www.palm.com/us/company/ pr/2003/111303.html; Update: palmOne Finally Takes Wraps Off Treo 650, SmartPhoneToday, Oct. 25, 2004, http://www.smartphonetoday.com/articles/2004/10/2004-10-25-palmOne-Finally-Takes.html.

n167. See, e.g., Thomas J. Fitzgerald, Music for Your Cellphone, N.Y. Times, July 7, 2005, at C9; Windows Mobile, http://www.microsoft.com/windowsmobile/devices/default.mspx (last visited Apr. 9, 2006).

n168. See Ina Fried, Digging Profits Out of Xbox, CNET News.com, Aug. 10, 2005, http:// news.com.com/Digging+profits+out+of+Xbox/2100-1043 3-5827110.html (describing the Xbox licensing scheme); Wikipedia, Xbox 360, http://en.wikipedia.org/wiki/Xbox 360#Dashboard (last visited Apr. 9, 2006) (describing non-gaming applications). This licensing structure may exist in part because the Xbox hardware is sold at a loss. See Fried, supra.

n169. Microsoft recognizes a variety of certificate authorities such as VeriSign, which in turn can accept signatures from authors of code. See, e.g., Press Release, Microsoft Corp., Microsoft and VeriSign Announce .NET Alliance (July 10, 2001), http://www.microsoft.com/presspass/press/ 2001/Jul01/07-10VeriSignPR.mspx. See generally PageBox, Trusted Sites, http://www.pagebox. net/java/java-trusted.html (last visited Apr. 9, 2006) (explaining the technology behind trusted website support).

n170. MSDN, Introduction to Code Signing, http://msdn.microsoft.com/library/default.asp?url=/ workshop/security/authcode/intro authenticode.asp (last visited Apr. 9, 2006) (describing the minimal credentials that a software publisher must present to be eligible for certification).

n171. See Microsoft, Digital Signature Benefits for Windows Users, http://www.microsoft.com/ winlogo/benefits/signature-benefits.mspx (last visited Apr. 9, 2006).

n172. See id. (describing Microsoft's "Certified for Windows" program).

n173. See supra note 168 and accompanying text.

n174. See David Bank, Microsoft Moves To Rule On-Line Sales, Wall St. J., June 5, 1997, at B1.

n175. See, e.g., Microsoft, Use Microsoft Update To Help Keep Your Computer Current, http:// www.microsoft.com/athome/security/update/msupdate keep current.mspx (last visited Apr. 9, 2006).

n176. In Windows XP Service Pack 2, this feature is automatically turned on. Should the user disable the feature, frequent warnings indicate that the computer is not fully protected due to its inability to receive updates. See Scott Spanbauer, Internet Tips: Tweak Windows XP SP2 Security to Your Advantage, PC World, Oct. 2004, at 166, 167-68, available at http://www.pcworld. com/howto/article/0,aid,117422,00.asp.

n177. This phenomenon is in use by PC applications as well as OSs. See, e.g., Ban Hits Half-Life 2 Pirates Hard, BBC News, Nov. 25, 2004, http://news.bbc.co.uk/2/hi/technology/4041289.stm; Peter Cohen, iTunes Update Disables Internet Playlist Sharing, Macworld, May 27, 2003, http://www.macworld.com/news/2003/05/27/itunes/.

n178. See Mark Ward, Toasting the Crackers, BBC News, Jan. 26, 2001, http://news.bbc.co.uk/1/ hi/sci/tech/1138550.stm; May Wong, DirecTV Fires Back at Hackers, Miami Herald, Jan. 27, 2001, at 1C.

n179. See, e.g., Ward, supra note 178.

n180. See John Battelle, Op-Ed., Building a Better Boom, N.Y. Times, Nov. 18, 2005, at A29; Tim O'Reilly, What Is Web 2.0, O'Reilly Network, Sept. 30, 2005, http://www.oreillynet.com/ pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html.

n181. For a close examination of the range and benefits of such applications as SETI@home, see Yochai Benkler, Sharing Nicely: On Shareable Goods and the Emergence of Sharing as a Modality of Economic Production, 114 Yale L.J. 273, 291-95 (2004).

n182. For a discussion of concerns that hosted services can create new bottlenecks in data ownership, see the conversation about network effects and service levels in Read/WriteWeb, http:// www.readwriteweb.com (Nov. 15, 2004).

n183. See Damon Darlin, A Journey to a Thousand Maps Begins with an Open Code, N.Y. Times, Oct. 20, 2005, at C9 (describing Google Maps Mania, which catalogues information-rich maps built using Google Maps).

n184. See Google, Google Maps API Terms of Use, http://www.google.com/apis/maps/terms.html (last visited Apr. 9, 2006) (stating that Google reserves the right to modify or discontinue Google Maps at any time and for any reason).

n185. Cohen, supra note 177.

n186. Id.

n187. 464 U.S. 417 (1984). In Sony, the Supreme Court declined to impose contributory liability on VCR manufacturers for enabling VCR users to infringe copyrights. See id. at 419-21.

n188. See A & M Records, Inc. v. Napster, Inc., 114 F. Supp. 2d 896, 927 (N.D. Cal. 2000), aff'd in part and rev'd in part, 239 F.3d 1004 (9th Cir. 2001); see also Napster Filter Welcomed by Music Industry, CNN.com, Mar. 2, 2001, http://archives.cnn.com/2001/LAW/03/02/napster.hearing.04 (analyzing the aftermath of the Napster litigation).

n189. See Randal C. Picker, Rewinding Sony: The Evolving Product, Phoning Home and the Duty of Ongoing Design, 55 Case W. Res. L. Rev. 749, 766-68 (2005) (arguing that the Sony test should be replaced by one that turns on whether the producer remains able to distribute updates to consumers).

n190. Id. at 752.

n191. Id. at 767.

n192. See Cyrus Farivar, New Food for IPods: Audio by Subscription, N.Y. Times, Oct. 28, 2004, at G5; Wikipedia, Podcasting, http://en.wikipedia.org/wiki/Podcasting (last visited Apr. 9, 2006).

n193. See generally Zittrain, supra note 24, at 272-73 (describing the open development model for software development).

n194. See Universal City Studios, Inc. v. Reimerdes, 111 F. Supp. 2d 294, 316-19 (S.D.N.Y. 2000); see also Elec. Frontier Found., supra note 88, at 6 (discussing Reimerdes and other DMCA litigation).

n195. For a discussion of the further prospect that law enforcement authorities could issue orders to software makers to use automatic updating to retrieve information from users' PCs, see Zittrain, supra note 77, at 89.

n196. See Benkler, supra note 181, at 356-58 (describing the value and power of non-market-mediated relationships to produce socially and economically useful results).

n197. S. 2048, 107th Cong. (2002). The CBDTPA was similar in substance to the Security Systems Standards and Certification Act, which was drafted in mid-2001 but never introduced in Congress. See Security Systems Standards and Certification Act, Aug. 6, 2001, http://cryptome. org/sssca.htm.

n198. See S. 2048 3.

n199. Id. 9(3).

n200. See id. 3(c).

n201. Maria Trombly et al., China's Bet on Linux, CIO, Oct. 15, 2005, at 21, 21 available at http:// www.cio.com/archive/101505/tl opensource.html.

n202. See, e.g., Lawrence Lessig, The Limits in Open Code: Regulatory Standards and the Future of the Net, 14 Berkeley Tech. L.J. 759, 768-69 (1999) (suggesting that free software can provide a check on government power because it is less regulable than software generated and maintained by a single firm).

n203. Skype was founded by the makers of the Kazaa filesharing program in August 2003 with $ 250,000 in seed funding. Its goal was to produce software for PC-to-PC voice communication. See Constance Loizos, Draper Cashes in on Billion-Dollar Skype, Private Equity Week, Sept. 19, 2005, http://www.privateequityweek.com/pew/freearticles/1122124881310.html. Skype received an additional $ 1 to $ 2 million in November 2003 and $ 18.8 million more in March 2004. Skype was purchased by eBay in September 2005 for at least $ 2.5 billion. See id.; Press Release, Skype, eBay Completes Acquisition of Skype (Oct. 14, 2005), http://www.skype.com/ company/news/2005/skype ebaycompletesacquisition.html. In partnership with Netgear, Skype plans to release a stand-alone wi-fi-based mobile phone in mid-2006. See Press Release, NETGEAR, Inc., NETGEAR, Skype To Connect on Family of Innovative Products Including World's First Skype WiFi Mobile Phone (Jan. 4, 2006), http://www.netgear.com/pressroom/press releasesdetail.php?id=305.

n204. Skype represents a good example of code that can evade censorship - its adoption by end users is notoriously difficult for ISPs to stop - and code that could be the vector for the "biggest botnet ever," given the fact that its code is largely encrypted, its adoption widespread, and its communication with other Skype programs continuous and opaque. See Philippe Biondi & Fabrice Desclaux, Silver Needle in the Skype 3-4, 112-13 (2006), available at http:// www.blackhat.com/presentations/bh-europe-06/bh-eu-06-biondi/bh-eu-06-biondi-up.pdf.

n205. See Yochai Benkler, Coase's Penguin, or, Linux and The Nature of the Firm, 112 Yale L.J. 369, 371 (2002) ("A new model of production has taken root, one that should not be there, at least according to our most widely held beliefs about economic behavior. [We] resist the idea that thousands of volunteers could collaborate on a complex economic project. It certainly should not be that these volunteers will beat the largest and best-financed business enterprises in the world at their own game. And yet, this is precisely what is happening in the software industry."); Julie E. Cohen, The Place of the User in Copyright Law, 74 Fordham L. Rev. 347, 348-49, 370-73 (2005) (rejecting the simplified notion of the "romantic user" who frequently contributes to debates and transforms others' work and asserting the possibility of generative contribution by the "situated user" engaged in "consumption, communication, self-development, and creative play"); Dan Hunter & F. Gregory Lastowka, Amateur-to-Amateur, 46 Wm. & Mary L. Rev. 951 (2004) (describing how the Internet facilitates amateur contribution).

n206. See Saltzer et al., supra note 37.

n207. See Marjory S. Blumenthal, End-to-End and Subsequent Paradigms, 2002 Law Rev. Mich. St. U.-Detroit C.L. 709, 717 (remarking that end-to-end arguments "interact[] with economics, public policy, and advocacy dynamically to shape access to communication and information and to influence innovation").

n208. For articles noting the centrality of end-to-end to the debate, see id., which describes end-to-end as the current paradigm for understanding the Internet, and Lawrence Lessig, The Architecture of Innovation, 51 Duke L.J. 1783 (2002), which argues that end-to-end establishes the Internet as a commons. For debate about the perceived values at stake in end-to-end arguments, see Yochai Benkler, e2e Map (Stanford Program in Law, Sci. & Tech., Conference Summary, The Policy Implications of End-to-End, 2000), http://cyberlaw.stanford.edu/e2e/e2e map.html. For arguments for the preservation of end-to-end neutrality in network implementation, see Written Ex Parte of Professor Mark A. Lemley and Professor Lawrence Lessig, In re Application for Consent to the Transfer of Control of Licenses MediaOne Group, Inc. to AT&T Corp., No. 99-251 (F.C.C. 1999), available at http://cyber.law.harvard.edu/works/lessig/cable/fcc/fcc.html; Lemley & Lessig, supra note 8; David D. Clark & Marjory S. Blumenthal, Rethinking the Design of the Internet: The End to End Arguments vs. the Brave New World (Stanford Program in Law, Sci. & Tech., Conference Paper, The Policy Implications of End-to-End, 2000), available at http:// cyberlaw.stanford.edu/e2e/papers/TPRC-Clark-Blumenthal.pdf, which describes the benefits of end-to-end and how those benefits are in tension with security concerns; Paul A. David, The Beginnings and Prospective Ending of "End-to-End": An Evolutionary Perspective on the Internet's Architecture 26 (Stanford Econ. Dept., Working Paper No. 01-012, 2001), available at http://www-econ.stanford.edu/faculty/workp/swp01012.pdf, which argues that end-to-end openness is a public good, the potential loss to society of which must be calculated when more extensive security solutions are considered; and David P. Reed et al., Active Networking and End-to-End Arguments (Stanford Program in Law, Science & Tech., Conference Paper, The Policy Implications of End-to-End, 2000), http://cyberlaw.stanford.edu/e2e/papers/Saltzer Clark Reed ActiveNetworkinge2e. html, which argues for the preservation of end-to-end and using end-to-end openness as an organizing principle against which to measure programmability and active networking.

n209. See supra pp. 1988-89.

n210. See Saul Hansell, Spam Fighters Turn to Identifying Legitimate E-Mail, N.Y. Times, Oct. 6, 2003, at C1 (discussing authentication and other possible solutions for limiting spam); Yakov Shafranovich, 2004: The Year That Promised Email Authentication, CircleID, Dec. 27, 2004, http://www.circleid.com/posts/2004 the year that promised email authentication (discussing various e-mail authentication proposals to limit spam on the receiving end); see also Saul Hansell, 4 Rivals Near Agreement on Ways To Fight Spam, N.Y. Times, June 23, 2004, at C1 (discussing approaches toward authentication proposed by major ISPs).

n211. See, e.g., Dudney, supra note 134 (providing a case study of traffic filtering by ISPs).

n212. See World Summit on the Info. Soc'y, Overview, http://www.itu.int/wsis/basic/about.html (last visited Apr. 9, 2006). For commentary on the World Summit, see, for example, John Markoff, Control the Internet? A Futile Pursuit, Some Say, N.Y. Times, Nov. 14, 2005, at C4; and Victoria Shannon, Other Nations Hope To Loosen U.S. Grip on Internet, N.Y. Times, Nov. 15, 2005, at C14.

n213. See World Summit on the Info. Soc'y, Why a Summit on the Information Society, http:// www.itu.int/wsis/basic/why.html (last visited Apr. 9, 2006); see also Jennifer L. Schenker, U.N. Meeting Debates Software for Poor Nations, N.Y. Times, Dec. 11, 2003, at C4 (reporting that the World Summit representatives portrayed "open-source, or free-to-share, software [as] crucial for the developing world because it would permit poorer countries to develop their own technology instead of having to import it").

n214. See, e.g., Anti-Spyware Coal., http://www.antispywarecoalition.org (last visited Apr. 9, 2006) (association of information technology companies that focuses on combating spyware); StopBadware.org, http://www.stopbadware.org (last visited Apr. 9, 2006) (nonprofit academic project aimed at fighting malicious software); see also The WildList Org. Int'l Home Page, http://www. wildlist.org (last visited Apr. 9, 2006) (grassroots organization devoted to disseminating information about viruses).

n215. See L. Jean Camp & Allan Friedman, Good Neighbors Can Make Good Fences: A Peer-to-Peer User Security System (Univ. of Mich. Sch. of Info., Conference Paper, Telecommunications Policy and Research Conference, Sept. 24, 2005), available at http://web.si.umich.edu/tprc/papers/ 2005/453/tprc GoodNeighbors.pdf; Alla Genkina & L. Jean Camp, Re-Embedding Existing Social Networks into Online Experiences To Aid in Trust Assessment (Apr. 1, 2005), available at http:// ssrn.com/id=707139; L. Jean Camp et al., Net Trust, http://www.ljean.com/netTrust.html (last visited Apr. 9, 2006).

n216. Indeed, such a scenario need not be hypothetical. See, e.g., Ken Silverstein, The Radioactive Boy Scout, Harper's Mag., Nov. 1998, at 59 (recounting the story of a child who created a nuclear reactor in his backyard shed).

n217. Some scholars, however, rebuke the notion that nonphysical harm is always less injurious than physical harm. See, e.g., Eugene Volokh, Crime-Facilitating Speech, 57 Stan. L. Rev. 1095, 1217 (2005) (commenting on the significance of speech that facilitates grave nonphysical harm and suggesting, therefore, that it ought to enjoy no First Amendment protection).

n218. See supra p. 2006.

n219. For a preliminary sketch of such a division, see Butler Lampson, Accountability and Freedom (2005), available at http://www.ics.uci.edu/cybrtrst/Posters/Lampson.pdf.

n220. A virtual machine is a self-contained operating environment that isolates an application from the entire computer on which it runs, denying the application access to other compartments of the system. See Wikipedia, Virtual Machine, http://en.wikipedia.org/wiki/Virtual machine (last visited Apr. 9, 2006).

n221. See sources cited supra note 210. One example of such an authentication system is Microsoft's Sender ID. See Microsoft, Sender ID, http://www.microsoft.com/mscorp/safety/ technologies/senderid/default.mspx (last visited Apr. 9, 2006).

n222. See generally David R. Johnson et al., The Accountable Internet: Peer Production of Internet Governance, 9 Va. J.L. & Tech. 9 (2004), http://www.vjolt.net/vol9/issue3/v9i3 a09-Palfrey. pdf (discussing the imperfections of filtration).

n223. See Yochai Benkler, Some Economics of Wireless Communications, 16 Harv. J.L. & Tech. 25 (2002) (suggesting that open wireless networks will be more efficient at optimizing wireless communications capacity than spectrum property rights will be).

n224. See Michel Marriott, Hey Neighbor, Stop Piggybacking on My Wireless, N.Y. Times, Mar. 5, 2006, at A1 (explaining some of the dangers of open wireless networks).

n225. See supra p. 2022.

n226. See Wikipedia, Blue Screen of Death, http://en.wikipedia.org/wiki/Blue screen of death (last visited Apr. 9, 2006).