THE INDIAN JOURNAL OF LAW AND TECHNOLOGY
Volume 4, 2008

SAVING THE INTERNET

*Jonathan Zittrain**

TABLE OF CONTENTS

I. INTRODUCTION

The famed Warner Bros. Cartoon antagonist Wile E. Coyote demonstrates a fundamental principle of cartoon physics. He runs off the cliff unaware of its ledge, and continues without falling. The Coyote defies gravity until he looks down and sees that there is nothing under him. His mental gears whirr as he contemplates his predicament. Then: splat!

[*] Professor of Internet Law, Harvard Law School, and Co-Director, Berkman Center for Internet and Society. Reprinted by Permission of the *Harvard Business Review*. © 2007 by the Harvard Business School Publishing Corporation; all rights reserved.

The Internet and the PC are following a similar trajectory. They were designed by people who share the same love of amateur tinkering as the Coyote, and who dealt with problems only as they arose – or left them to individual users to deal with. This "procrastination principle", together with a design premised on contributions from anyone who cared to pitch in, have caused the Internet and PC to emerge from the realms of researchers and hobbyists, and to win out over far more carefully planned and funded networks and information appliances.

The runaway successes of the Internet and PC with the mainstream public have put them in positions of significant stress and danger. Though the Internet's lack of centralized structure makes it difficult to assess the sturdiness of its foundations, there are strong signals that our network and computers are subject to abuse in ways that have become deeper and more prevalent as their popularity has grown.

The core boon and bane of the combined Internet and PC is its *generativity*: its accessibility to people all over the world – people without particular credentials or wealth or connections – who can use and share the technologies' power for various ends, many of which are unanticipated or, if anticipated, would never have been thought to be valuable.

The openness that has catapulted these systems and their evolving uses to prominence has also made them vulnerable. We face a crisis in PC and network security, and it is not merely technical in nature. It is grounded in something far more fundamental : the double-edged ability of members of the public to choose what code they run, which in turn determines what they can see, do and contribute online.

Poor choices about what code to run – and the consequences of running it – could cause Internet users to ask to be saved from themselves. One model to tempt them is found in today's "tethered appliances." These devices, unlike PCs, cannot be readily changed by their owners, or by anyone the owners might know, yet they can be reprogrammed in an instant by their vendors or service providers (think of TiVo, cell phones, iPods, and PDAs). As Steve Jobs said when introducing the Apple iPhone earlier this year, "We define everything that is on the phone. You don’t want your phone to be like a PC, the last thing you want is to have loaded three apps on your phone, and then you go to make a call and it doesn’t work anymore. These are more like iPods then they are like computers.”

If enough internet users begin to prefer PCs and other devices designed along the locked down lines of tethered appliances, that change will tip the balance in a long standing tug of war from a generative system open to dramatic change to a more stable, less interesting system that locks in the *status quo.* Some parties to debates over control of the Internet will embrace this shift. Those who wish to monitor and block network content, often for legitimate and even noble ends, will see novel chances for control that have so far eluded them.

To firms with business models that depend on attracting and communicating easily with customers online, the rise of tethered appliances is a threat. It means that a new gatekeeper is in a position to demand tribute before customers and vendors can connect – a discriminating “2” inside “B2C.”

II. TWO GENERATIVE TRIUMPHS: NETWORK AND PC

Some brief history: The mainstream consumer network environment of the early 1990s looked nothing like today’s Internet, nor did it evolve to become the Internet we have today. As late as 1995, conventional wisdom held that the coalescing global network would be some combination of the proprietary offerings of the time, services like CompuServe, AOL and Prodigy. Yet those companies went extinct or transformed into entirely different businesses. They were crushed by a baling-wire-and-twine network built by government researchers and computer scientists, one that had no CEO and no master business plan.

The leaders of the proprietary networks can be forgiven for not anticipating the Internet’s rise. Not only was there no plan for the provision of content on the Internet, there was an outright hostility towards many forms of it. The Internet’s backbone, operated by the U.S. National Science Foundation, had an acceptable-use policy prohibiting commercial endeavors. For years the

Internet remained a backwater, a series of ad hoc connections among universities and research laboratories whose goal was to experiment with networking. Yet what the developers made was a generative system, open to unanticipated change by large and varied audiences. It is this generativity that has caused its great – and unanticipated – success.

Consumer applications were originally nowhere to be found on the Internet, but that changed in 1991, after the Internet's government patrons began permitting personal and commercial interconnections without network research pretexts, and then ceased any pretense of regulating the network at all. Developers of Internet applications and destinations now had access to a broad, commercially driven audience. Proprietary network service providers who had seen themselves as offering a complete bundle of content and access became mere on-ramps to the Internet, from which their users branched out to quickly thriving Internet destinations for their programs and services. For example, CompuServe's Electronic Mall, an e-commerce service intended to be the exclusive means by which outside vendors could sell products to CompuServe subscribers, disappeared under the avalanche of individual Web sites selling directly to anyone with Internet access.

PCs likewise started off slowly in the business world (even the name "personal computer" evokes a mismatch). Businesses first drew upon custom-programmed mainframes – the sort of complete package IBM offered in the 1960s, for which software was an afterthought – or relied on information appliances like smart typewriters. Some businesses obtained custom-programmed minicomputers, and employees accessed the shared machines through dumb workstations using small, rudimentary local area networks. The minicomputers typically ran a handful of designated applications – payroll, accounts receivable, accounts payable, and company specific programs, such as case management systems for hospitals or course-registration programs for universities. There was not much opportunity for skilled users to develop and share innovative new applications.

Through the 1980s, the PC steadily gained traction. Its ability to support a variety of programs from a variety of makers meant that its utility soon outpaced that of specialized appliances like word processors. Dedicated word processors were built to function the same way over their entire product lifetimes, whereas PC word-processing software could be upgraded or replaced with an application from a competitor without having to replace the PC itself. This IT ecosystem, comprising fixed hardware and flexible software, soon proved its worth.

PCs had some drawbacks for businesses – documents and other important information ended up stored across different PCs, and enterprise wide backup was a real headache. But the price was right, and people entering the workforce soon could be counted on to have skills in word processing and other basic PC tools. As a round of mature applications emerged, there was reason for almost every white collar worker to be assigned a PC, and for an ever broader swath of people to want one at home. These machines might have been bought for one purpose, but their flexible architecture meant that they could quickly be redeployed for many others. A person who bought a PC for word processing might then discover the joys of e-mailing, gaming, or the Web.

Bill Gates used to describe Microsoft's vision as "a computer on every desk". That may have reflected a simple desire to move units – nearly every PC sold meant more money for Microsoft – but as the vision came true in the developed world, the implications went beyond Microsoft's profitability. Whether running Mac or Windows, an installed base of tens of millions of PCs meant that there was tilled soil in which new software could take root. A developer writing an application would not need to convince people that it was worth buying new hardware to run it. He or she would only need to persuade them to buy the software itself. With the advent of PCs connected to the Internet, people would need only click on the right link and new software could be installed. The fulfillment of Gates' vision significantly boosted the generative potential of the Internet and PC, opening the floodgates to innovation.

III. BENEFITS OF GENERATIVITY: INNOVATION AND PARTICIPATION

Generativity is a system's capacity to produce unanticipated change through unfiltered contributions from broad and varied audiences. As such, generativity produces two main benefits: The first is innovative output – new things that improve people's lives. The second is participatory input – the opportunity to connect to other people, to work with them, and to express one's own ability through creative endeavors.

Nongenerative systems can grow and evolve, but such growth is channeled through their makers; Sunbeam releases a new toaster in response to anticipated customer demand, or an old proprietary network like CompuServe adds a new form of instant messaging by programming itself. When users pay for products or services, they can exert pressure on the companies to develop the desired improvements or changes. This is an indirect path to innovation, and there is a growing body of literature about its chief limitation – a persistent bottleneck that prevents large incumbent firms from developing and cultivating certain new uses, despite the benefits they could enjoy with a breakthrough.

For example, Columbia Law School professor Tim Wu has shown that when wireless telephone carriers control what kind of mobile phones their subscribers may use, those phones often have undesirable features that are difficult for third parties to improve. Some carriers have forced telephone providers to limit the mobile phones' Web browsers to certain carrier-approved sites. They have eliminated call timers on the phones, even though these would be trivial to implement and are much desired by users, who would like to monitor whether they have exceeded the allotted minutes for their monthly plan. These limitations persist despite competition among several carriers.

The reason big firms exhibit such innovative inertia, according to a theoretical framework by Clayton Christiensen, is twofold: Big firms have ongoing investments in their existing markets and on established ways of doing business, and disruptive innovations often capture only minor or less profitable markets – at first. By the time the big firms recognize the threat, they are not able to adapt. They lose, but the public wins.

For disruptive innovation to come about, newcomers need to be able to reach people with their offerings. Generative systems make this possible. Indeed, they allow users to try their hand at implementing and distributing new ideas and technologies, filling a crucial gap that is created when innovation is undertaken only in a profit-making model, especially one in which large firms dominate.

Consider novel forms of commercial and social interaction that have bubbled up from unexpected sources in recent years. Online auctions might have been ripe for plucking by Christie's or Sotheby's, but upstart eBay got there first and stayed. Craigslist, initiated as a dot-org by a single person, dominates the market for classified advertising online. Web based e-mail, hosting services for personal web pages, instant messaging software, social networking sites and next-generation search engines emerged from individuals or small groups wanting to solve their own problems or try something neat, rather than from firms realizing there were profits to be gleaned.

Eric von Hippel, head of MIT's innovation and Enterpreneurship Group, has written extensively about how rarely firms welcome improvements to their products by outsiders, including their customers, even when they could stand to benefit from them. In his work, von Hippel makes the case to otherwise rational firms that the users of their products can and often do serve as disruptive innovators, improving products and sometimes adapting them to entirely new purposes. They come up with ideas before there is widespread demand, and vindicate them sufficiently to get others interested. These users are commonly delighted to see their improvements shared. When interest gets big enough, companies can step in and fully commercialize the innovation.

We have thus settled into a landscape in which both amateur and professional small- and large-scale ventures contribute to major innovations. Consumers can become enraptured by a sophisticated "first-person shooter" video game designed by a large firm in one moment, and by a simple animation featuring a dancing hamster in the next. So it is unsurprising that the Internet and PC today comprise a fascinating juxtaposition of sweepingly ambitious software designed and built like a modern aircraft carrier by a large contractor, alongside killer applets that can fit on a single floppy diskette. OS/2, an operating system created as a joint venture between IBM and Microsoft, absorbed more than $2 billion of research and development investment before the plug was pulled, whereas Mosaic, the first graphical PC Web browser, was written by a pair of students during a university break.

Generative growth can blend well with traditional market models. Big firms can produce software where market structure and demand call for such enterprise; smaller firms fill in niches; and amateurs, working alone and in groups, can design both inspirational applets and more labor intensive software that increases the volume and diversity of the technological ecosystem. Once an eccentric and unlikely invention from outsiders has gained traction, traditional means of raising and spending capital to improve technology can shore it up and ensure its exposure to as wide an audience as possible. An information technology ecosystem comprising only the products of a free software movement would be much less usable by the public at large than one in which big firms help sand off rough edges. GNU/Linux has become user friendly thanks to the firms that package and sell copies, even if they cannot claim proprietary ownership of the software itself. Tedious tasks that improve the ease of mastery for the uninitiated are probably best done through corporate models: creating smooth installation engines, extensive help guides, and other forms of hand-holding to help users embrace what otherwise might be an off-putting technical software program or Web service.

For the individual, there is a unique joy to be had in building something-even if one is not the best craftsperson (this is a value best appreciated by experiencing; those who demand proof may not be easy to convince). The joy of being helpful to others – to answer a question simply because it is asked and one knows a useful answer, to be part of a team driving towards a worthwhile goal – is among the best aspects of being human. Our information technology architecture has stumbled into a zone where helpfulness and teamwork can be elicited among and affirmed for tens of millions of people. Novel invention by engineers at the technical layer allows artists to contribute at the content layer. The feeling is captured fleetingly when strangers are thrown together in adverse situations and unite to overcome them – an elevator breaks down, a blizzard or blackout temporarily paralyses the normal cadences of life – but that leads to wonder and camaraderie rather than fear. The internet of the early twenty-first century has distilled some of these values, promoting them without the kind of adversity or physical danger that would make a blizzard fun for the first day but divisive and lawless after the first week without structured relief.

IV. THE GENERATIVE STALL

Generative technologies need not produce forward progress, if by progress one means something like enhancing social welfare. Rather, they foment change. Generative systems are by their nature unfinished, awaiting further elaboration from users and firms alike. As such they can be threatened as soon as their popularity causes abusive business models to pop up. The very openness and user-adaptability that make the Internet a creative wellspring also allow for the propagation of assorted evils – viruses, spam, porn, predation, fraud, vandalism, privacy violations and potentially ruinous attacks on Web sites and on the integrity of the Internet itself. This is becoming an existential threat to the generative IT ecosystem.

The benefit of the generative PC is that it may be repurposed by the neophyte user at the click of a mouse. That is also a huge problem, for two main reasons. First, the PC user who clicks on bad code in effect hands over control of the PC to a total stranger. Second, the threat presented by bad code has been steadily increasing. The most well known viruses have so far had completely innocuous payloads. The 2004 Mydoom worm spread like wildfire and affected connectivity in millions of computers around the world. Though it costs billions of dollars in lost productivity, Mydoom did not tamper with data, and it was programmed to stop spreading at a set time. Viruses like Mydoom are more like the crime of graffiti, with no economic incentive, than like the sale of illegal drugs, with its large markets and sophisticated crime syndicates.

There is now a business model for bad code – one that gives many viruses and worms payloads for purposes other than simple reproduction. What seemed truly remarkable when it was first discovered is now commonplace: viruses that comprise PCs to create large “botnets” to open to later instructions. Such instructions have included directing the PC to become the botnet’s own e-mail server, sending spam by the millions to email addresses harvested from the hard disk of the machine itself or from Web searches, all in a process typically unnoticeable to the PC’s owner. One estimate pegs the number of PC’s involved in such botnets at 100 million to 150 million – one quarter of all the computers on the Internet as of early 2007. Such zombie computers were responsible for more than 80% of the world’s spam in June 2006, and spam in turn accounted for an estimated 80% of the world’s e-mail that month.

Because the current computing and networking environment is so sprawling and dynamic, and its ever-more-powerful building blocks are owned by and managed by regular citizens rather than technical experts, its vulnerability has increased substantially. The public will not and cannot maintain their PCs to the level that professional network administrators do, despite the fact that their machines are significantly more powerful than the minicomputers of the 1970s and 1980s. That vulnerability is exacerbated by people's increasing dependence on the Internet. Well-crafted worms and viruses routinely infest vast swaths of Internet-connected personal computers. In 2004, for example, the Sasser worm infected more than half a million computers in three days. The Sapphire/Slammer worm in January 2003 went after a particular kind of Microsoft server and infected 90% of them – 120,000 machines – within 10 minutes. These hijacked machines together were performing 55 million searches per second for new targets just three minutes after the first computer fell victim. If any of these pieces of malware had truly "mal" or nefarious purposes – for example, to erase hard drives or randomly transpose numbers in spreadsheets – nothing would stand in the way.

The fundamental tension is that the point of a PC is to be easy for users to reconfigure to run new software, but when users make poor decisions about what new software to run, the results can be devastating to their machines and, if they are connected to the Internet, to countless others. Simply choosing a more secure platform does not solve the problem. To be sure, Microsoft Windows has been the target of malware infections for years, but this in part represents Microsoft's dominant market share. As more users switch to other platforms, those platforms will become appealing targets as well. And the most enduring way to subvert security measures may be through the front door – by simply asking the user's permission to add some malware designed as new functionality – rather than trying to steal in through the back and silently exploit an operating system flaw.

PC and Internet security vulnerabilities are a legitimate menace, and people are right to be concerned. However, the most likely reactions if they are not forestalled will be as unfortunate as the security problems themselves. Users will choose PCs that operate more like appliances, forfeiting the ability to install new code themselves. Instead they will use their machines as mere dumb terminals linked to Web sites that offer added interactivity. Many of these Web sites are themselves amenable to appliance-like behavior. Indeed, what some people have applauded as Web 2.0 – a new form of peer-to-peer networks and collective, collaborative content production – is an architecture that can be tightly controlled and maintained by a central source, which may choose to operate in a generative way but is able to curtail those abilities at any time.

Consider Google's terrific map service. It is not only highly useful to end users, it also has an open application programming interface to its map data. Thanks to the open API, a third party website creator can start with a mere list of street addresses and immediately produce on her site a Google map with a digital push-pin at each address. This allows any number of "mashups" to be made, combining Google maps with third party geographic data sets. Web developers are using the Google Maps API to create websites that find and map the nearest Starbucks; create and measure running, hiking or biking routes; pinpoint the locations of traffic light cameras; and collate prospective partners on the Internet dating sites to produce instant displays to show where one's best matches are located.

In allowing coders to access its map data, Google's mapping service is generative. But its generativity is contingent: Google assigns each web developer a key and reserves the right to revoke it at any time for any reason – or to terminate the whole service. It certainly is understandable that Google, in choosing to make a generative service out of something in which it has invested heavily, would want to control it. But this puts within the control of Google and anyone who can regulate Google, all downstream uses of Google maps (and maps in general) to the extent that Google Map's excellence means that other mapping services will fail or never be built.

The business model of other next generation Internet appliances and services are neither enduringly generative nor, in some instances, as unambiguously generative as the open Internet and PC. For example, Microsoft's Xbox is a video game console that has as much computing power as a PC and is networked to other users with Xboxes. Microsoft loses money on every Xbox it sells but makes it back by selling its own games and other software to run on it. Third-party developers can write Xbox games, but they must obtain a license from Microsoft (which includes giving Microsoft a share of the profits) before they can distribute them.

Most mobile phones are similarly constrained: They are smart and many can access the Internet, but the access is channeled through browsers provided and controlled by the phone service vendor. Many PDAs come with software provided through special arrangements between the device and software vendors, as Sony's Mylo does when it offers Skype. Without first inking deals with device makers, software developers cannot have their code run on the devices even if the users desire it. In 2006, AMD introduced the Internet Box, a device that looks just like a PC but cannot run any new software without AMD's permission. What's more, AMD can install on the machines any software it chooses – even after they have been purchased.

The growing profusion of tethered applications takes many Internet innovations and wraps them up neatly and compellingly, which is good – but only if the Internet and PC can remain sufficiently in the center of the digital ecosystem to produce the next round of innovations and to provide competition for the locked-down appliances. The balance between the two spheres is precarious, and it is slipping toward the appliances. People buy these devices for their convenience or functionality, and some may appreciate the fact that they limit the damage users can do through ignorance or carelessness. But appliances also circumscribe the beneficial applications users can create or receive from others – applications they may not realize are important to them when they purchase the device. The risk, then, is that users unwittingly trade away the future benefits of generativity, a loss that may go unappreciated even as innovation tapers off.

Eliminate the PC from many dens and living rooms, and we eliminate the test bed and distribution point for new software. We also eliminate the safety valve that keeps information appliances honest. If TiVo makes a digital video-recorder that too strictly limits what people can do with their recorded video, customers will turn to DVR software like MythTV, which records and plays TV shows on PCs; if mobile phones are too expensive, people will use Skype.

Of course, people don't buy PCs as insurance policies against appliances that limit their freedom (even though they serve this vital function); they buy them to perform certain preconceived tasks. But if Internet security breaches and other sorts of anarchy threaten the PC's ability to perform those tasks reliably, most consumers will not see the PC's merit, and the safety valve will be lost. If the PC ceases to be at the center of the information technology ecosystem, the most restrictive aspects of information appliances will become commonplace.

V. INFORMATION APPLIANCES AND REGULATION

When information appliances stay connected to their makers, those companies can be asked to implement changes to the way they work long after they have been purchased for a specific use. Consider the case of *TiVo v. Echostar.* TiVo introduced the first digital recorder in 1998, allowing customers to record and time-shift TV shows. In 2004, TiVo sued satellite TV distributor Echostar for infringing TiVo's patents by building DVR functionality into some of Echostar's dish systems, TiVo won and was awarded $90 million in damages and interest – but that was not all. In August 2006 the court issued an order directing Echostar to disable the DVR functionality in most of the infringing units then in operation.

In other words, the court ordered Echostar to kill DVRs in the living rooms of people around the world who had bought them and might be watching programs recorded on them at that very instant. Imagine sitting down to watch a much anticipated TV show or sportscast and instead finding that all your recordings have been zapped along with the DVR functionality itself – killed by a remote signal traceable to the stroke of a judge's quill. The logic is plain: If an article infringes intellectual property rights, under certain circumstances it can be impounded and destroyed. It is typically impractical to go round impounding every item that falls under this category (police officers don't go door-to-door looking for Rolex and Louis Vuitton knockoffs), so plaintiffs and prosecutors traditionally go after only those selling contraband goods. But the tethered functionality of a DVR means that Echostar can easily effect the remote modification or even destruction of its units.

Remote modification can also allow makers to repurpose their appliances, sometimes in ways that are undesirable to their owners. General Motors and BMW offer onboard systems like OnStar to provide car owners with a variety of useful services and functions, including hands free calling, turn-by-turn driving directions, tire pressure monitoring, and emergency roadside assistance. Because the systems are networked and remotely upgradeable, the U.S. Federal Bureau of Investigation sought to use the technology to eavesdrop on conversations occurring in a vehicle by remotely reprogramming the onboard system to function as a roving bug. The bureau obtained secret orders requiring on carmaker to carry out that modification, and the company complied under protest. A U.S. federal appellate court found in *The Company v. The United States* that the anonymous carmaker could theoretically be ordered to perform the modifications but that the FBI's surveillance interfered with the computer system's normal use. A car with a secret open line to the FBI could not simultaneously connect to the automaker. If occupants tried to use the system to summon emergency help, it would not function (presumably, the FBI would not come to the aid of the motorist the way the automaker promises to do). The implication of the ruling was that secret FBI surveillance of this sort would be legally permissible if the system were redesigned to simultaneously process emergency requests.

A shift to smarter appliances, ones that can be updated by – and only by – the makers, is fundamentally changing the ways in which we experience our technologies. They become contingent: Even if you pay up front for them, such appliances are rented instead on owned, subject to the revision by the maker at any moment.

What price will that control exact? It is difficult to sketch a picture of all the innovative changes that will not happen in a future dominated by appliances, but history offers a guide. Before the generative PC and Internet entered the mainstream around 1995, the IT landscape saw comparatively few innovations. In the dozen years since then, the Internet and PC have combined to inspire accelerated technical innovation outside the traditional firm-based R&D process: new web-powered forms of business value, new social networks and communities of interest, and experiments in collaborative, collective interest. They are crucibles, for new forms of culture, political action and participation, and they will lose their power if the Internet and its end points migrate towards more reliable but less changeable configurations.

VI. SAVING THE GENERATIVE INTERNET

If the Internet *status quo* is untenable, and the solution of tethered appliances creates too many undesirable consequences, we must look for other solutions. The central challenge facing today's information technology ecosystem is to maintain a generative openness to experiments that can be embraced by the mainstream with as few barriers as possible, in the face of potentially overwhelming problems that arise precisely because it is so flexible and powerful. We may draw useful general guidelines from some of the success stories of the generative model that have shown staying power. Here is a brief sampling.

A. Netizenship

One solution to the generative problem deploys tools for people to use, usually in small groups, to prevent what they see as abuse. For example, Wikipedia offers easy-to-master tools that make it possible for self-identified editors to combat vandalism that arises from allowing anyone to edit entries. It is a system at once naïve and powerful compared with the more traditional levers of regulation and control designed to stop outliers from doing bad things. It is the opposite of the client-service model in which a customer calls a help line. Rather, it's like a volunteer fire department or neighborhood watch. Not everyone will be able to fight fires or watch the neighborhood – to be sure, some will be setting the fires – but even a small subset can become a critical mass.

The propogation of bad code is a social problem as well as a technical one, and people can enter into a social configuration to attack it. A small application could run unobtrusively on PCs of participating users and report either to a central source, or perhaps only to each other, information about the vital signs and running code of that PC, which would help other PCs understand whether the code is risky or not. With that information, one PC could use other unidentified PCs' experiences to empower the user. At the moment the user is deciding whether to run some new software, the application's connections to other machines could show, say, how many of the other machines were running the code, whether the machines of self-described experts were running it, whether those experts had been moved to vouch for it, and how long the code had been available. It could also signal the amount of unintended network traffic, pop-up ads, or crashes the code appears to cause. These sorts of data could be viewed on a simple dashboard, letting PC users make quick judgements in light of their own risk preferences.

B. Virtual Machines

For those people who simply want their PCs to operate reliably, a medium-term solution may lie in technologies that allow mission-critical work to be isolated from whimsical, experimental activities that might be dangerous – or might become the next key use of the Internet. Computer scientist Butler Lamson and others are developing promising architectures that allow single PCs to have multiple zones, two or more virtual machines running within one box. A meltdown in the red experimental zone cannot affect the more secure green zone, and thus the consumer is spared having to choose between a generative box and an appliance. Tax returns and other important documents go in green; Skype starts out in red, and then moves over only when it seems prime time.

C. More Help from ISPs

Maintaining the security of a generative system is, by its nature, an ongoing process, one requiring the continuing ingenuity of those who want it to work well, and the broader participation of others to counter the actions of a determined minority to abuse it. If the network is completely open, the end points can come under assault. If the end points remain free as the network becomes slightly more ordered, they act as safety valves should network filtering begin to block more than bad code. Today, ISPs turn a blind eye to zombie computers on their networks, so they do not have to spend time working with their subscribers to fix them. Whether through new industry best practices or through a rearrangement of liability requiring ISPs to take action in the most flagrant and egregious of zombie situations, we can buy another measure of time in the continuing cat-and-mouse game of security

D. Network Neutrality for Mashups

Those who provide content and services over the Internet have lined up in favor of 'network neutrality' by which the ISPs would not be permitted to disfavor certain legitimate content that passes through their servers. Similarly those who offer open APIs on the Internet ought to be application neutral, so all those who want to build on top of their interfaces can rely on certain basic functionality.

Generative systems offer extraordinary benefits. As they go mainstream, the people using them can share some sense of the experimentalist spirit that drives them. The solutions above are sketched on the most basic of terms, but what they share is the idea that for the generative Internet to save itself, it must generate its own solutions. The more we maintain the Internet as a work in progress, the more progress we can make.

APPENDIX I: FOUR ELEMENTS OF GENERATIVITY

Four main features define generativity: (1) how strongly a system or technology leverages a possible set of tasks; (2) its adaptability to a set of tasks; (3) its ease of mastery; and (4) its accessibility. The greater extent to which these features are represented in a system, the more readily it can be changed in unanticipated ways – and the more generative it is. For example, many tools can be leveraging and adaptable but are difficult to master – thus decreasing generativity.

Leverage. Generative systems make difficult jobs easier. The more effort they save, the greater number of instances in which their use can make a difference to someone, the more generative they are. Leverage is not exclusively a feature of generative systems; non-generative specialized technologies (a plowshare, for instance) can provide great leverage for the tasks they have been designed to perform.

Adaptability. Adaptability applies to both the breadth of a system's uses without change and the ease with which it can be modified to broaden its range of uses. Adaptability is a spectrum – a technology that offers hundreds of different kinds of uses is more adaptable, and thus more generative than a technology that offers fewer.

Ease of Mastery. How easy is it for broad audiences to both adopt and adapt a technology? An airplane is neither simple to fly nor simple to modify for new purposes. Paper, on the other hand can be readily mastered and adapted-whether to draw on or to fold into airplanes. The skills needed to use many otherwise generative technologies may be hard to absorb, requiring apprenticeship, formal training or long practice.

Accessibility. The easier it is to obtain the technology, tools and information necessary to achieve mastery – and convey changes to others – the more generative the system is. Barriers to access include the sheer expense of producing (and therefore consuming) the technology; taxes and regulations surrounding its adoption or use; and secrecy or obfuscation that its producers wield in order to maintain scarcity or control.

APPENDIX II: WHAT'S GENERATIVE AND WHAT'S NOT?

LEGOs and a Dollhouse. Legos are highly adaptable, accessible and easy to master. They can be built, deconstructed and rebuilt into whatever form the user wishes, and third parties can publish "recipes" for new forms. The less generative dollhouse supports imaginative play, but is itself unmodifiable.

Hammers and Jackhammers. A hammer is accessible, easy to master, and useful in any number of household tasks. A jackhammer is less broadly accessible, harder to master and good only for breaking up asphalt, concrete and stone.

PC and TiVo. A PC is an adaptable multipurpose tool whose leverage extends through networked access to new software and other users. TiVo is an inflexinle tethered appliance. Though based on the same technology as a PC, it can be modified only by its maker, restricting its uses to those that TiVo invents.

Bicycles and Airplanes. Bicycles are accessible (there's no license to pedal), relatively easy to master, and adaptable by large communities of avid users and accessorizing firms. Airplanes are highly useful for long distance travel, but not very accessible, adaptable or easy to master.