Global Networks and the Effects on Culture

By
ALEXANDER R. GALLOWAY

This analysis aims to derive general principles for understanding the information age through an examination of the global computer networks that facilitate it. Computer networks are created via shared technical standards called protocols. These protocols exhibit several key characteristics, including openness, flexibility, robustness, and voluntary adoption. While computer networks such as the Internet were originally invented to avoid specific social and political threats during the height of the cold war, today networks suffer from a host of new vulnerabilities. Computer viruses provide a case study for understanding these new vulnerabilities and the future political challenges posed by networks of all kinds.

Keywords: network; Internet; protocol; technical standards; computer virus

In recent years, social scientists of all stripes have struggled to come to terms with digital media and the distributed computer networks facilitated by them. It appears that few sectors of culture have survived this dramatic historical shift unscathed. Yet much contemporary research begins from the assumption, incorrect in my mind, that the "information society" is an adequate stand-in for the information machines that power it (Castells 1996; Lessig 2001), and thus it is possible for one to study the former while skipping the latter. At the same time, those who *do* know how information machines work (computer scientists and electrical engineers, mostly) spend little time ruminating on the larger cultural implications of such machines (Stevens 1994; Hall 2000). This article is an attempt to reverse this trend. Indeed, to understand cultural production in the digital age, one must have a firm grasp of both the digital and cultural spheres. Otherwise, one risks conclud-

Alexander R. Galloway received his Ph.D. from Duke University and is currently an assistant professor of media ecology at New York University. His first book, Protocol: How Control Exists after Decentralization, *is published by MIT Press.*

DOI: 10.1177/0002716204270066

ing things about culture that are based in misunderstandings of technological facts. Let me start, then, with an overview of some of the relevant historical transformations that have taken place in the digital age.

The Internet Protocols

The Internet is a global distributed computer network, rooted in the American academic and military culture of the 1950s and 1960s. In the late 1950s, in response to the Soviet Sputnik launch and other fears connected to the cold war, Paul Baran at the Rand Corporation created a computer network that was independent of centralized command and control and thus able to withstand a nuclear attack that targeted centralized hubs. American anxiety over Soviet technological advancement was very real after the Sputnik launches of the late 1950s. "The launching of the sputniks told us," wrote John Dunning for *The New York Times Magazine* in 1957, "that a great despotism is now armed with rockets of enormous thrust, and guidance systems that could deliver a hydrogen warhead of one or more megatons to any spot in the United States" (see also Denning 1990, 19). So in August 1964, Baran published an eleven-volume memorandum for the Rand Corporation outlining his research, documents that "were primarily written on airplanes in the 1960 to 1962 era," he recounts (Baran 1999). Katie Hafner and Matthew Lyon (1996) dispute the purely militaristic origins of the Internet, arguing instead that the Internet derived from the altruistic concerns of a few academics rather than the strategic interests of the Department of Defense. Yet they equivocate, writing on one hand that "the project had embodied the most peaceful intentions—to link computers at scientific laboratories across the country so that researchers might share computer resources. . . . The ARPAnet and its progeny, the Internet, had nothing to do with supporting or surviving war—never did" (p. 10); while on the other hand, they admit that Baran "developed an interest in the survivability of communications systems under nuclear attack" (p. 54).

Baran's network was based on a technology called "packet-switching" that allows messages to break themselves apart into small fragments. The term was coined not by Baran but by British scientist Donald Davies who, unaware of Baran's work, also invented a system for sending small packets of information over a distributed network. Both scientists are credited with the discovery; however, because of Baran's proximity to the newly emerging Advanced Research Projects Agency (ARPA) network, which would be the first to use Baran's ideas, Davies's historical influence has diminished. In packet-switching, each fragment, or packet, is able to find its own way to its destination. Once there, the packets reassemble to create the original message. In 1969, ARPA at the U.S. Department of Defense started the ARPAnet, the first network to use Baran's packet-switching technology. The ARPAnet allowed academics to share resources and transfer files. In its early years, the ARPAnet (later renamed DARPAnet) existed unnoticed by the outside world, with only a few hundred participating computers, or "hosts." All addressing for this network was maintained by a single machine located at the Stanford

Research Institute in Menlo Park, California. By 1984, the network had grown larger. Paul Mockapetris invented a new addressing scheme, this one decentralized, called the Domain Name System (DNS). The computers had changed also. By the late 1970s and early 1980s, personal computers were coming to market and appearing in homes and offices. In 1977, researchers at Berkeley released the highly influential "BSD" (Berkeley Software Distribution) flavor of the UNIX operating system, which was available to other institutions at virtually no cost. With the help of BSD, UNIX would become the most important computer operating system of the 1980s.

In the early 1980s, the suite of protocols known as TCP/IP (Transmission Control Protocol/Internet Protocol) was also developed and included with most UNIX servers. TCP/IP allowed for cheap, ubiquitous connectivity. In 1988, the Defense Department transferred control of the central "backbone" of the Internet over to the National Science Foundation, which in turn transferred control to commercial telecommunications interests in 1995. In that year, there were 24 million Internet users. Today, the Internet is a global distributed network connecting about a billion people around the world.

At the core of networked computing is the concept of *protocol*. A computer protocol is a set of recommendations and rules that outline specific technical standards. The protocols that govern much of the Internet are contained in what are called RFC (Request for Comments) documents. The expression derives from a memorandum titled "Host Software" sent by Steve Crocker on April 7, 1969, which is known today as RFC 1. Called "the primary documentation of the Internet" (Loshin 2000, xiv), these technical memoranda detail the vast majority of standards and protocols in use on the Internet today. The RFCs are published by the Internet Engineering Task Force (IETF). They are freely available and used predominantly by engineers who wish to build hardware or software that meets common specifications. The IETF is affiliated with the Internet Society (ISOC), an altruistic, technocratic organization that wishes "to assure the open development, evolution and use of the Internet for the benefit of all people throughout the world" (ISOC 2004). Other protocols are developed and maintained by other organizations. For example, many of the protocols used on the World Wide Web (a network within the Internet) are governed by the World Wide Web Consortium (W3C). This international consortium was created in October 1994 to develop common protocols such as Hypertext Markup Language (HTML) and Cascading Style Sheets. Scores of other protocols have been created for a variety of other purposes by many different professional societies and organizations.

Protocol is not a new word. Prior to its usage in computing, protocol referred to any type of correct or proper behavior within a specific system of conventions. It is an important concept in the area of social etiquette as well as in the fields of diplomacy and international relations. Etymologically it refers to a flyleaf glued to the beginning of a document, but in familiar usage the word came to mean any introductory paper summarizing the key points of a diplomatic agreement or treaty.

With the advent of digital computing, however, the term has taken on a slightly different meaning. Now, protocols refer specifically to standards governing the

implementation of specific technologies. Like their diplomatic predecessors, computer protocols establish the essential points necessary to enact an agreed-upon standard of action. Like their diplomatic predecessors, computer protocols are vetted out between negotiating parties and then materialized in the real world by large populations of participants (in one case citizens and in the other computer users). Yet instead of governing social or political practices as did their diplomatic predecessors, computer protocols govern how specific *technologies* are agreed to, adopted, implemented, and ultimately used by people around the world. What was once a question of consideration and sense is now a question of logic and physics.

To help understand the concept of computer protocols, consider the analogy of the highway system. Many different combinations of roads are available to a person driving from point *A* to point *B*. However, en route one is compelled to stop at red lights, stay between the white lines, follow a reasonably direct path, and so on. These conventional rules that govern the set of possible behavior patterns within a heterogeneous system are what computer scientists call protocol. Thus, protocol is a technique for achieving voluntary regulation within a contingent environment.

These regulations always operate at the level of coding—they encode packets of information so they may be transported; they code documents so they may be effectively parsed; they code communication so local devices may effectively communicate with foreign devices. Protocols are highly formal; that is, they encapsulate information inside a technically defined wrapper, while remaining relatively indifferent to the content of information contained within. Viewed as a whole, protocol is a distributed management system that allows control to exist within a heterogeneous material milieu.

Unique Characteristics of the Internet Protocols

In this day and age, technical protocols and standards are established by a self-selected oligarchy of scientists consisting largely of electrical engineers and computer specialists. Composed of a patchwork of many professional bodies, working groups, committees, and subcommittees, this technocratic elite toils away, mostly voluntarily, in an effort to hammer out solutions to advancements in technology. Many of these scientists are university professors. Most all of them either work in industry or have some connection to it. Membership in this technocratic ruling class is open. "Anyone with something to contribute could come to the party" (Feinler 1999), wrote one early participant. But, to be sure, because of the technical sophistication needed to participate, this loose consortium of decision makers tends to fall into a relatively homogeneous social class: highly educated, altruistic, liberal-minded science professionals from modernized societies around the globe. And sometimes not so far around the globe. Of the twenty-five or so original protocol pioneers, three of them—Vint Cerf, Jon Postel, and Steve Crocker—came from a single high school in Los Angeles's San Fernando Valley (Cerf 1988). Furthermore, during his long tenure as RFC editor, Postel was the single gatekeeper through whom all protocol RFCs passed before they could be published. Hafner

and Lyon (1996, 145) describe this group as "an ad-hocracy of intensely creative, sleep-deprived, idiosyncratic, well-meaning computer geniuses" (see also Malkin 1992).

There are few outsiders in this community. Here, the specialists run the show. To put it another way, while the Internet is used daily by vast swaths of diverse communities, the standards makers at the heart of this technology are a small entrenched group of techno-elite peers. The reasons for this are largely practical. "Most users are not interested in the details of Internet protocols," Cerf (personal communication 2002) observes. "They just want the system to work." Or as former IETF Chair Fred Baker (personal communication 2002) reminds us, "The average user doesn't write code. . . . If their needs are met, they don't especially care how they were met."

To keep the system working, the protocol designers built into the system several key characteristics. Inspired by Baran's original vision, the Internet protocols are designed to accommodate massive *contingency*. This characteristic is illustrated best by the TCP. TCP makes communication on the Web notably reliable: information is monitored during transport and is re-sent if lost or corrupted. The robust quality of these networks is achieved by following a general principle: "be conservative in what you do, be liberal in what you accept from others" (Postel 1981). This means that TCP hosts should "liberally" accept as much information as possible from other, foreign devices. But if any of the information is corrupted, the host, acting "conservatively," will delete the information and request a fresh copy be resent. As the RFC notes, the goal of TCP is "robustness in the presence of communication unreliability and availability in the presence of congestion" (ibid.). This is known in the standards community as the "robustness principle."

It is worth looking at a single standards body in detail, one that illustrates well the general characteristics I wish to highlight in the standards community at large. ANSI, the American National Standards Institute, is responsible for aggregating and coordinating the standards creation process in the United States. While it does not create any standards itself (Internet protocols or otherwise), it is a conduit for federally accredited organizations in the field that are developing technical standards. The accredited standards developers must follow certain rules designed to keep the process open and equitable for all interested parties. ANSI then verifies that the rules have been followed by the developing organization before the proposed standard is adopted. ANSI is also responsible for articulating a national standards strategy for the United States. This strategy helps ANSI advocate in the international arena on behalf of U.S. interests. ANSI is the only organization that can approve standards as American national standards.

Many of ANSI's rules for maintaining integrity and quality in the standards development process revolve around principles of openness and transparency. For this reason, they are a good case study for understanding the unique characteristics of today's network standards. ANSI writes that

- Decisions are reached through *consensus* among those affected.
- Participation is *open* to all affected interests.

- The process is *transparent*—information on the process and progress is directly available.
- The process is *flexible*, allowing the use of different methodologies to meet the needs of different technology and product sectors. (ANSI 2004)

Besides being consensus-driven, open, transparent, and flexible, ANSI standards are also voluntary, which means that no one is bound by law to adopt them. Voluntary adoption in the marketplace is the ultimate test of a standard. Standards may disappear in the advent of a new superior technology or simply with the passage of time. Voluntary standards have many advantages. By not forcing industry to implement the standard, the burden of success lies in the marketplace. And in fact, proven success in the marketplace generally predates the creation of a standard. The behavior is emergent, not imposed. (An interesting counterexample to this trend of voluntary adoption happened on January 1, 1983, when the ARPAnet instigated a mandatory rollover to the then-new protocol suite TCP/IP. If a host did not roll over, it would eventually have been dropped from the network.) Yet it is important to underscore that while most technical standards today are voluntary, this does not mean that they are haphazardly or infrequently adopted. In fact, the core standards of the Internet (TCP/IP) are some of the most universally adopted technologies in the history of mankind. Statisticians estimate that there are approximately 1 billion Internet users today, and to connect, each one must implement dozens of identical standards.

Case Study: Computer Viruses

But what are the social and cultural effects of universal network standards? The principles of flexibility and robustness have changed everything from economic supply chains (with "just in time" fulfillment) to how one goes about buying a book (with "collaborative filtering" on Web sites like Amazon.com). But to carry these ideas further, it is worth looking at a specific example: computer viruses.

While a few articles on viruses and worms appeared in the 1970s and the beginning of the 1980s, Frederick Cohen's work in the early 1980s is cited as the first sustained examination of computer viruses (see Cohen 1994; Burger 1988, 19). He approached this topic from a scientific viewpoint, measuring infection rates, classifying different types of viruses, and so on.

> The record for the smallest virus is a Unix "sh" command script. In the command interpreter of Unix, you can write a virus that takes only about 8 characters. So, once you are logged into a Unix system, you can type a 8 character command, and before too long, the virus will spread. That's quite small, but it turns out that with 8 characters, the virus can't do anything but reproduce. To get a virus that does interesting damage, you need around 25 or 30 characters. If you want a virus that evolves, replicates, and does damage, you need about 4 or 5 lines. (Cohen 1994, 38)

Cohen first presented his ideas on computer viruses to a seminar in 1983. His paper "Computer Viruses—Theory and Experiments" was published in 1984, and

his Ph.D. dissertation titled "Computer Viruses" (University of Southern California) was published in 1986. Cohen defines a computer virus as "a program that can 'infect' other programs by modifying them to include a, possibly evolved, version of itself" (Cohen 1994, 2). Other experts agree: "a virus is a self-replicating code segment which must be attached to a host executable" (Polk et al. 1995, 4). Variants in the field of malicious code include worms and Trojan horses. A worm, like a virus, is a self-replicating program but one that requires no host to propagate. A Trojan horse is a program that appears to be doing something useful but also executes some piece of undesirable code hidden to the user.

At the core of networked computing is the concept of protocol. *A computer protocol is a set of recommendations and rules that outline specific technical standards.*

In the 1960s in places like Bell Labs, Xerox PARC, and MIT, scientists were known to play a game called Core War (Dewdney 1984a, 22). In this game, two self-replicating programs were released into a system. The programs battled over system resources and eventually one side came out on top. Whoever could write the best program would win. These engineers were not virus writers; nor were they terrorists or criminals. Just the opposite, they prized creativity, technical innovation, and exploration. Core War was a fun way to generate such intellectual activity. The practice existed for several years unnoticed. "In college, before video games, we would amuse ourselves by posing programming exercises," said Ken Thompson, codeveloper of the UNIX operating system, in 1983. "One of the favorites was to write the shortest self-reproducing program" (Thompson 1990, 98). The engineer A. K. Dewdney (1984b, 14) recounts an early story about a self-duplicating program called Creeper that infested the computer system and had to be brought under control by another program designed to neutralize it, Reaper. Dewdney brought to life this battle scenario using his own gaming language called Redcode.

At 5:01:59 p.m. on November 2, 1988, Robert Morris, a twenty-three-year-old graduate student at Cornell University and son of a prominent computer security engineer at the National Computer Security Center (a division of the National Security Agency), released an e-mail worm into the ARPAnet (Rochlis and Eichin 1990, 202). The precise time of day comes from analyzing the computer logs at Cornell University. Others suspect that the attack originated from a remote login at an MIT computer. This self-replicating program entered approximately sixty thou-

sand computers in the course of a few hours, infecting between twenty-five hundred and six thousand of them (Cohen 1994, 49). And while it is notoriously difficult to calculate such figures, some speculations put the damage caused by Morris's worm at more than $10 million, a figure calculated by summing the number of hours of labor required to repair infected machines, the cost of hardware replacement (if any), and the cost of lost productivity due to downed machines.

Computer viruses thrive in environments that have low levels of diversity. Wherever a technology has a monopoly, you will find viruses.

While the media cited Morris's worm as "the largest assault ever on the nation's computers," the program was largely considered a sort of massive blunder, a chain reaction that spiraled out of control through negligence (*The New York Times*, November 4, 1988, A1). Computer viruses thrive in environments that have low levels of diversity. Wherever a technology has a monopoly, you will find viruses. They take advantage of technical standardization to propagate through the network. Consider the following analogy: what if nine out of ten people carried identical genes? In such a world, disease would spread far and wide with ease. For if a genetic vulnerability was discovered in one victim, the next victim and the next would automatically share the same vulnerability. Disease would have little difficulty jumping from host to host, quickly infecting large sections of the population.

Now imagine if 90 percent of computers were identical. This is roughly how the world of computers looks today. Nine out of ten computers today carry identical genetic code, known by a slightly more familiar name: Microsoft Windows. Microsoft's 90 percent market share creates what scientists call a monoculture. In biology, a monoculture exists whenever a single crop or organism takes over an entire ecology. The same thing exists today in the computerized realm. We are living in a computer monoculture. The monoculture is why computer viruses and e-mail worms exist. Computer viruses are able to propagate far and wide in computer networks by leveraging a single vulnerability from computer to computer. In the summer of 2003, the "Blaster" e-mail worm infected more than four hundred thousand computers in a short period of time. By contrast, it took eight months for SARS to infect eight thousand people. Biodiversity is a natural barrier to contagion. And diversity within the human species helped keep SARS from spreading further. Likewise, the computer monoculture fueled epidemics like Blaster.

If diversity existed in the computer sector, viruses would die out overnight. Linux or Macintosh machines are disproportionately saved from the digital plague. The reason lies not in superior security systems (both platforms are vulnerable to digital exploits); it lies in the fact that neither is a monoculture. They occupy the 10 percent fringe. Viruses that originate on Windows machines have a difficult time jumping to other species of computer, leaving Linux or Macintosh users uninfected (though they may still feel the effects of e-mail worms without becoming infected themselves).

As the example of computer viruses illustrates, the various internal characteristics of the Internet can be leveraged in powerful ways by malicious code. Because the Internet is so highly standardized, viruses can propagate quickly by exploiting technical vulnerabilities. Because the Internet is globally interconnected, a single virus will likely have massive repercussions. Because the Internet is so robust, viruses can route around problems and stoppages. And because the Internet is so decentralized, it is virtually impossible to kill viruses once they are released.

Epilogue: The Political Challenges Posed by Networks

I have tried to illustrate how some of the core features of digital networks exert influence over culture and society. Let me return to the original sentiments of networking pioneer Paul Baran. Here, he mentions explicitly how the Internet was invented to avoid certain vulnerabilities of nuclear attack:

> The weakest spot in assuring a second strike capability was in the lack of reliable communications. At the time we didn't know how to build a communication system that could survive even collateral damage by enemy weapons. Rand determined through computer simulations that the AT&T Long Lines telephone system, that carried essentially all the Nation's military communications, would be cut apart by relatively minor physical damage. While essentially all of the links and the nodes of the telephone system would survive, a few critical points of this very highly centralized analog telephone system would be destroyed by collateral damage alone by missiles directed at air bases and collapse like a house of card. (Baran 1999)

In Baran's original vision, the organizational design of the Internet involved a high degree of redundancy, such that destruction of a part of the network would not threaten the viability of the network as a whole. After World War II, strategists called for moving industrial targets outside urban cores in a direct response to fears of nuclear attack. Peter Galison (2001, 20) calls this dispersion the "constant vigilance against the re-creation of new centers." These are the same centers that Baran derided as an "Achilles' heel" and that he longed to purge from the telecommunications network. "City by city, country by country, the bomb helped drive dispersion" (ibid., 25), Galison continues, highlighting the power of the A-bomb to drive the push toward distribution in urban planning. Whereas the destruction of a fleet of Abrams tanks would certainly impinge upon army battlefield maneuvers,

the destruction of a rack of Cisco routers would do little to slow down broader network communications. Internet traffic would simply find a new route, thus circumventing the downed machines. *New Yorker* writer Peter Boyer (2002, 61) reports that DARPA is in fact rethinking this opposition by designing a distributed tank, "a tank whose principal components, such as guns and sensors, are mounted on separate vehicles that would be controlled remotely by a soldier in yet another command vehicle." This is what the military calls Future Combat Systems (FCS), an initiative developed by DARPA for the U.S. Army. It is described as "flexible" and "network-centric." Thus, the Internet can survive attacks not because it is stronger than the opposition, but precisely because it is weaker. The Internet has a different diagram than a nuclear attack does; *it is in a different shape*. And that new shape happens to be immune to the older.

[T]he current global crisis is one between centralized, hierarchical powers and distributed, horizontal networks.

All the words used to describe the World Trade Center after the attacks of September 11, 2001, revealed its design vulnerabilities vis-à-vis terrorists: it was a tower, a center, an icon, a pillar, a hub. Conversely, terrorists are always described with a different vocabulary: they are cellular, networked, modular, and nimble. Groups like al-Qaeda specifically promote a modular, distributed structure based on small, autonomous groups. They write that new recruits "should not know one another" and that training sessions should be limited to "7-10 individuals." They describe their security strategies as "creative" and "flexible" (al-Qaeda 2002, 50, 62).

This is indicative of two conflicting diagrams. The first diagram is based on the strategic massing of power and control, while the second diagram is based on the distribution of power into small, autonomous enclaves. "The architecture of the World Trade Center owed more to the centralized layout of Versailles than the dispersed architecture of the Internet," wrote Jon Ippolito (2001, A27) after the attacks. "New York's resilience derives from the interconnections it fosters among its vibrant and heterogeneous inhabitants. It is in decentralized structures that promote such communal networks, rather than in reinforced steel, that we will find the architecture of survival." The war against terrorism resembles the war in Vietnam or the war against drugs—conflicts between a central power and an elusive network. It does not resemble the Gulf War, or World War II, or other conflicts between states.

"As an environment for military conflict," *The New York Times* reported, "Afghanistan is virtually impervious to American power" (Taubman 2001). (In addition to the stymied U.S. attempt to rout al-Qaeda after September 11, the failed Soviet occupation in the years following the 1978 coup is a perfect example of grossly mismatched organizational designs.) Being "impervious" to American power today is no small feat. Destruction of a network is an all-or-nothing game. One must destroy all nodes, not simply take out a few key hubs. But the opposite is not true. A network needs only to destroy a single hub within a hierarchical power to score a dramatic triumph. Thus, Baran's advice to the American military in 1964 was to become network-like. And once it did, the nuclear threat was no longer a catastrophic threat to communications and mobility (but remains, of course, a catastrophic threat to human life, material resources, and so on). The category shift that defines the difference between state power and guerilla force shows that through a new diagram, guerillas, terrorists, and the like can gain a foothold against their opposition. But as Ippolito (2001) points out, this should be our category shift too, for antiterror survival strategies will arise not from a renewed massing of power on the American side, but precisely from a distributed (or to use his less precise term, decentralized) diagram. Heterogeneity, distribution, and communalism are all features of this new diagrammatic solution.

In short, the current global crisis is one between centralized, hierarchical powers and distributed, horizontal networks. John Arquilla and David Ronfeldt (2001), two researchers at the Rand Corporation who have written extensively on the hierarchy-network conflict, offer a few propositions for thinking about future policy:

- Hierarchies have a difficult time fighting networks.
- It takes networks to fight networks.
- Whoever masters the network form first and best will gain major advantages. (p. 15)

In recent decades, the primary conflict between organizational designs has been between hierarchies and networks, an asymmetrical war. However, in the future the world is likely to experience a general shift downward into a new bilateral organizational conflict—networks fighting networks.

"Bureaucracy lies at the root of our military weakness," wrote advocates of military reform in the mid-1980s. "The bureaucratic model is inherently contradictory to the nature of war, and no military that is a bureaucracy can produce military excellence" (Hart and Lind 1986, 240, 249). The dilemma, then, is that while hierarchy and centralization are almost certainly politically tainted due to their historical association with fascism and other abuses, networks are both bad and good. Drug cartels, terror groups, black hat hacker crews, and other denizens of the underworld all take advantage of networked organizational designs because they offer effective mobility and disguise. But more and more, one witnesses the advent of networked organizational design in corporate management techniques, manufacturing supply chains, advertisement campaigns, and other novelties of the global economy, as well as all the familiar grassroots activist groups who have long used network structures to their advantage. In a sense, networks have been vilified

simply because the terrorists, pirates, and anarchists made them notorious, not because of any negative quality of the organizational diagram itself. In fact, liberatory movements have been capitalizing on network design protocols for decades if not centuries. The goal, then, is not to destroy technology in some neo-Luddite delusion, but to push it into a state of hypertrophy, further than it is meant to go. Then, in its injured, sore, and unguarded condition, technology may be sculpted anew into something better, something in closer agreement with the real wants and desires of its users.

To end, I offer a few instructions for those interested in the effects of global computer networks on cultural production. First is the principle of openness. As the success of the Internet protocols or the Linux operating system illustrates, open technologies have scored major victories over proprietary technologies in recent decades. Value comes less from the protectionist hoarding of social and cultural assets and more from the open deployment of those assets. Leverage the swelling mass of social momentum; lead with a carrot, not a stick. Second, build social institutions that can "route around" problems, just like the core protocols do. Design high degrees of flexibility into social systems. Remove centralized hubs and bottlenecks. Do not just match a problem with a suitable solution; make the problem irrelevant. Third, a warning: following the robustness principle, network technology will tend to standardize rather than diversify. This means that cultural producers will become more and more encumbered by technologies that exploit standardized systems, such as spam, e-mail worms, and computer viruses. So while many of the previous cold war vulnerabilities are gone, a new set of vulnerabilities exist. Finally, those interested in innovative cultural production must understand the *political* import of networks. They are no longer solely the tool of grassroots groups and guerrillas. The "powers that be" have finally come to understand networks too, and what was previously liberating about networks may very well not be liberating in the future.

References

RFC (Request for Comments) documents are archived in several locations online and are accessible via a normal Web search.

al-Qaeda. 2002. *The al-Qaeda documents*. Vol. 1. Alexandria, VA: Tempest.

American National Standards Institute (ANSI). 2004. National standards strategy for the United States. http://www.ansi.org (accessed July 21, 2004).

Arquilla, John, and David Ronfeldt. 2001. *Networks and netwars: The future of terror, crime, and militancy*. Santa Monica, CA: Rand.

Baran, Paul. 1964. *On distributed communications*. Santa Monica, CA: Rand.

———. 1999. Electrical engineer, an oral history conducted in 1999 by David Hochfelder, IEEE History Center, Rutgers University, New Brunswick, NJ. http://www.ieee.org/organizations/history_center/oral_histories/transcripts/baran.html.

Boyer, Peter. 2002. A different war. *The New Yorker*, July 1.

Burger, Ralf. 1988. *Computer viruses*. Grand Rapids, MI: Abacus.

Castells, Manuel. 1996. *The rise of the network society*. Oxford, UK: Blackwell.

Cerf, Vinton. 1988. I remember IANA. RFC 2468.

Cohen, Frederick. 1984. Computer viruses—Theory and experiments. In *Proceedings of the 7th National Computer Security Conference*, 240-63. Gaithersburg, MD: NCSC.

———. 1986. Computer viruses. Ph.D. diss., University of Southern California, Los Angeles.

———. 1994. *A short course on computer viruses*. New York: John Wiley.

Denning, Peter, ed. 1990. *Computers under attack: Intruders, worms, and viruses*. New York: ACM.

Dewdney, A. K. 1984a. Computer recreations. *Scientific American* 251 (September): 22.

———. 1984b. In the game called Core War hostile programs engage in a battle of bits. *Scientific American* 250 (May): 14.

Dunning, John. 1957. If we are to catch up in science. *The New York Times Magazine*, November 10.

Feinler, Jake. 1999. 30 years of RFCs. RFC 2555.

Galison, Peter. 2001. War against the center. *Grey Room* 4:6-33.

Hafner, Katie, and Matthew Lyon. 1996. *Where wizards stay up late: The origins of the Internet*. New York: Touchstone.

Hall, Eric. 2000. *Internet protocols: The definitive guide*. Sebastopol, CA: O'Reilly.

Hart, Gary, and William Lind. 1986. *America can win*. Bethesda, MD: Adler & Adler.

Internet Society (ISOC). 2004. Internet Society mission statement. http://www.isoc.org/isoc/mission/ (accessed July 21, 2004).

Ippolito, Jon. 2001. Don't blame the Internet. *Washington Post*, September 29, A27.

Lessig, Lawrence. 2001. *The future of ideas: The fate of the commons in a connected world*. New York: Random House

Loshin, Pete. 2000. *Big book of FYI RFCs*. San Francisco: Morgan Kaufmann.

Malkin, Gary. 1992. Who's who in the Internet: Biographies of IAB, IESG and IRSG members. RFC 1336. FYI 9.

Polk, W. Timothy, Lawrence E. Bassham, Lisa J. Carnahan, and John P. Wack. 1995. *Anti-virus tools and techniques for computer systems*. Park Ridge, NJ: Noyes Data Corporation.

Postel, Jon, ed. 1981. Transmission Control Protocol. RFC 793.

Rochlis, Jon A., and Mark W. Eichin. 1990. With microscope and tweezers: The worm from MIT's perspective. In *Computers under attack: Intruders, worms, and viruses*, ed. Peter Denning. New York: ACM.

Stevens, W. Richard. 1994. *TCP/IP illustrated*. Vol. 1. New York: Addison Wesley.

Taubman, Philip. 2001. An imbalance of power: Afghanistan's deceptive strength. *The New York Times*, September 20, sec. A, col. 1, p. 30.

Thompson, Ken. 1990. Reflections on trusting trust. In *Computers under attack: Intruders, worms, and viruses*, ed. Peter Denning. New York: ACM.