Free software and the political philosophy of the cyborg world

S. Chopra and S. Dexter
Brooklyn College of the City University of New York
Department of Computer and Information Science
2900 Bedford Avenue
Brooklyn, NY 11210 USA
1+718-951-5657
{schopra, sdexter}@sci.brooklyn.cuny.edu

Abstract

Our freedoms in cyberspace are those granted by code and the protocols it implements. When man and machine interact, co-exist, and intermingle, cyberspace comes to interpenetrate the real world fully. In this cyborg world, software retains its regulatory role, becoming a language of interaction with our extended cyborg selves. The mediation of our extended selves by closed software threatens individual autonomy. We define a notion of freedom for software that does justice to our conception of it as language, sketching the outlines of a social and political philosophy for a cyborg world. In a cyberspace underwritten by free software, political structures become contingent and flexible: the polity can choose to change the extent and character of its participation. The rejection of opaque power is an old anarchist ideal: free software, by making power transparent, carries the potential to place substantive restrictions on the regulatory power of cyborg government.

Keywords: free software, political philosophy, autonomy, cyborg, cyberspace, e-government

Introduction

Software's philosophical implications are not only social and political but also metaphysical. It both creates and destroys distinctions, reworking our ontologies and therefore necessitating a revision of our politics. Code may both advance and counteract political imperatives: in this context, free software is not just a question of managing technology but of determining the contours of our selves and the politics we choose. Technology and politics become inseparable when technologized entities are political actors and objects of political philosophy. A new political philosophy for this technological age must reflect the blurring of boundaries, and the new obscurities, that technology induces. The liberatory potential of free software lies in its potential to address both these effects.

'Cyborg,' a term coined in the early days of the space race, with an eye toward manned exploration of alien terrain, describes a biological entity that 'deliberately incorporates exogenous components extending the self-regulating control function of the organism in order to adapt it to new environments' (Clynes and Kline 1960). Cyborgs wear glasses, use electronic pacemakers, communicate wirelessly, shop online, have credit card applications evaluated by software agents, are policed by digital eyes, and write code to change the behavior of machines that work with, for, and against them. Their world is populated with technology and flesh, demarcated by the inevitably porous boundary of organic and machinic. The dominant form of technological augmentation is the computing device: our cyborg selves are not just any old man-machine hybrid but a new entity that blends the physical and the informational. Human behavior emerges from the interaction of different components of its technologically enhanced cognitive environment. While human agency still exists, it is 'distributed and largely unconscious, or at least a-conscious.' This perspective of the human as intelligent machine allows us to view the human self as just one of many 'easily spliced . . . distributed cognitive systems' in which intelligence inheres equally in the human, in the machinic infrastructure, and in the interface between the two (Borgmann and Hayles 1999). The human use of any sufficiently advanced technology must rely on cyborgian decision-making; thus, human agency and decision-making are blurred by nonhuman agents.

These distributed environments for cognition, while enhancing human cognitive function, raise the possibility that we might not control the parameters of our interactions with them. If cyborg intelligence resides partially in its human component and partially in the machine component, then code becomes a subject of inquiry into the distributed self, and provokes questions about its control. The worry that we surrender decision making to our cognitive extensions is not idle Luddite speculation. It is the real fear of becoming passive recipients of opaque technology. Questions of technology are no longer external to us: to inquire into the nature, shape, form and control of technology is to inquire into our selves. Those to whom we grant this control are those to whom we vouchsafe control of our selves.

In a world in which computer technology infiltrates all interactions with the physical world, when its prosthetic enhancements become ubiquitous, our world is no longer a physical one populated by cyborgs, but is itself a cyborg world. In the cyborg world, humans and machines commingle, a merger enabled and governed by software. This interaction, ranging from mundane uses of computers for personal productivity to networking, from e-government to computer prosthetics, to our saturation by the informational content of media, is in part determined and limited by the abilities of the machine, which are in turn determined by its software. Software, the machines on which it runs, and the humans that use it, create the cyborg world.

As the cyborg displaces man/machine dualism, the cyborg world dissolves the dichotomy of physical space and cyberspace. Cyberspace is sometimes treated as vastly different in both its being and its attributes. This projected separateness of cyberspace enables optimistic theoretical views of it as beyond requiring full social and political protection. But, when we are machines rather than only their users, when our interactions with others are modulated by this shared space, when all is interface, cyberspace is not elsewhere. As cyberspace fully interpenetrates the real world, questions about a suitable politics for cyberspace, and its relationship to the politics of physical space, are replaced by questions of the politics of the cyborg world. To devise an appropriate normative political philosophy for the cyborg world, we need not turn away from the physical world to study cyberspace; rather, we must focus on the world in which man and machine have blended. The political philosopher's first task is to uncover the roots of power in this world.

They key to unlocking the modern nation-state, this vast assemblage of political and technical power, lies in its code:

> The only way to understand this strange machine-body of the State is to read the actual machine-body of the human being. . . . How can or ought the state work? What controls the automaton or machine and how might one decipher this control? (Gray and Mentor 1995, 221)

The act of reading such a machine-body enables us to understand the constitution of the machinic state. When the state resides in machines, we may decipher its meanings by reading its code. Reading the ‘text of the body politic’ (Gray and Mentor 1995, 221) becomes the decoding of the system's structures of control, as it always has. In the political institutions of days gone by, priests and kings derived authority from texts both sacred and profane. To interrogate these texts and decipher their meanings was to question the grounds of authority and uncover pathways for political change.

Code, Sovereignty and Cyberspace

Methodologically, the significance of protocols in the cyborg world can be illustrated effectively through example, considering the protocols that drive the Internet and the software that implements and interacts with them. The fundamental protocols of the Internet are open: their specifications are openly available in the form of Internet Standards. But the applications that depend on these protocols, through which we actually use the Internet, have equal impact on the character of the spaces created by the Internet. The physical structure of the Internet presents a suggestive story about the concentration of power – it contains ‘backbones’ and ‘hubs’ – but power on the Internet is not spatial but informational; power inheres in protocols. The techno-libertarian utopianism associated with the Internet, in the gee-whiz articulations of the Wired crowd, is grounded in an assumption that the novelty of governance by computer protocols precludes control by corporation or state. But those entities merely needed to understand the residence of power in protocol and to craft political and technical strategies to exert it.

Adherents of a ‘catechism of Net inviolability’ (Boyle 1997) believe that the basic architecture of the Net conveys an invisibility that insulates it from traditional applications of state and corporate power. But there is a difference between a politics that views the relationship of sovereign and citizen as mediated by the sovereign's exertion of state power through the imposition of rules, and one in which systemic power could be delivered through a diversity of material or technological pressure points (Boyle 1997). In the latter, the effect of such ‘a tightly knit grid of material coercions’ (Foucault 1980) is an opaqueness that makes this immanent power more subtle and thus harder to counteract. This Foucauldian perspective is especially relevant as an antidote to relentless optimism about the Net; the Net may be invulnerable to traditional modes of control and enforcement, but its architecture provides plenty of avenues for the exertion of state and corporate power.

As the Internet became a trading zone in the mid-1990s, governments were faced with crises of policy induced by the inadequacy of contemporary legal structures and enforcement mechanisms to contend with ‘[t]he treatment of content, the treatment of personal information, and the preservation of ownership rights’ (Reidenberg 1998) within and across national borders that straddled the Internet. This crisis was resolved by the deployment of technological solutions that functioned as legal proxies, as states increasingly called upon the technical power and flexibility of the Net (Rustad 2004). That is, 'law and governmental regulation are not the only source of law-making' (Reidenberg 1998).

Code in cyberspace is functionally equivalent to law in society (Lessig 2000). Our personal and social freedoms in this domain are precisely the freedoms granted by software and the protocols it implements. Lessig unpacks this equivalence via a fourfold taxonomy of constraints on human behavior: norms and conventions (social), physical restrictions (architectural), market restrictions (financial), and punitive restrictions (legal). These constraints originated in the physical world, but are now to be found as well on the Internet. Newsgroup members may consider some kinds of dialogue offensive and shun those that indulge in them; software may prevent us from using offensive language in the IRC channel #family; we may not be able to afford broadband Internet access; we may be subject to legal sanction for online exchanges of copyrighted material.

The constraint of law has multiple modalities: it can levy financial penalties, construct norms through the approbation often attached to illegal behaviors, and regulate architecture by, for example, mandating physical access for the disabled. In cyberspace, fundamentally a 'software world,' reconstruction is a matter of rewriting code. Because software determines interaction between user and machine, cyberspace is plastic, and is equally amenable to change by governments, criminals, law-abiding citizens, or corporations. Thus, code can function as law even when the laws of the physical world lag behind. Regulation on the Internet is implemented in both software and hardware, and modulated by local policies: content-filtering software may block child pornography; hardware firewalls may block intruders; the absence of effective legislation against spam is compensated by spam-filtering software and practices (Post 2000). As the Digital Millennium Copyright Act (DMCA) shows, however, legal and technical policy-making and enforcement can act in concert. Law has merged with architecture.

Thus, through its technological capability, government can impose rules on its citizens; the design choices it makes determine the spaces its citizens inhabit. Governmental policy imperatives are embedded in network designs, standards, and system configurations so that the structure of the technical system reveals the exertion of governmental power. No longer are issues of diplomacy or jurisdictional reach barriers to enforcing the law. States may simply use the 'long arm of the code' to implement decisions and policies that can have impact even outside their borders. The infrastructure of the Internet empowers the automatic enforcement of policies and decisions. Infrastructure design offers the state an ex ante means to assure that policy decisions are enforced. States can require that rules for the treatment of information be embedded within the technical system architecture. By 'hard-wiring' particular rules within the infrastructure, states preclude violations and automate the enforcement of public decisions. (Reidenberg 2003-2004, 218)

The government can attempt to 'shape code' through regulatory methods such as prohibitions, taxes, liability and copyright law, and disclosure requirements, while leaving 'safe harbors' for technology that does not clash with its imperatives (Kesan and Shah 2005). The DMCA provides a new layer of protections for digital content: under its terms, the technical measures taken by content providers to protect copyrighted material are backed up by governmental prohibitions on their circumvention. These provisions, which make it illegal to devise technologies that could bypass or disable content protections, turn the ‘code is law’ metaphor into reality (Lessig 1999). A programmer who writes code capable of decrypting the Content Scramble System (CSS) is in violation of the law. Even if the program's application is perfectly legitimate, such as the fair use of a legally purchased DVD, the act of writing the program is illegal. This combination of legislation and technology can extend the reach of protective regimes, taking away even those freedoms explicitly provided in copyright law.

The code of a content protection system like CSS is no longer software code but has become legal code as well. It grants to its owner the power to control the dissemination and distribution of culture: who gets it and how, who plays it, who hears it, who can share it and how. In the hypertechnical world, control passes to those who wield technical power. This power, if closed off by technological obfuscation, is opaque and beyond challenge. Our only chance for autonomous reactions rests in our ability to view this power.

Though the technologies of the Internet still pose challenges to the exertion of corporate or governmental power, their particular uses may reinforce such power. As illustrated by the ‘Net Neutrality’ controversy, the political economy of the networked world also locates power in corporations; control of the net can therefore be merely a reinforcement of their corporate imperatives. The control of cyberspace with restrictions implemented in code promises a world in which near-perfect control is possible: the logistic and financial costs associated with legal controls are greatly reduced, and fine-grained tracking ensures effortless supervision. The issue is no longer whether code should regulate. What matters now is a different set of questions: ‘Does it do so in the open? Is it transparent about its means? Does it advance values that we believe are fundamental?’ (Lessig 1999). A political philosophy that will do cyberspace justice must be cognizant of its unique population and the composition, structure, and organization of its politics and communities.

Language, Free Speech and Free Software

Richard Stallman has described the meaning of free software with the slogan, ‘Think 'free' as in 'free speech' not 'free beer.'‘ This slogan suggests that in the cyborg world, the relationship between free software and free speech is not an analogy but an identity. The ‘free’ in ‘free speech’ refers to absence of restriction on expression. The freedoms of free software entail the absence of restrictions on use and modification of the software. These freedoms apply to different modalities of software: they both guarantee the freedom to use software by protecting the availability of its executable and guarantee the freedom to access and modify the source code. If we consider both modalities of software – source code and executable – then the true resonance for ‘free’ as in ‘free speech’ is the freedom to use the software for any purpose: I am free to use the executables for any task and for distribution, while I am free to use the source code for the purpose of making derivative works. Therefore, software is not free when it is subject to restrictions on the availability of its source, the functionality it provides, the applications in which it is used, or the field of endeavor in which it is deployed. These restrictions should be rejected just as upholders of free speech reject restrictions on free speech, for the ramifications of these restrictions, in both contexts, are enormous both now and for our future polity. Yet restrictions are placed on speech in most democratic polities. Most commonly, speech is regulated when it can be shown to cause harm; software is deemed to be harmful when it facilitates the breaking of law, as in the DeCSS case.

In 1999, Jon Lech Johansen, a Norwegian student, wrote and published source code for software intended to circumvent DVD access control mechanisms. As part of the movie industry's technical protection of intellectual property, DVDs were encrypted using the proprietary Content Scramble System (CSS), which is intended to allow only licensed players to play DVDs. At the time, the only computers with licensed players were Windows- and Macintosh-based; Linux users could not play DVDs – even if legally purchased – on their machines. Called DeCSS because it reverses the operation of CSS, Johansen's software, circumventing the protective encryption, allowed DVDs to be played on Linux computers. As far as Linux users were concerned, this software merely enabled the fair use of their legally purchased DVDs, though the movie industry had a different perspective. In 2000 American authorities contacted Norwegian police, who raided Johansen's home; arrested and tried in Norwegian court, he was fully acquitted in 2003.

In the United States, Universal Studios filed suit under the terms of the DMCA against Eric Corley, Shawn C. Reimerdes, and Roman Kazan, Americans who had posted a copy of the DeCSS code on the Web site 2600.com. In the ensuing legal proceedings, the defense argued that source code is subject to First Amendment protections, citing the landmark decision in Bernstein v. US Dept. of Justice, which had found that ‘the particular language one chooses [does not] change the nature of language for First Amendment purposes.’ The Court agreed to an extent, but as a preamble for its finding in favor of placing restrictions on code, stated, ‘the long history of First Amendment jurisprudence makes equally clear that the fact that words, symbols and even actions convey ideas and evoke emotions does not inevitably place them beyond the power of government.’ While this has never been contested, even by First Amendment proponents, the Court seemed to be failing to confront the question of whose rights would be protected, and whose infringed, by such restrictions.

The Court concluded that the functional nature of code overshadows its expressive, speechlike aspects. After weighing the relative importance of consumers' fair use rights and content providers' protection, it ruled that DeCSS code was entitled only to a weak form of First Amendment protection, and found for the motion picture industry. Stripped of the veneer of a property rights debate, at its essence, this ruling reflects and reinscribes an old chauvinism that stresses the mechanic/organic, natural/synthetic, and biological/technological dichotomies. We suspect the issue is not ultimately one of functionality outweighing speech; it is simply that the Court cannot conceive of human-machine communication as speech.

The finding was appealed, and a group of computer scientists, in an amicus curiae brief, contested the Court's argument about the functionality of code. Their arguments seek to establish that software (whether source or object code) is not only an avenue of human expression but also one which should be subject to regulation only to the extent it is a form of speech. The amici point out that code is used not only to communicate with computers but with computer scientists as well. That is, code is an integral part of ‘a complex system of understood meanings within specific communities.’ Further, the expressive quality of source code, containing the ‘ideas, commands, objectives’ (Tyre

2001) of the programmer, is carried into the executable code during a translation process. Thus, code has both communicative and expressive aspects. Like any other form of speech, code can challenge power and ideology. The act of writing DeCSS was a fundamentally political one, contesting a particular unjust restriction on freedom. The Court's ruling, implicitly recognizing the political implications of this act, upholds that restriction. In the context of the cyborg world, the free speech protections for which the amici advocate generalize in the broadest sense to communication among its hybrid denizens.

Political Philosophy in the Cyborg World

Political philosophies are concerned with autonomy and the distribution of power: to enter political society is to enter into power-sharing relationships that may require the surrender of some autonomy. Comparative political theory measures, among other things, the degrees to which these surrenderings and agreements are employed by different political systems. Individual autonomy, the capacity of a person to alter the circumstances that affect her decision making, is a moral good in classical political philosophy. It is also a political tool used to identify and publicize oppression and injustice: prescriptions to increase individual autonomy in a sociopolitical arrangement are attempts to devise a more just society.

In the cyborg world, the mediation of our extended selves by closed software threatens individual autonomy; the advocacy for, and the provision of, closed software is a form of paternalism, diminishing cyborg autonomy as it controls and regulates the nature of human-machine interaction. The proprietary-software industry makes this paternalism explicit by suggesting the rejection of proprietary software will lead to the collapse of the industry; by using free software we hurt the software industry and its consumers, who will be denied its benefits. Thus, it is in consumers' best interest to continue to adopt closed software.

Autonomy characterizes the processes through which agents identify desires and take decisions. If this process is problematically constrained by external factors then the resulting decision and subsequent actions are not autonomously chosen. While consumers appear to consent to the use of closed software by accepting its licenses, closed software holds an effective monopoly in many application domains, limiting the extent to which consumers can grant their full consent. Subsequently, interactions with the software are determined entirely by the software itself; the user plays a passive role. A human agent functioning in this manner, uncritically accepting the constraints of closed software, is a 'happy slave,' convinced its desires to regulate its interactions with the machine are mere fantasies, not in accord with pragmatic economic realities. But to imagine that this position of dependence and servitude is one a rational agent would will for itself is incoherent. It is similarly implausible to assert that users desire the constraints imposed on them by software with limited, unalterable functionality. The ability to control one's interactions with the machine is not a specialized, esoteric concern, but is a core freedom in the cyborg world. Software, treated as constitutionally protected free speech, enables a full range of expression and protects the ability to manipulate technological artifacts; most fundamentally, it protects the autonomy of the individual.

Political and social institutions are legitimate to the extent they are subject to an 'endorsement constraint' (Dworkin 2000, 216-18). In democratic polities, such endorsement follows a process of participatory democratic discussion (Bessette 1994). When these institutions depend on software, freedom in both its choice and usage is the key to enabling public discussion and endorsement; a politics underwritten by the constraints of closed software is illegitimate.

Our polity is that of the cyborg world, one in which distinctions between man and machine, natural and computer languages, have been displaced. In this world, governmentality resides in machines; our spaces are constructed by technology; we are hybrids of biology and technology. If not only regulation but also political function devolves to code, then we must place the same normative constraints on the technology that we place on the socio-human-political machinery. The social and political philosophy of such a world must capture the technological inflection of its material forms of life. Our arguments about treating software as free speech, then, amount to a claim that software must be protected as speech in the cyborg world. More generally, the construction of such a philosophy requires the selection of designs and specifications for this hybrid world, for there is a strong connection between principles for designing an online virtual community and those for governing any technologically constructed space. As governments exert power through and over code, the 'programmed polity' is already with us in the contemporary cyborg world; contemporary 'e-government' initiatives provide powerful illustrations of the importance of free software in this world.

E-Government

Driven by technocratic imperatives of efficiency and cost-saving, as well as populist imperatives of accessibility, e-government promises to 'modernize' government. Evoking the rhetorical disjunction between the free and open source movements, most discussion about FOSS in the context of e-government centers on concerns with cost and technical efficacy. There is, however, occasional 'negligible, parenthetical and delphic' (Berry and Moss 2006) mention of using FOSS to promulgate social goals having to do with quality of life, citizen engagement, and social inequality. Cost is certainly a political issue, especially in the developing world, but more significant is FOSS's political and moral message of transparency; by ignoring this aspect of free software, governments pass over an opportunity to remake their machinery toward a different relationship with the polity (Berry and Moss 2006).

Transparency in government requires open procedures so they may be the subject of public scrutiny and critique toward the ends of accountability and legitimacy. In the context of e-government, the characteristics of government depend on the characteristics of code. A system of e-government built on closed software is itself closed, one whose laws and policies are unknowable; by closing off participation, it denies the public nature of democracy. Indeed, many governmental activities only become trustworthy when the code that runs them is open. With FOSS, the innards of governments are laid bare. Their conduct becomes a subject of public inquiry and accountability. This transparency is not sufficient for a democratic polity, but is a minimum standard.

The open code of e-government would only be studied by a very few technically competent people, just as laws and legal decisions are largely impenetrable to most of us.

As most of us are not able to interpret the texts of Congressional transcripts or legislation, lawyers, with their technical training, serve as proxies for the citizenry in this regard. The mere fact that these documents can be publicly read affects the functioning of the system, and leads to a self-regulation of conduct by political and legislative actors (Berry and Moss 2006, 28).

E-government based on closed source carries the potential to create a political subject similar to the user of proprietary software: passive and uncritical. A collective, participatory approach to creating the code of government would mitigate these dangers. FOSS, then, provides an opportunity for us to modify the code of government. The FOSS mode of open communication could become a model for political discourse: the populace could actively intervene by 'developing,' 'bug-fixing,' and 'iterating' true participatory government. Citizens could examine the source code of government to determine how it encodes values important to them, and modify it if they deem it unacceptable. For example, it is possible to modify the source code of open source Web browsers, potentially integral components of e-government apparatus, to support privacy and the user's right to give informed consent (Friedman, Howe, and Felten 2002).

The interplay of governmentality and technology is particularly visible in the challenges of electronic voting. Oversight of elections, considered by many to be the cornerstone of modern representational democracies, is a governmental function; election commissions are responsible for generating ballots; designing, implementing, and maintaining the voting infrastructure; coordinating the voting process; and generally insuring the integrity and transparency of the election. But modern voting technology, specifically that of the computerized electronic voting machine that utilizes closed software, is not inherently in accord with these norms. In elections supported by these machines, a great mystery takes place. A citizen walks into the booth and 'casts a vote.' Later, the machine announces the results. The magical transformation from a sequence of votes to an electoral decision is a process obscure to all but the manufacturers of the software. The technical efficiency of the electronic voting process becomes part of a package that includes opacity and the partial relinquishing of citizens' autonomy.

The opaqueness of these machines' design is a secret compact between governments and manufacturers of electronic voting machines, who alone are privy to the details of the voting process. The norms that govern the use of these machines must be encapsulated in citizens' requirements, among which transparency is foremost. Access to their source code provides the polity an explanation of how voting results are reached, just as publicly available transcripts of congressional sessions illustrate governmental decision-making. The use of FOSS would ensure that, at minimum, technology is held to the same standards of openness.

Conclusion: Free Software as an Anarchist Ideal

The activities, mechanisms, and power of government are constitutive of political systems. In an open polity, the government makes the content of laws available for inspection and debates. To use an anarchist term, this power is transparent: the reasons for a particular interaction are available for any citizen to view. When these are concealed, the polity is subject to a system of opaque power. Any arrangement that negates our ability autonomously to affect the material circumstances of the cyborg world, most crucially the behavior of computing devices, is one that creates and sustains opaque power relations. In a system governed by such relations, no identifiable person or institution bears ultimate responsibility for decisions. In the cyborg world, a ring of software agents – the true face of interaction with the polity – surrounds the human core of government. If the software is proprietary, un-free, the relationships it mediates remain opaque. Such a system allows for the concealed, untrammeled growth of governmental and corporate power through exquisite systems of control that make tracking, surveillance, privacy and trespass invasions, and restrictions on sharing of information a matter of course. This co-optation of a supposedly liberatory technology is not inevitable. The potentials for both exquisite control and unfettered freedom lie in the way we choose to use this technology.

Most important, a cyborg world underwritten by free software renders participation in its political structure as contingent and flexible: because law is implemented as code in cyberspace, the polity can choose to change the extent and character of its acceptance (Lessig 2000). Governmental regulation requires 'an unmovable, and unmoving, target of regulation' (Lessig 2000, 106), an entity for whom the cost of obedience is lower than that of disobedience. Government regulation is often applied to intermediaries, such as Internet service providers or television manufacturers, which carry the effect of the regulation to citizens/consumers. Thus, citizens may experience, for example, restrictions on content without direct intervention by the government in their activities. To the extent these manufacturers and service providers – proxies for governmental power – are private entities, citizens have few tools with which to resist this intervention; if these proxies were public, instead, such opacity would be untenable.

If the code-as-law that regulates us is available for us to change, then it ceases to have a hold on us. If code is architecture, then the spaces created through our interaction with it are modifiable ones. But if adherence to code as law is voluntary, will not chaos result? The laws that citizens tolerate are laws enforced by coercion, they involve the backup arsenals of punitive restrictions, punishments, incarcerations, fines and the like. But when citizens can opt out of compliance, these laws lose their impact. This provides the opportunity for a creative moment in determining which strictures are relevant to which spheres of activity, whether political or cultural.

The mapping between anarchism and open code, closed code and the state, is revealed by the structural similarity of arguments for the indispensability of proprietary code and for the necessity of the state. Both these arguments rely on creation myths and idealized reconstructions. In the case of the state, the story goes, a Hobbesian state of nature was brought to an end when citizens banded together and submitted to the benevolent yet authoritarian Leviathan, resulting in a safer existence, the protection of property, and dramatically improved standards of living and culture. Similarly, the mythology of the software industry insists that at one time, users struggled with little code, most of it poorly written, only to be rescued by technocratic entrepreneurs who, insisting on technical and business standards, brought truly useful, life-enhancing software to the people and employment to hundreds of thousands of programmers. But, as we have seen, the embryonic software industry drew heavily on, and subsequently depleted, a then-flourishing hacker culture inhabited by hobbyists, master programmers imbued with both social and technical vision. And anarchist history suggests the creation of the modern state came at great societal cost, accompanied by much violence, most notably the destruction of the medieval city and its trade guilds (Kropotkin 1902).

Because the cyborg world brings to life new legal and political structures, as code merges with law, we can see in its fundamentals the glimmerings of a new, transparent society, one that by making participation in it voluntary attains the true meaning of a compact, one achieved without coercion. When code is opened, we have the power to view the machinery of authority. The panopticon is inverted, and we observe the state, or what remains of it.

References

Berry, David M., and Giles Moss. 2006. Free and Open Source Software: Opening and Democratising E-Government's Black Box. *Information Polity* 11:21–34.

Bessette, Joseph. 1994. *The Mild Voice of Reason: Deliberative Democracy and American National Government*. Chicago: University of Chicago Press.

Borgmann, Albert, and N. Katharine Hayles. 1999. An Interview/Dialog on Humans and Machines. University of Chicago Press, http://www.press.uchicago.edu/Misc/Chicago/borghayl.html.

Boyle, James. 1997. Foucault in Cyberspace: Surveillance, Sovereignty, and Hard-Wired Censors. *University of Cincinnati Law Review* 66:177.

Clynes, Manfred E., and Nathan S. Kline. 1960. Cyborgs and Space. *Astronautice* 14 (9): 26–27, 74–76.

Dworkin, Ronald. 2000. *Sovereign Virtue: The Theory and Practice of Equality*. Cambridge, MA: Harvard University Press.

Foucault, Michel. 1980. Two Lectures. In *Power/Knowledge: Selected Interviews and Other Writings, 1972–1977*, edited by C. Gordon. New York: Pantheon.

Friedman, B., D. C. Howe, and E. Felten. 2002. Informed Consent in the Mozilla Browser: Implementing Value Sensitive Design. Paper read at 35th Annual Hawaii International Conference on System Sciences (HICSS'02).

Gray, Chris Hables, and Steven Mentor. 1995. The Cyborg Body Politic and the New World Order. In *Prosthetic Territories: Politics and Hypertechnologies*, edited by G. Brahm and M. Driscoll. Colorado Springs: Westview Press.

Kesan, Jay P., and Rajiv C. Shah. 2005. Shaping Code. *Harvard Journal of Law & Technology* 18 (2): 319–99.

Lessig, Lawrence. 1999. The Code Is the Law. *The Standard*, April 9, http://www.lessig.org/content/standard/0,1902,4165,00.html.

Lessig, Lawrence. 2000. *Code and Other Laws of Cyberspace*. New York: Basic Books.

Post, David G. 2000. Internet: Of Black Holes and Decentralized Law-Making in Cyberspace. *2 Vanderbilt Journal of Enterntainment Law and Practice 70.*

Reidenberg, Joel E. 1998. Lex Informatica: The Formulation of Information Policy Rules Through Technology. *Texas Law Review* 76 (3): 553–84.

Reidenberg, Joel E.. 2003–2004. States and Internet Enforcement. *University of Ottawa Law and Technology Journal* 1 (1–2): 213–30.

Rustad, Roger E., Jr. 2004. Joel Reidenberg on Hack Toolz, Lex Informatica, and Affirming Non-US Democratic Values. March 23, http://grep.law.harvard.edu/articles/04/03/23/1640243.shtml.

Tyre, James. 2001. Brief of Amici Curiae, Universal City Studios, Inc., et al. vs. Eric Corley, a/k/a Emmanuel Goldstein, 2600 Enterprises, Inc., Shawn C. Reimerdes, Roman Kazan. http://cryptome.org/mpaa-v-2600-bac.htm

Book Publication Announcement

We are pleased to announce the publication of:

Samir Chopra and Scott D. Dexter. *Decoding Liberation: The Promise of Free and Open Source Software*. New York, NY: Routledge, New Media and Cyberculture Series, 2007.

http://www.sci.brooklyn.cuny.edu/~bcfoss/DL/

Software is more than instructions for computing machines: it enables (and disables) political imperatives and policies. Nowhere is this potential for radical social and political change more apparent than in the practice and movement known as free software. Free software makes the knowledge and innovation of its creators publicly available. This liberation of code—celebrated in free software's explicatory slogan "Think free speech, not free beer"—is the foundation, for example, of the GNU/Linux phenomenon.

In *Decoding Liberation*, we provide a synoptic perspective on the relationships between free software and freedom. We begin by asking, "What is the emancipatory potential of free software and how is it manifested?" and suggest some answers by focusing on five main themes: free software's reworking of economic concepts such as property and production, the ethical import of free software for communities and individuals, the facilitation of creativity, the objectivity of computing as a scientific practice, and the role of software in a cyborg world. While "open source" continues to be the focus of business hype, we argue that the truly exciting phenomenon is free software, which promises to transform not only technology and business but society as well.

Contents:
Chapter 1: Free Software and Political Economy
Chapter 2: The Ethics of Free Software
Chapter 3: Free Software and the Aesthetics of Code
Chapter 4: Free Software and the Scientific Practice of Computer Science
Chapter 5: Free Software and the Political Philosophy of the Cyborg World