INTRODUCTION

A Tale of Two Worlds

Free and open-source software (F/OSS) refers to nonproprietary but licensed software, much of which is produced by technologists located around the globe who coordinate development through Internet-based projects. The developers, hackers, and system administrators who make free software routinely include the following artifact in the software they write:

> This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

While seemingly insignificant, this warning is quite meaningful for it reveals something important about the nature of free software and my subsequent representation of it. This legal notice is no doubt serious, but it also contains a subtle irony available to those who know about free software. For even if developers cannot legally guarantee the so-called FITNESS of software, they know that in many instances free software is often as useful as or in some cases superior to proprietary software. This fact brings hackers the same sort of pleasure, satisfaction, and pride that they derive when, and if, they are given free reign to hack. Further, even though hackers distribute their free software WITHOUT ANY WARRANTY, the law nevertheless enables them to create the software that many deem superior to proprietary software—software that they all "hope [. . .] will be useful." The freedom to labor within a framework of their own making is enabled by licenses that cleverly reformat copyright law to prioritize access, distribution, and circulation. Thus, hackers short-circuit the traditional uses of copyright: the right to exclude and control.

This artifact points to the GNU General Public License (GPL), an agreement that many hackers know well, for many use it (or other similar licenses) to transform their source code—the underlying directions of all software—into "free software." A quick gloss of the license, especially its preamble, reveals a more passionate language about freedom and rights:

> When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.[1]

This type of language spills far beyond licensing agreements. It is routinely voiced in public discourse and everyday conversation. Such commitments to freedom, access, and transparency are formalized in a Linux distribution known as Debian, one of the most famous free software projects. These values are reflected in a pair of charters—the Debian Constitution and the Debian Social Contract—that articulate an organizational vision and formulate a set of promises to the wider free software community. These charters' names alone unmistakably betray their liberal roots, even if they were not explicitly created with the goal of "advancing" liberal ideals.

By liberalism, I do not mean what may first come to mind: a political party in Europe usually associated with politicians who champion free market solutions, or in the United States, a near synonym for the Democratic Party. Nor is it just an identity that follows from being a proud, card-carrying member of the American Civil Liberties Union or Electronic Frontier Foundation, although these certainly can be markers.

Here I take liberalism to embrace historical as well as present-day moral and political commitments and sensibilities that should be familiar to most readers: protecting property and civil liberties, promoting individual autonomy and tolerance, securing a free press, ruling through limited government and universal law, and preserving a commitment to equal opportunity and meritocracy. These principles, which vary over time and place, are realized institutionally and culturally in various locations at different times. Perhaps the most famous of these are the institutions of higher education, market policies set by transnational institutions, and the press, but they are also at play on the Internet and with computer hackers, such as those who develop free software.[2]

The small statement that prefaces the GNU GPL thus hints at two elements of this community: one is esoteric, and grounded in technology and its material practices; and the other concerns a broader, culturally familiar vision of freedom, free speech rights, and liberalism that harks back to constitutional ideals. We should not take either for granted but instead open them up to critical reflection, and one route to do so is by bringing them together. This ethnography takes seriously free software's visions of liberty and freedom as well as the mundane artifacts that hackers take pleasure and joy in creating. In considering them together, important lessons are revealed about the incomplete, sometimes fraught, but nonetheless noticeable relationship between hacking and liberalism, and the transformations and tensions evident within the liberal tradition and computer hacking.

A Liberal Critique within Liberalism

The terms free and open as applied to software are distinct yet often come paired. This is in part because they designate the same alternative licenses and collaborative methodologies, but they differ in their moral orientation: the term free software foremost emphasizes the right to learn and access knowledge, while open source tends to flag practical benefits.[3] Many participants, whether they are volunteers or corporate employees paid to work on free software, refer to themselves with pride as hackers—computer aficionados driven by an inquisitive passion for tinkering and learning technical systems, and frequently committed to an ethical version of information freedom.

Although hackers hold multiple motivations for producing their software, collectively they are committed to *productive freedom*. This term designates the institutions, legal devices, and moral codes that hackers have built in order to autonomously improve on their peers' work, refine their technical skills, and extend craftlike engineering traditions. This ethnography is centrally concerned with how hackers have built a dense ethical and technical practice that sustains their productive freedom, and in so doing, how they extend as well as reformulate key liberal ideals such as access, free speech, transparency, equal opportunity, publicity, and meritocracy.

I argue that F/OSS draws from and also rearticulates elements of the liberal tradition. Rather than designating only a set of explicitly held political, economic, or legal views, I treat liberalism in its cultural registers.[4] Free software hackers culturally concretize a number of liberal themes and sensibilities—for example, through their competitive mutual aid, avid free speech principles, and implementation of meritocracy along with their frequent challenge to intellectual property provisions. Indeed, the ethical philosophy of F/OSS focuses on the importance of knowledge, self-cultivation, and self-expression as the vital locus of freedom. Hackers bring these values into being through an astounding range of social and technical practices, covered in detail throughout this book.

Because hackers challenge one strain of liberal jurisprudence, intellectual property, by drawing on and reformulating ideals from another one, free speech, the arena of F/OSS makes palpable the tensions between two of the most cherished liberal precepts—both of which have undergone a significant deepening and widening in recent decades. Thus, in its political dimension, and even if this point is left unstated by most developers and advocates, F/OSS represents a liberal critique from within liberalism. Hackers sit simultaneously at the center and margins of the liberal tradition.

The expansion of intellectual property law, as noted by some authors, is part and parcel of a broader neoliberal trend to privatize what was once public or under the state's aegis, such as health provision, water delivery, and military services. "Neoliberalism is in the first instance," writes David Harvey (2005, 2), "a theory of political economic practices that proposes human well-being can be best advanced by liberating entrepreneurial freedoms and skills within an institutional framework characterized by strong property rights, free markets, and free trade." As such, free software hackers not only reveal a long-standing tension within liberal legal rights but also offer a targeted critique of the neoliberal drive to make property out of almost anything, including software.

While most of this ethnography illustrates how free software hacking critiques neoliberal trends and reinvents liberal ideals by asserting a strong conception of productive freedom in the face of intellectual property restrictions, it also addresses the material, affective, and aesthetic dimensions of hacking. In pushing their personal capacities and technologies to new horizons (and encountering many frustrations along the way), hackers experience the joy that follows from the self-directed realization of skills, goals, and talents. At times, hacking provides experiences so completely overpowering, they hold the capacity to shred self-awareness, thus cutting into a particular conception of the liberal self—autonomous, authentic, and rational—that these hackers otherwise routinely advance. Thus, at least part of the reason that hacker ethics takes its liberal form is connected to the aesthetic experiences of hacking, which are informed by (but not reducible to) liberal idioms and grammars. Hacking, even if tethered to liberal ideologies, spills beyond and exceeds liberal tenets or liberal notions of personhood, most often melding with a more romantic sensibility concerned with a heightened form of individual expression, or in the words of political theorist Nancy Rosenblum (1987, 41), a "perfect freedom."

Fieldwork among Hackers

For most of its history, anthropology stuck close to the study of non-Western and small-scale societies. This started to shift following a wave of internal and external critiques that first appeared in the 1960s, expanded in the 1970s, and peaked in the 1980s. Now referred to as "the critical turn in anthropology," the bulk of the critique was leveled against the discipline's signature concept: culture. Critics claimed that the notion of culture—as historically and commonly deployed—worked to portray groups as far more bounded, coherent, and timeless than they actually are, and worse, this impoverished rendition led to the omission of topics concerning power, class, colonialism, and capitalism (Abu-Lughod 1991; Asad 1973; Clifford 1988; Clifford and Marcus 1986; Dirks 1992; Said 1978). Among other effects, the critique cracked open new theoretical and topical vistas for anthropological inquiry. An anthropologist like myself, for example, could legitimately enter nontraditional "field sites" and address a new set of issues, which included those of technoscientific practice, information technologies, and other far-flung global processes stretching from labor migration to transnational intellectual property regulations.

Partly due to these disciplinary changes, in winter 2000, I left a snowy Chicago and arrived in a foggy San Francisco to commence what cultural anthropologists regard as our foundational methodological enterprise: fieldwork. Based on the imperative of total immersion, its driving logic is that we can gain analytic insight by inserting ourselves in the social milieu of those we seek to understand. Fieldwork mandates long-term research, usually a year or more, and includes a host of activities such as participating, watching, listening, recording, data collecting, interviewing, learning different languages, and asking many questions.

When I told peers of my plan to conduct fieldwork among hackers, many people, anthropologists and others, questioned it. How does one conduct fieldwork among hackers, given that they just hang out by themselves or on the Internet? Or among those who do not understand the name, given that they are all "outlaws"? Often playfully mocking me, many of my peers not only questioned how I would gather data but also routinely suggested that my fieldwork would be "so easy" (or "much easier than theirs") because I was studying hackers in San Francisco and on the Internet.

The subtext of this light taunting was easy enough to decipher: despite the transformations in anthropology that partially sanctioned my research as legitimate, my object of study nonetheless still struck them as patently atypical. My classmates made use of a socially acceptable medium—joking—to raise what could not be otherwise discussed openly: that my subjects of study, primarily North American and European (and some Latin American) hackers, were perhaps too close to my own cultural world for critical analysis, or perhaps that the very activity of computing (usually seen as an instrumental and solitary activity of pure rationality) could be subject only to thin, anemic cultural meanings.[5]

By the turn of the twenty-first century, although anthropology had certainly "reinvented" itself as a field of study—so that it is not only acceptable but one is in fact, at some level, also actively encouraged to study the West using new categories of analysis—Michel-Rolph Trouillot (2003, 13) has proposed that "anthropologists reenter the West cautiously, through the back door, after paying their dues elsewhere." As a young, aspiring anthropologist who was simply too keen on studying free software during graduate school and thus shirked her traditional dues, I knew that for myself as well as my peers, my project served as an object lesson in what constitutes an appropriate anthropological "location" (Gupta and Ferguson 1997) for study—in particular for graduate students and young scholars.

I myself wondered how I would ever recognize, much less analyze, forms of cultural value among a group of mostly men of relatively diverse class and national backgrounds who voluntarily band together online in order to create software. Would I have to stretch my ethnographic imagination too far? Or rely on a purely formal and semiotic analysis of texts and objects—a methodology I wanted for various reasons to avoid? Amid these fears, I took some comfort in the idea that, as my peers had indicated, my initial fieldwork would be free of much of the awkwardness that follows from thrusting oneself into the everyday lives of those who you seek to study, typically in an unfamiliar context. At the very least, I could communicate to hackers in English, live in a familiar and cosmopolitan urban setting, and at the end of the day, return to the privacy and comfort of my own apartment.

As it turned out, my early ethnographic experiences proved a challenge in many unexpected ways. The first point of contact, or put more poetically by Clifford Geertz (1977, 413), "the gust of wind stage" of research, was harder than I had imagined. Although not always discussed in such frank terms among anthropologists, showing up at a public gathering, sometimes unannounced, and declaring your intent to stay for months, or possibly years, is an extraordinarily difficult introduction to pull off to a group of people you seek to formally study. More difficult is describing to these strangers, whose typical understanding of anthropology stems from popular media representations like the Indiana Jones trilogy, our methodology of participant observation, which is undertheorized even among anthropologists.[6] Along with the awkwardness I experienced during the first few weeks of fieldwork, I was usually one of the only females present during hacker gatherings, and as a result felt even more out of place. And while I may have recognized individual words when hackers talked shop with each other—which accounted for a large percentage of their time—they might as well have been speaking another language.

At the start of my research period, then, I rarely wanted to leave my apartment to attend F/OSS hacker social events, user group meetings, or conferences, or participate on email lists or Internet relay chat channels—all of which were important sites for my research. But within a few months, my timidity and ambivalence started to melt away. The reason for this dramatic change of heart was a surprise to me: it was the abundance of humor and laughter among hackers. As I learned more about their technical world and was able to glean their esoteric jokes, I quickly found myself enjoying the endless stream of jokes they made in all sorts of contexts. During a dinner in San Francisco's Mission district, at the office while interning at the Electronic Frontier Foundation, or at the monthly gatherings of the Bay Area Linux User Group held in a large Chinatown restaurant, humor was a constant bedfellow.

Given the deep, bodily pleasures of laughter, the jovial atmosphere overcame most social barriers and sources of social discomfort, and allowed me to feel welcome among the hackers. It soon became clear to me, however, that this was not done for my benefit; humor saturates the social world of hacking. Hackers, I noticed, had an exhaustive ability to "misuse" most anything and turn it into grist for the humor mill. Once I began to master the esoteric and technical language of pointers, compilers, RFCs, i386, X86, AMD64, core dumps, shells, bash, man pages, PGP, GPG, gnupg, OpenPGP, pipes, world writeable, PCMCIA, chmod, syntactically significant white space, and so on (and really on and on), a rich terrain of jokes became sensible to me.

My enjoyment of hacker humor thus provided a recursive sense of comfort to a novice ethnographer. Along with personally enjoying their joshing around, my comprehension of their jokes indicated a change in my outsider status, which also meant I was learning how to read joking in terms of pleasure, creativity, and modes of being. Humor is not only the most crystalline expression of the pleasures of hacking (as I will explore later). It is also a crucial vehicle for expressing hackers' peculiar definitions of creativity and individuality, rendering partially visible the technocultural mode of life that is computer hacking. As with clever technical code, to joke in public allows hackers to conjure their most creative selves—a performative act that receives public (and indisputable) affirmation in the moment of laughter. This expression of wit solidifies the meaning of archetypal hacker selves: self-determined and rational individuals who use their well-developed faculties of discrimination and perception to understand the "formal" world—technical or not—around them with such perspicuity that they can intervene virtuously within this logical system either for the sake of play, pedagogy, or technological innovation. In short, they have playfully defiant attitudes, which they apply to almost any system in order to repurpose it.

A few months into my research, I believed that the primary anthropological contribution of this project would reside in discussing the cultural mores of computer hacking, such as humor, conjoined with a methodological analysis of conducting research in the virtual space of bits and bytes. Later in my fieldwork, I came to see the significance of another issue: the close relationship between the ethics of free software and the normative, much broader regime of liberalism. Before expanding on this connection, I will first take a short ethnographic detour to specify *when* it became unmistakably apparent that this technical domain was a site where liberal ideals, notably free speech, were not only endowed with concrete meaning but also made the fault lines and cracks within liberalism palpably visible.

It was August 29, 2001, and a typical San Francisco day. The abundant morning sun and deep blue skies deceptively concealed the reality of much cooler temperatures. I was attending a protest along with a group of about fifty programmers, system administrators, and free software enthusiasts who were demanding the release of a Russian programmer, Dmitry Sklyarov, arrested weeks earlier in Las Vegas by the Federal Bureau of Investigation (FBI) as he left Defcon, the largest hacker conference in the world. Arrested at the behest of the Silicon Valley software giant Adobe, Sklyarov was charged with violating the recently ratified and controversial Digital Millennium Copyright Act (DMCA). He had written a piece of software, the Advanced eBook Processor software, for his Russian employer. The application transforms the Adobe eBook format into the Portable Document Format (PDF). In order for the software to perform this conversion, it breaks and therefore circumvents the eBook's copy control measures. As such, the software violated the DMCA's anticircumvention clause, which states that "no person shall circumvent a technological protection measure that effectively controls access to a work protected under this measure."[7]

We had marched from the annual LinuxWorld conference being held in San Francisco's premier conference center, the Moscone Center, to the federal prosecutor's office. Along the way, a few homeless men offered solidarity by raising their fists. Two of them asked if we were marching to "Free Mumia"—an assumption probably influenced by the recent string of protests held in Mumia Abu-Jamal's honor. Indeed, as I learned soon after my first arrival in San Francisco, the city is one of the most active training grounds in the United States for radical activists. This particular spring and summer was especially abuzz with activity, given the prominence of counterglobalization mobilizations. But this small and intimate demonstration was not typical among the blizzard of typically left-of-center protests, for none of the participants had a way of conveying quickly nor coherently the nature of the arrest, given how it was swimming in an alphabet soup of acronyms, such as DRM, DMCA, and PDF, as opposed to more familiar ideas like justice and racism. A few members of our entourage nonetheless heartily thanked our unlikely though clearly sympathetic supporters, and assured them that while not as grave as Mumia's case, Dmitry's situation still represented an unfair targeting by a corrupt criminal justice system, especially since he was facing up to twenty-five years in jail "simply for writing software."

Once at the Hall of Justice, an impassioned crew of programmers huddled together and held up signs, such as "Do the Right Thing," "Coding Is Not a Crime," and "Code Is Speech."

There must have been something about directly witnessing such fiery outpourings among people who tend to shy away from overt forms of political action that led me to evaluate anew the deceptively simple claim: code is speech. It dawned on me that day that while I had certainly heard this assertion before (and in fact, I was only hearing it increasingly over time), it was more significant than I had earlier figured. And after some research,

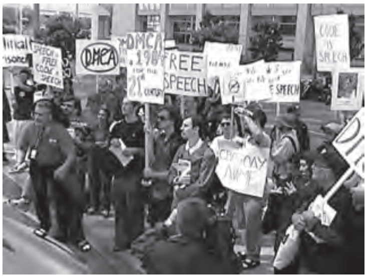

Figure Intro.1. Protesting the DMCA, San Francisco
Photo: Ed Hintz.

it was clear that while the link between free speech and source code was fast becoming entrenched as the new technical common sense among many hackers, its history was remarkably recent. Virtually nonexistent in published discourse before the early 1990s, this depiction now circulates widely and is routinely used to make claims against the indiscriminate application of intellectual property law to software production.

Early in my research, I was well aware that the production of free software was slowly but consistently dismantling the ideological scaffolding supporting the expansion of copyright and patent law into new realms of production, especially in the US and transnational context. Once I considered how hackers question one central pillar of liberal jurisprudence, intellectual property, by reformulating ideals from another one, free speech, it was evident that hackers also unmistakably revealed the fault line between two cherished sets of liberal principles.

While the two-hundred-year history of intellectual property has long been freighted with controversies over the scope, time limits, and purpose of various of its instruments (Hesse 2002; Johns 2006, 2010; McGill 2002), legal scholars have only recently given serious attention to the uneasy coexistence between free speech and intellectual property principles (McLeod 2007; Netanel 2008; Nimmer 1970; Tushnet 2004). Copyright law, in granting creators significant control over the reproduction and circulation of their work, limits the deployment of copyrighted material in other expressive activity, and consequently censors the public use of certain forms of expressive content. Legal scholar Ray Patterson (1968, 224) states this dynamic eloquently in terms of a clash over the fundamental values of a democratic society: "A society which has freedom of expression as a basic principle of liberty restricts that freedom to the extent that it vests ideas with legally protected property interests."

Because a commitment to free speech and intellectual property is housed under the same roof—the US Constitution—the potential for conflict has long existed. For most of their legal existence, however, conflict was noticeably absent, largely because the scope of both free speech and intellectual property law were more contained than they are today. It was only during the course of the twentieth-century that the First Amendment and intellectual property took on the unprecedented symbolic and legal meanings they now command in the United States as well as many other nations. (Although the United States has the broadest free speech protections in the world, many other Western nations, even if they limit the scope of speech, have also expanded free speech and intellectual property protections in the last fifty years.)

For example, copyright, which grants authors significant control over their expression of ideas, was initially limited to fourteen years with one opportunity for renewal. Today, the copyright term in the United States has ballooned to the length of the author's life plus seventy years, while works for hire get ninety-five years, regardless of the life of the author. The original registration requirement has also been eliminated. Most any expression—a scribble on a piece of paper, a blog post, or a song—automatically qualifies for protection, so long as it represents the author's creation.

Free speech jurisprudence follows a similar trajectory. Even though the Constitution famously states that "Congress shall make no law [. . .] abridging the freedom of speech, or of the press," during the first half of the twentieth century the US Supreme Court curtailed many forms of speech, such as political pamphleteering, that are now taken to represent the heart and soul of the democratic process. It is thus easy to forget that the current shape of free speech protections is a fairly recent social development, largely contained within the last fifty years (Bollinger and Stone 2002).

Due to the growing friction between free speech and intellectual property, US courts in the last twenty-five years have openly broached the issue by asserting that any negative consequences of censoring speech are far outweighed by the public benefit of copyright law. In other words, as a matter of public policy, copyright law represents an acceptable restriction on speech because it is the basis for what is designated as "the marketplace of ideas."[8] The theory animating the marketplace of ideas is that if and when ideas are allowed to publicly compete with each other, the truth—or in its less positivist form, the best policy—will become evident.

Given this historical trajectory, the use of F/OSS licenses challenges the current, intellectual property regime, growing ever more restrictive, and thus dubbed ominously by one legal scholar as the contemporary motor for "the second enclosure movement" (Boyle 2003). Many free software developers do not consider intellectual property instruments as the pivotal stimulus for a marketplace of ideas and knowledge. Instead, they see them as a form of restriction so fundamental (or poorly executed) that they need to be counteracted through alternative legal agreements that treat knowledge, inventions, and other creative expressions not as property but rather as speech to be freely shared, circulated, and modified.

The Aesthetics of Hacking

If free software hackers render the tensions between two liberal principles visible, and offer a targeted, if not wholesale, critique of neoliberalism in challenging intellectual property law (but rarely using the language of neoliberalism), their commitment to free speech also puts forth a version of the liberal person who strays from the dominant ideas of liberal personhood: a self-interested consumer and rational economic seeker. Among academics, this has often been placed under the rubric of "possessive individualism," defined as "those deeply internalized habits of thinking and feeling [. . .] viewing everything around them primarily as actual or potential commercial property" (Graeber 2007, 3; see also Macpherson 1962). Among hackers, selfhood has a distinct register: an autonomous being guided by and committed to rational thought, critical reflection, skills, and capacity—a set of commitments presupposed in the free speech doctrine (Peters 2005).[9]

However important these expressive and rational impulses are among programmers, they don't fully capture the affective stances of hackers, most notably their deep engagement, sometimes born of frustration, and at other times born of pleasure, and sometimes, these two converge. Soon after commencing fieldwork, what I quickly learned is that hacking is characterized by a confluence of constant occupational disappointments *and* personal/collective joys. As many writers have noted, and as I routinely observed, hacking, whether in the form of programming, debugging (squashing errors), or running and maintaining systems (such as servers), is consistently frustrating (Rosenberg 2007; Ullman 2003). Computers/software are *constantly* malfunctioning, interoperability is frequently a nightmare to realize, users are often "clueless" about the systems they use (and therefore break them or require constant help), the rate and pace of technological change is relentless, and meeting customer expectations is nearly impossible to pull off predictably. The frustration that generally accompanies the realities of even mundane technical work is depicted as swimming with sharks in *xkcd*, one of the most beloved geeks' comic strips (figure Intro.2).

What this comic strip captures is how hackers, as they work, sometimes swim in seas of frustration. To tinker, solve problems, and produce software, especially over one's lifetime, will invariably be marked by varying degrees of difficulties and missteps—a state of laboring that one theorist of craftspersonship describes as material "resistance" (Sennett 2008). In encountering obstacles, adept craftspeople, such as hackers, must also build an abundant "tolerance for frustration" (ibid., 226), a mode of coping that at various points will break down, leading, at best, to feelings of frustration, and at worst, to anguish and even despair and burnout.

Despite these frustrations and perhaps because of them, the craft of hacking demands a deep engagement from hackers, or a state of being most commonly referred to in the literature as "flow" (Csikszentmihalyi 1990).

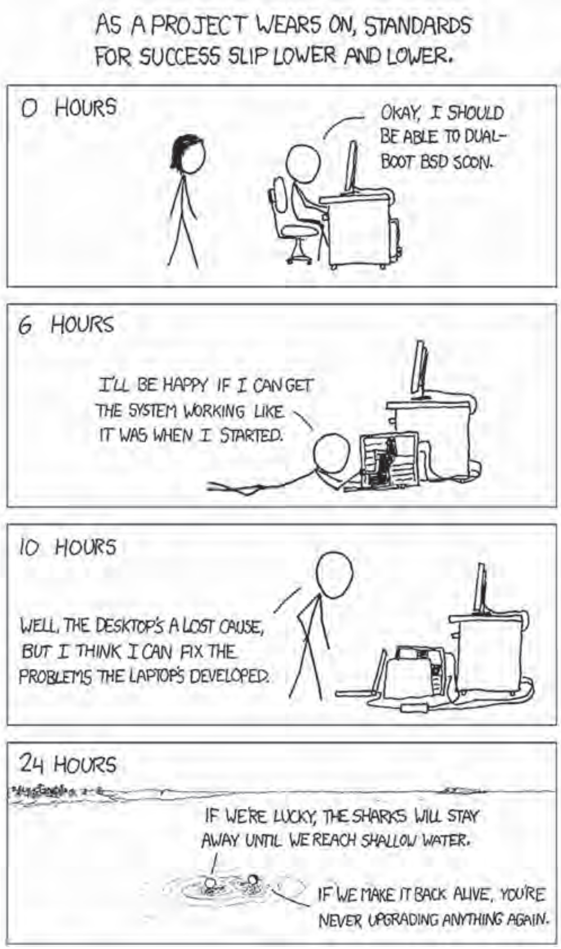
AS A PROJECT WEARS ON, STANDARDS FOR SUCCESS SLIP LOWER AND LOWER.
0 HOURS
OKAY, I SHOULD BE ABLE TO DUAL-BOOT BSD SOON.
6 HOURS
I'LL BE HAPPY IF I CAN GET THE SYSTEM WORKING LIKE IT WAS WHEN I STARTED.
10 HOURS
WELL, THE DESKTOP'S A LOST CAUSE, BUT I THINK I CAN FIX THE PROBLEMS THE LAPTOP'S DEVELOPED.
24 HOURS
IF WE'RE LUCKY, THE SHARKS WILL STAY AWAY UNTIL WE REACH SHALLOW WATER.
IF WE MAKE IT BACK ALIVE, YOU'RE NEVER UPGRADING ANYTHING AGAIN.

Figure Intro.2. "Success," *xkcd*
Credit: Randall Munroe.

In its more mild and commonplace form, hacker pleasure could be said to approximate the Aristotelian theory of eudaemonia, defined succinctly by philosopher Martha Nussbaum (2004, 61) as "the unimpeded performance of the activities that constitute happiness." In pushing their personal capacities and skills though playing around with and making technologies, hackers experience the joy that follows from the self-directed realization of skills, goals, and talents. Indeed, overcoming resistance and solving problems, some of them quite baffling, is central to the sense of accomplishment and pride that hackers routinely experience.

Hacker pleasure, however, is not always so staid and controlled; it far exceeds the pride of eudaemonia. Less frequently, but still occurring often, hackers experience a more obsessive and blissful state. Hacker descriptions of immersing themselves in technology remind me of Rainer Maria Rilke's terse and beautiful depiction of the passion that drives his intellectual pursuits: "All the soarings of my mind begin in my blood." This form of pleasure approximates what Roland Barthes (1975) has portrayed as bliss or jouissance—a pleasure so complete, engrossing, and enveloping that it has the capacity to obliterate every last shred of self-awareness. In native hack jargon, the state of bliss is the "Deep Hack Mode." Matt Welsh, a well-known hacker and computer scientist, humorously describes the utter magnetism of this mode, "very few phenomena can pull someone out of Deep Hack Mode, with two noted exceptions: being struck by lightning, or worse, your *computer* being struck by lightning."[10]

Because hackers often submit their will and being to technology—and are famous for denying their bodies sleep, at least for short periods—the joy that hackers derive from attending to and carefully sculpting technologies are at times experienced as transcendent bliss. In these moments, utility is exceeded. The self can at once express its most inner being and collapse within the objects of its creation. In the aftermath of a particularly pleasurable moment of hacking, there is no autonomous liberal self to be found.

To be sure, these forms of pleasure and engagement were impossible for me, the ethnographer, to touch and feel. But I routinely witnessed the social markers of the joy of hacking, as hackers talked shop with each other, as they joked about technical minutiae, and especially during their festive hacker celebrations. The key point is that the multifaceted pleasures of hacking signal that utility is not the only driving force in hackers' creative acts. Although hackers are fiercely pragmatic and utilitarian—technology after all must work, and work exceptionally well—they are also fiercely poetic and repeatedly affirm the artistic elements of their work. One of the clearest expressions of technology/software as art is when source code is written as poetry, or alternatively when poetry is written in source code (Black 2002). For many free software hackers, the act of writing software and learning from others far exceeds the simple enactment of an engineering ethic, or a technocratic calculus for the sake of becoming a more proficient as well as efficient programmer or system administrator.

This is hacking in its more romantic incarnation—a set of characterizations and impulses that hold an affinity with liberalism, and yet also stray into different, largely aesthetic and emotional territory. Liberalism, as a body of thought, certainly allows for pleasure, but for the most part does not theorize the subjective and aesthetic states of pleasure, which the Romantic tradition has centralized and made its own. Romanticism, explains

Rosenblum (1987, 10), is a "lavish departure from sober individualism," but also "amounts to an exploitation of liberal ideals." Although it is important to differentiate liberal from romantic sensibilities, they nonetheless can coexist without much friction, as Rosenblum contends in her account on Romanticism. She draws on various prominent historical figures, such as John Stuart Mill and Henry David Thoreau, to examine the compatibilities and symbiosis between liberalism and Romanticism. Hackers, borrowing from free speech commitments *and* also committed to aesthetic experiences, are a social group whose sensibilities lie at the interface between a more rational liberal calculus and a more aesthetic, inward-looking one.

Hackers are not alone in embracing this aesthetic, expressive sensibility, which philosopher Charles Taylor (1992) argues persuasively is a fundamental part of our contemporary imaginary, or what he calls the "expressive self." First visibly emerging in the eighteenth century, this sentiment formed the basis for "a new fuller individualism," and places tremendous weight on originality, sentiments, creativity, and at times, even disengagement. What must be noted is that expressive individualism and the moral commitments it most closely entails—self-fulfillment, self-discovery, and self-improvement—can be secured, as many critics have shown, through consumption, self-help, human enhancement technologies, and body modification (Bellah et al. 1985; Elliott 2003; Hogle 2005), and thus can converge seamlessly with elements of possessive individualism. Today to liberate and express the "authentic," "expressive" self is usually synonymous with a lifelong engagement with consumption, fine tuned by a vast advertising apparatus that helps sustain the desire for a seemingly limitless number of consumer goods and, increasingly, human enhancement technologies such as plastic surgery.

The example set by free software (and a host of similar craftlike practices), however, should make us at least skeptical of the extent to which an ethic of consumption has colonized expressive individualism. Free software hackers undoubtedly affirm an expressive self rooted not in consumption but rather in production in a double sense: they produce software, and through this technical production, they also sustain informal social relations and even have built institutions. Given the different ethical implications entailed in these visions of fulfillment, expression, and self-development (consumerist versus productive self), it behooves us to analytically pry them apart.

While the liberal articulations made by free software hackers, notably those of free speech, carry a familiar political imprint, their material experiences, the frustrations and pleasures of hacking, (including the particularities of making, breaking, and improving software) might seem politically irrelevant. Yet the passionate commitment to hacking and especially the ethics of access enshrined in free software licensing, express as well as celebrate unalienated, autonomous labor, which also broadcasts a powerful political message. A number of theorists (Galloway 2004; Söderberg 2007; Wark 2004) have previously highlighted this phenomenon. Hackers insistence on never losing access to the fruits of their labor—and indeed actively seeking to share these fruits with others—calls into being Karl Marx's famous critique of estranged labor: "The external character of labour for the worker appears in the fact that it is not his own, but someone else's, that it does not belong to him, that in it he belongs, not to himself, but to another" (Marx and Engels 1978, 74). It evokes Marx's vision precisely because free software developers seek to avoid the forms of estrangement that have long been nearly synonymous with capitalist production. Freedom is thus not only based on the right to speak free of barriers but also conceived as (although primarily through practice) "the utopian promise of unalienated labor, of human flourishing through creative and self-actualizing production," as Barton Beebe (2010, 885) aptly describes it.

F/OSS hacker morality is therefore syncretic—a quality that is also patently evident in its politics. It enunciates a liberal politics of free speech and liberty that speaks to an audience beyond hackers as well as a nonliberal politics of cultural pleasure and political detachment, which is internally and intensely focused on the practice of hacking only and entirely for its own sake, although certainly inspiring others to follow in their footsteps. When assessing the liberal ethics and affective pleasure of hacking, we should not treat pleasure as the authentic face of hacking, and the other (liberalism) as an ideological veneer simply in need of debunking (or in need of celebrating). From an ethnographic vantage point, it is important to recognize many hackers are citizens of liberal democracies, and have drawn on the types of accessible liberal tropes—notably free speech—as a means to conceptualize their technical practice and secure novel political claims. And in the process, they have built institutions and sustain norms through which they internalize these liberal ideals as meaningful, all the while clearly upholding a marked commitment to unalienated labor.

On Representing Hacker Ethics

If I was comforted by the fact that hacking could be analyzed in light of cultural issues like humor, liberalism, and pleasure, and that I had some methodological tools at my disposal to do so, as I learned more about hacking, my ease vanished as I confronted a new set of concerns. I increasingly grew wary of how I would convey to others the dynamic vitality and diversity that marks hackers and hacking, but also the points of contention among them. To further illustrate this point, allow me to share a brief story.

Soon after ending my official fieldwork, I was having dinner in Chicago with three local free software developers. One of them asked me about some of my memorable fieldwork experiences. There were many stories I could have chosen, but I started to tell the story of a speech by Kevin Mitnick—a more transgressive hacker (for he had engaged in illegal behavior) than most free software developers and one of the most infamous of all time—that I heard during summer 2004 at Hackers on Planet Earth (HOPE)—a conference founded in 1994 to publicize his legal ordeals. Mitnick is known to have once been a master "social engineer," or one who distills the aesthetics of illicit acts into the human art of the short cons. Instead of piercing through a technological barricade, social engineers target humans, duping them in their insatiable search for secret information. Because of various legendary (and at times, illegal) computer break-ins, often facilitated by his social engineering skills, Mitnick spent a good number of his adult years either running from the law or behind bars, although he never profited from his hacks, nor destroyed any property (Coleman and Golub 2008; Mitnick 2011; Thomas 2003).

In July 2004, free at last and allowed to use computers again, Mitnick attended HOPE in New York City for the first time. He delivered his humorous and enticing keynote address to an overflowing crowd of hackers, who listened, enraptured, to the man who had commanded their political attention for over a decade as part of a "Free Kevin Campaign." He offered tale after tale about his clever pranks of hacking from childhood on: "I think I was born as a hacker because at ten I was fascinated with magic," he explained. "I wanted a bite of the forbidden fruit." Even as a kid, his victims were a diverse lot: his homeroom teacher, the phone company, and even the Los Angeles Rapid Transit District. After he bought the same device used by bus drivers for punching transfers, he adopted the persona of Robin Hood, spending hours riding the entire bus network, punching his own pirated transfers to give to customers. He found transfer stubs while dumpster diving, another time-honored hacker practice for finding information that was especially popular before the advent of paper shredding. Despite the way that lawyers and journalists had used Mitnick's case to give hackers a bad name, Mitnick clearly still used the term with pride.

When I finished my story describing what I personally thought was a pretty engrossing speech, one hacker, who obviously disapproved of my reference to Mitnick as a "hacker," replied, "Kevin is *not* a hacker. He is a cracker." In the mid-1980s, some hackers created the term cracker to deflect the negative images of them that began appearing in the media at that time. According to *The Hacker Jargon File*, crackers are those who hack for devious, malicious, or illegal ends, while hackers are simply technology enthusiasts. Although some hackers make the distinction between crackers and hackers, others also question the division. To take one example, during an interview, one free software hacker described this labeling as "a whitewashing of what kind of people are involved in hacking. [. . .] Very often the same techniques that are used in hacking 2 [the more illegal kind] are an important part of hacking 1."

To be sure, hackers can be grasped by their similarities. They tend to value a set of liberal principles: freedom, privacy, and access. Hackers also tend to adore computers—the glue that binds them together—and are trained in specialized and esoteric technical arts, primarily programming, system, or Net administration, security research, and hardware hacking. Some gain unauthorized access to technologies, though the degree of illegality varies greatly (and much of hacking is legal). Foremost, hacking, in its different forms and dimensions, embodies an aesthetic where craft and craftiness tightly converge. Hackers thus tend to value playfulness, pranking, and cleverness, and will frequently perform their wit through source code, humor, or both: humorous code.

Hackers, however, evince considerable diversity and are notoriously sectarian, constantly debating the meaning of the words hack, hacker, and hacking. Yet almost *all* academic and journalistic work on hackers commonly whitewashes these differences, and defines all hackers as sharing a singular "hacker ethic." Offering the first definition in *Hackers: Heroes of the Computer Revolution*, journalist Steven Levy (1984, 39) discovered among a couple of generations of MIT hackers a unique as well as "daring symbiosis between man and machine," where hackers placed the desire to tinker, learn, and create technical beauty above all other goals. The hacker ethic is shorthand for a list of tenets, and it includes a mix of aesthetic and pragmatic imperatives: a commitment to information freedom, a mistrust of authority, a heightened dedication to meritocracy, and the firm belief that computers can be the basis for beauty and a better world (ibid., 39–46).

In many respects, the fact that academics, journalists, and hackers alike refer to the existence of this ethic is a testament not only to the superb account that Levy offers—it is still one of the finest works on hacking—but also to the fact that the hacker ethic in the most general sense is an apt way to describe some contemporary ethics and aesthetics of hacking. For example, many of the principles motivating free software philosophy reinstantiate, refine, extend, and clarify many of those original precepts. Further, and rarely acknowledged, Levy's account helped set into motion a heightened form of reflexivity among hackers. Many hackers refer to their culture and ethics. It is an instance of what Marshall Sahlins (2000, 197; see also Carneiro da Cunha 2009) describes as "contemporary culturalism"—a form of "cultural self-awareness" that renders culture into an "objectified value." This political dynamic of self-directed cultural representation is suggested in the following quote by Seth Schoen, an avid free software advocate and staff technologist at the Electronic Frontier Foundation. In the first line of text that appears on his Web page, Schoen announces, with pride: "I read [Levy's *Hackers*] as a teenager. [. . .] I was like, 'God damn it, I should be here!' Then, about ten years later, I thought back about it: 'You know, if there was a fourth section in that book, maybe I would be in there!' That's a nice thought."[11]

As I delved deeper into the cultural politics of hacking, though, I began to see serious limitations in making any straightforward connections between the hacker ethic of the past and the free software of the present (much less other hacker practices). Most obviously, to do so is to overlook how ethical precepts take actual form and, more crucially, how they transform over time. For example, in the early 1980s, "the precepts of this revolutionary Hacker Ethic," Levy (1984, 39; emphasis added) observes, "were not so much debated and discussed as silently agreed upon. *No Manifestos were issued*." Yet (and somewhat ironically) a mere year after the publication of his book, MIT programmer Richard Stallman charted the Free Software Foundation (FSF) ([1996] 2010) and issued "The GNU Manifesto," insisting "that the golden rule requires that if I like a program I must share it with other people who like it."[12] Today, hacker manifestos are commonplace. If hackers did not discuss the intricacies of ethical questions when Levy first studied them, over the span of two decades they would come to argue about ethics, and sometimes as heatedly as they argue over technology. And now many hackers recognize ethical precepts as one important engine driving their productive practices—a central theme to be explored in this book.

Additionally, and as the Mitnick example provided above illustrates so well, the story of the hacker ethic works to elide the tensions that exist among hackers as well as the different genealogies of hacking. Although hacker ethical principles may have a common core—one might even say a general ethos—ethnographic inquiry soon demonstrates that similar to any cultural sphere, we can easily identify great variance, ambiguity, and even serious points of contention.

Therefore, once we confront hacking in anthropological and historical terms, some similarities melt into a sea of differences. Some of these distinctions are subtle, while others are profound enough to warrant what I, along with Alex Golub, have elsewhere called genres of hacking (Coleman and Golub 2008). F/OSS hackers, say, tend to uphold political structures of transparency when collaborating. In contrast, the hacker underground, a more subversive variant of hacking, is more opaque in its modes of social organization (Thomas 2003). Indeed, these hackers have made secrecy and spectacle into something of a high art form (Coleman 2012b). Some hackers run vibrant technological collectives whose names—Riseup and Mayfirst—unabashedly broadcast that their technical crusade is to make this world a better one (Milberry 2009). Other hackers—for example, many "infosec" (information security) hackers—are first and foremost committed to security, and tend to steer clear of defining their actions in such overtly political terms—even if hacking usually tends to creep into political territory. Among those in the infosec community there are differences of opinion as to whether one should release a security vulnerability (often called full disclosure) or just announce its existence without revealing details (referred to as antidisclosure). A smaller, more extreme movement that goes by the name of antisec is vehemently against any disclosure, claiming, for instance, in one manifesto that it is their "goal that, through mayhem and the destruction of all exploitive and detrimental communities, companies, and individuals, full-disclosure will be abandoned and the security industry will be forced to reform."[13] There is also an important, though currently untold, story about gaming and hacking, not only because hackers created some of the first computer games, notably Space Wars, written in 1962, but because of the formal similarities between gaming and hacking as well (Dibbell 2006).

National and regional differences make their mark as well. For instance, southern European hackers have followed a more leftist, anarchist tradition than their northern European counterparts. Chinese hackers are quite nationalistic in their aims and aspirations (Henderson 2007), in contrast to those in North America, Latin America, and Europe, whose antiauthoritarian stance makes many—though certainly not all—wary of joining government endeavors.

Finally, while the brilliance of Levy's account lies in his ability to demonstrate how ethical precepts fundamentally inhere in hacker technical practice, it is important to recognize that hacker ethics, past and present, are not entirely of their own making. Just a quick gloss of the language many hackers frequently invoke to describe themselves or formulate ethical claims—freedom, free speech, privacy, the individual, and meritocracy—reveals that many of them unmistakably express liberal visions and romantic sensibilities: "We believe in freedom of speech, the right to explore and learn by doing," explains one hacker editorial, "and the tremendous power of the individual."[14] Once we recognize the intimate connection between hacker ethics and liberal commitments *and* the diversity of ethical positions, it is clear that hackers provide less of a unitary and distinguishable ethical position, and more of a mosaic of interconnected, but at times divergent, ethical principles.

Given this diversity, to which I can only briefly allude here, the hacker ethic should not be treated as a singular code formulated by some homogeneous group called hackers but instead as a composite of distinct yet connected moral genres. Along with a common set of moral referents, what hacker genres undoubtedly share is a certain relation to legality. Hacker actions or their artifacts are usually either in legally dubious waters or at the cusp of new legal meaning. Hence, they make *visible* emerging or contentious dilemmas.

Although hackers certainly share a set of technical and ethical commitments, and are in fact tied together by virtue of their heated debates over their differences, given the existence of the diversity just noted, my claims and arguments should not be taken as representative of all hacking, even though for the sake of simplicity (and stylistic purposes), in the chapters that follow I will often just refer to hackers and hacking. My discussion is more modest and narrow for it will stick primarily to the example of free software.[15] My preference for announcing the "self-conscious, serious partiality" (Clifford 1986, 7) of this account comes from witnessing motivations, ethical perceptions, desires, and practices far more plastic, flexible, sublime, contradictory, and especially fiery and feverish than usually accounted for in academic theories. The world of hacking, as is the case with many cultural worlds, is one of reckless blossoming, or in the words of Rilke: "Everything is blooming most recklessly; if it were voices instead of colors, there would be an unbelievable shrieking into the heart of the night."

Omissions and Chapter Overview

Some readers may be asking why I have not addressed Silicon Valley entrepreneurship and Web 2.0, both of which might further illuminate the ethics and politics of F/OSS.[16] For those interested in Web 2.0—a term that is bandied around to refer to nearly all contemporary digital tools and the social practices that cluster around these technologies—you might want to jump to the short epilogue, where I critique this term. It is a moniker that obscures far more than it reveals, for it includes such a wide range of disparate phenomena, from corporate platforms like Flickr, to free software projects, to dozens of other digital phenomena. In fact, by exploring in detail free software's sociocultural dynamics, I hope this book will make it more difficult to group free software in with other digital formations such as YouTube, as the media, pundits, and some academics regularly do under the banner of Web 2.0.

The relationship between Silicon Valley and open source is substantial as well as complicated. Without a doubt, when it comes to computers, hackers, and F/OSS, this high-tech region matters, as I quickly came to learn within weeks of my arrival there. For the last thirty years, hackers have flocked to the Bay Area from around the world to make it one of their most cherished homelands, although it certainly is not the only region where hackers have settled and set deep roots. At the turn of this century, open source also became the object of Silicon Valley entrepreneurial energy, funding, and hype, even though today the fever for open source has diminished significantly, redirected toward other social media platforms.

The book is thus not primarily about free software in Silicon Valley. In many respects my material tilts toward the North American and European region but, nevertheless, I have chosen to treat free software in more general than regional registers as well, so as to capture the reality of the legal transnational processes under investigation along with the experience of the thousands and thousands of developers across the world. Debian, for example, has developers from Japan, Australia, Canada, New Zealand, all over western and eastern Europe, Brazil, Venezuela, Argentina, and Mexico.[17] I decided on this approach as it is important to demonstrate different values and dynamics at play than those found in Silicon Valley, which are too often mistaken to represent *the* commitments of all engineers, computer scientists, and hackers.[18]

Coding Freedom is composed of six chapters, divided conceptually into pairs of two. The first two chapters are historically informed, providing the reader with a more general view of free software. Chapter 1 ("The Life of a Free Software Hacker") provides what is a fairly typical life history of a F/OSS hacker from early childhood to the moment of discovering the "gems" of free software: source code. Compiled from over seventy life histories, I demonstrate how hackers interact and collaborate through virtual technologies, how they formulate liberal discourses through virtual interactions, how they came to learn about free software, and how they individually and collectively experience the pleasures of hacking. I also offer an extended discussion of the hacker conference, which I argue is the ritual (and pleasurable) underside of discursive publics. Chapter 2 ("A Tale of Two Legal Regimes") presents what were initially two semi-independent legal regimes that over the last decade have become intertwined. The first story pertains to free software's maturity into a global movement, and the second turns to the globalization and so-called harmonization of intellectual property provisions administered through global institutional bodies like the World Trade Organization. By showing how these trajectories interwove, I emphasize various unexpected and ironic outcomes as I start to elaborate a single development that will continue to receive considerable treatment later in the book: the cultivation, among hackers, of a well-developed legal consciousness.

The next two chapters provide a close ethnographic analysis of free software production. Chapter 3 ("The Craft and Craftiness of Hacking") presents the central motif of value held by hackers by examining the practices of programming, joking, and norms of socialization through which they produce software and their hacker selves. Partly by way of humor, I tackle a series of social tensions that mark hacker interactions: individualism and collectivism, populism and elitism, hierarchy and equality as well as artistry and utility. These tensions are reflected but also partially attenuated through the expression of wit, especially jokes, and even funny code, whereby jokes ("easter eggs") are included in source code. Chapter 4 ("Two Ethical Moments in Debian") addresses ethical cultivation as it unfolds in the largest free software project in the world—Debian. This project is composed of over one thousand developers who produce a distribution of the Linux operating system (OS). I present and theorize on the tensions between Debian's governance, which blends democratic majoritarian rule, a guildlike meritocracy, and ad hoc deliberations. In comparing these three modes of governance, I unearth various ethical processes—informal, formal, pedagogical, and dramatic—by which Debian developers inhabit a liberally based philosophy of free software, and use it as an opportunity to revisit the tension between liberal individualism and corporate sociality explored earlier.

The final two chapters engage with more overtly political questions, examining two different and contrasting political elements of free software. Chapter 5 ("Code Is Speech") addresses two different types of legal pedagogy common among free software developers. First, in the context of Debian, I look at everyday legal learning, where debating and learning about the law is an integral part of project life. I then compare this with a series of dramatic arrests, lawsuits, and political protests that unfolded between 1999–2004 in the United States, Europe, and Russia, and on the Internet, and that allowed for a more explicit set of connections to be drawn between code and speech. These demonstrations were launched against what was, at the time, a relatively new copyright statute, the DMCA, and the arrest of two programmers. These multiyear protests worked, I argue, to stabilize a relatively nascent cultural claim—nearly nonexistent before the early 1990s—that source code should be protected speech under the First Amendment (or among non-American developers, protected under free speech laws). In contrast to the political avowal of the DMCA protests, my conclusion ("The Politics of Disavowal and the Cultural Critique of Intellectual Property Law") discusses how and why hackers disavow engagement in broad-based politics, and instead formulate a narrow politics of software freedom. Because a commitment to the F/OSS principles is what primarily binds hackers together, and because many developers so actively disavow political associations that go beyond software freedom, I contend that the technoscientific project of F/OSS has been able to escape the various ideological polarizations (such as liberal versus conservative) so common in our current political climate. F/OSS has thus been taken up by a wide array of differently positioned actors and been placed in a position of significant social legibility whereby it can publicly perform its critique of intellectual property law.

Finally, to end this introduction, it is worth noting that this book is not only an ethnography but also already an archive of sorts. All cultural formations and ethical commitments are, of course, in motion, undergoing transformation, and yet many technological worlds, such as free software, undergo relentless change. What is written in the forthcoming pages will provide a discrete snapshot of F/OSS largely between 1998 and 2005. Much of this book will still ring true at the time of its publication, while other elements have come and gone, surely to have left a trace or set of influences, but no longer in full force. And despite my inability to provide a warranty for this archival ethnography, I hope such an account will be useful in some way.