

## **Viral Style: Technology, Culture, and the Politics of Infection**

Douglas Thomas  
Annenberg School for Communication  
University of Southern California  
Los Angeles, CA 90089-0281  
(213) 740-3937

[douglast@usc.edu](mailto:douglast@usc.edu)

## Viral Style: Technology, Culture, and the Politics of Infection

### Overview

This book explores the phenomenon of computer viruses through an examination of the history of computer viruses in computer culture and industry, in the “computer underground,” and in the popular imagination. These three, seemingly distinct, ways of thinking inflect on each other in interesting and often surprising ways.

Computer viruses are perhaps one of the best known, yet least understood elements of the “computer underground.” While most people are acquainted with the ideas and effects of computer viruses, few understand the social, cultural, and political implications from either the point of view of virus writers or from the impact that narratives of viral infection have had on the popular imagination. From AIDS and Ebola to the Michelangelo computer virus, the discourse of viral infection has permeated the social consciousness for the past two decades. What has been remarkable is the manner in which the notion of a virus, as a purely biological phenomenon, has undergone a technological transformation, becoming instituted in the popular vocabulary and imagination as a something that is able to infect technology. The result has not only to give life to the virus itself, but has, in the process, rendered the technology of the computer organic as well.

This work combines three approaches. The first is an effort to historicize the notion of viral infection in terms of its relation to the moment of technological transformation when it became possible to speak of a “computer virus.” The second approach is focused on media and cultural criticism in an effort to trace out the cultural, social, and political implications of what viral infection has connoted in the last half of the 20<sup>th</sup> century. Third, and perhaps most interestingly, this work approaches the culture of underground “virus writers” ethnographically to understand the cultural, social, and political motives of this groups of programmers who actively constitute themselves as a subculture of technology that is both driven by these narratives of viral infection, but also constitutive of them.

As these stories unfold, what is made clear is that the notion of “viral style,” the means by which these programmers constitute themselves as a subculture, has a great deal to tell us about popular anxieties over technology and infection. The story of these virus programmers’ culture cannot be divorced from either the history that has produced it or from the popular representations of viral infection that often serves as its inspiration and site of reception.

I bring several unique advantages to the study of the computer underground. I have just completed a similar study, *Hacking Culture* (University of Minnesota Press), an ethnography and examination of the underground world of computer hackers and in doing so have had the opportunity to speak with a number of hackers who also participate in virus culture. Because of my former experiences participating in hacker subculture and discussing these matter with virus programmers, I have both a technical and cultural literacy required to understand the ways in which virus programmers address issues of representation, performance, and identity both as expressed in their programming and in response to cultural and media representations of them. In my research, I have spoken or corresponded with a number of virus programmers who figure

prominently in both the underground culture and in my analysis of it. Over the past year I have begun to engage in ethnographic research, spending time with virus writers on-line and through a series of private correspondences. Virus programmers, by nature, are suspicious and distrustful of academic research. Having a connection to and a familiarity with the culture of the underground allows me a kind of access that is usually difficult to attain.

As a result, I am in a position uniquely to inform my audience about the ways in which this culture functions and provide insight into a subculture that is, itself, cautious and secretive about its own activities and behaviors.

### **Competing Works and Target Audiences**

No other work to date has taken up the task of theorizing the culture of virus production in the computer underground. Although there are a range of books which have dealt with the social and political dimensions of infection, particularly around AIDS, this book differs markedly from those approaches based on its intent to interrogate the subculture of technological virus production. While those works are tremendously important in thinking through the manner in which the concept of infection has permeated American culture, they serve more as a parallel discourse upon which virus writers have capitalized upon, rather than a discourse inherent to the subculture itself.

Peter Denning's *Computers Under Attack : Intruders, Worms, and Viruses* (1990) is perhaps the best example of the typical computer-related work on viruses. In it, he documents and details the component parts of computer viruses and discusses strategies for system protection. On the other side, George Smith's *The Virus Creation Labs : A Journey into the Underground* (1994) provides an interesting narrative history of both virus and anti-virus programmers. The account, however, is primarily journalistic in nature and theorizes neither the underground culture itself nor the broader implications for mainstream culture. Other works, such as Mungo and Clough's *Approaching Zero: The Extraordinary World of Hackers, Phreakers, Virus Writers, and Keyboard Criminals* (1992), or Charles Platt's *Anarchy Online: Net Crime* (1996), do address viruses, but also do so from a journalistic perspective, detailing how and when viruses were created, but do little to explain the social and political motivations of virus writers themselves. In addition, there is no work to date which historicizes the culture of virus production or addresses the phenomenon in terms of cultural anxieties or attitudes toward it.

This book will be of interest to a number of different audiences, both academic and popular. Most directly and most obviously, as a cultural study of technology, it will appeal to those interested in "Cyberculture Studies" and those interested in issues of the cultural reception of technology or subculture studies. Second, the book also inflects on questions of cultural and popular representation through film, TV, and newspapers and, accordingly, will be interest to those in media studies, popular culture, cinema/TV, and communication and journalism scholars. Third, because the book discusses in detail the relationship between computer crime, the law, and the legal status of viruses and virus writers it will hold some appeal for those interested in the questions of identity, particularly in relation to the juridical constitution of the subject. Finally,

there will be a popular appeal for this work, given the importance of technology in everyday life. Newspapers such as the *LA Times* and the *New York Times* frequently report on viruses and periodicals including *Time* and *Newsweek* have each reported regularly on computer viruses. As a work that provides insight into a phenomenon (computer viruses) which is often discussed, but poorly understood, this book should be of interest to anyone who spends time on-line, surfs the net, or is curious about the computer virus's place in the history and practice of technology.

## **Table of Contents**

### I. Historicizing Viruses

- Ch 1. The Nature of the Beast: The Anatomy of a Virus
- Ch 2. The History of Infection: From Shockwave Rider to Robert Morris
- Ch 3. Viruses in the Popular Imagination: from War of the Worlds to Independence Day
- Ch 4. From Biology to Technology: Technological Transformations in the Discourse of Infection

### II. The Culture of Virus Production

- Ch 5. The Culture of Virus Production
- Ch 6. Tools of the Trade: VCL and other Goodies
- Ch 7. Smashing the System: From Boot Sectors to Microsoft Word

### III. Viral Style: Technology, Culture and the Politics of Infection

- Ch 8. Computer Style and Incorporation
- Ch 9. The Industry and Its Discontents: From Microsoft to McAfee and Associates
- Ch 10. Viral Style: Polymorphism and Resistance

## **Manuscript Specifics**

I estimate the manuscript will be between 200-250 pages in length and will be completed in Fall of 1999. In addition to text, the manuscript will contain between 10 and 15 still photographs.

## **Outline of the Book**

### **I. Historicizing Viruses**

The computer virus is perhaps the most feared, yet least understood phenomenon of the information age. Cast in biological terms, computer viruses are seen as microscopic organisms too small to see, yet capable of wreaking untold damage. They also appear to be a recent phenomenon, the product of the widespread use and adoption of technology both in the workplace and in the home and, most specifically, tied to the growth and usage of the Internet. Most viruses, however, are not spread via the Internet, but instead come from the sharing of data over smaller networks (such as LANs) or, most commonly, from people sharing floppy disks and exchanging software. One is much more likely to get a virus from someone you know or work with than from a download off of the Internet.

This section offers a view of viruses which both explores the nature of computer viral infection by examining what viruses are and how they work and by historicizing their place in both computer and popular culture.

#### **The Nature of the Beast: The Anatomy of a Virus**

The question of what precisely defines a “virus” is the subject of debate within the computer community. There are, however, at least two characteristics that show up in most definitions. First, a virus must in some manner “self-replicate,” meaning that it must make a copy of itself which has the ability to also make a copy of itself. Accordingly, the quality of self-replication is the means by which the virus is also able to propagate. The second characteristic is that the virus must function without the user’s knowledge, meaning simply that it must propagate undetected and must perform its basic task or function of self-replication without the users knowledge or consent. Apart from those two characteristics, there are a number of qualities which may or may not define a virus, but in almost every case, absent from the list is the idea of a malicious intent.

While the popular understanding of viruses is that they are primarily to cause damage, virus writers see their own productions differently. Indeed, all virus writers seek first and foremost to have their viruses keep replicating. While some viruses do cause damage, referred to as “malware,” short for “malicious software,” some are completely benign. In fact, often times viruses are propagated that display greetings or messages on a particular date or simply send the message “Have a Nice Day,” when exiting a particular program or application. The idea behind damage and more benign messages is the same, they serve as a marker to indicate how successfully the virus has been in its mission to self-replicate. News of damage spreads more quickly than “Have A Nice Day,” and is therefore often times selected as the virus’s payload in the hopes the virus writer will be better able to view the impact of his or her programming.

In order to understand exactly what a virus is and how it functions, this chapter examines the basic workings of a personal computer and illustrates how virus is written, executed and the

processes by which viruses go about their mission of self-replication. In doing so, I trace out the basic ideas behind “bootstrapping” or “booting” the system, fundamental elements of the operating system, and how computers access and store information. With those basic understandings, I trace out how virus writers program code which takes advantage of the basic system architecture and how such code is able to hide and later replicate itself within the system. In each case, it is easy to see how a short (often times only a few dozen lines of code) program can be used to either send silly messages or wreak havoc inside a user’s computer and how these programs begin to take on a life of their own, jumping from machine to machine, constantly replicating themselves and attaching new code to programs.

### **The History of Infection: From Shockwave Rider to Robert Morris**

Understanding the basic structure and function of a virus provides some insight into the technical aspects of how viruses function. Of equal interest is the social and cultural dynamics that make these programs of interest. The choice of the term “virus” was not serendipitous, but instead was the product of intense and heated debates within the computer community. It was also a term that conveyed both the sense of danger and mystery to the public. Before there were viruses, however, there were “worms,” another biological entity that carried far less threatening overtones.

This chapter traces the debate over nomenclature which grew out of John Brunner’s 1975 novel *The Shockwave Rider* (often considered a manifesto for hackers and virus writers) and how the concept of both hacking and virus writing grew out of fears of technological dominance that Brunner suggests is inevitable. Brunner suggests that any system of technological dominance is subject to a basic strategy of resistance: using technology to defeat technology. Technological systems are inherently fragile and the more complicated they become, the more fragile they are. Brunner’s hero, who is capable of changing identities to hide from authorities, creates a program that, essentially, smashes the system, causing the government to lose its technological domination over the population. The power comes not from the ability to control or manipulate the system, but from the ability to interfere with and disrupt it. That power comes not from a specific assault, but from a decentralized attack, a *networked* attack.

Thirteen years later, on November 2, 1988, Robert Tappan Morris released what would become known as the “Internet Worm,” a program which tested systems of the Internet for a known security hole, accessed the system, and tried to break passwords by comparing them to a list of commonly used words. It was the actualization of Brunner’s “tapeworm,” which spun out of control, crashing over 6,000 systems. Although the damage was unintentional, Morris’s worm illustrated precisely what was at stake--the ability to crash the system. Computer programmers and experts debated over the proper designation for Morris’s program, was it a “worm” or was it a “virus?” Eventually, the designation “worm” won out, but the discussion opened up a new area of examination, one which the media and the public would become aware of in the years that followed as viruses began to spread.

The transformation from “worm” to “virus” marks an important element for the study of virus culture for two reasons: first, it marks the transformation from a useful, and most important, identifiable creature to a sinister, invisible object that is tied to illness, disease, and death and second, it enacts the metaphor of infection which transforms computer virus infection from something which can be located and defined into a public health threat. That threat has produced a multi-million dollar business in the form of anti-virus software and updates.

### **Viruses in the Popular Imagination: from War of the Worlds to Independence Day**

The computer industry is not the only arena in which the notion of the virus has played an important role. In the popular imagination, infection has (at least) twice saved the human race. The first time, in H.G. Well’s *War of the Worlds*, invaders from Mars are thwarted in their efforts to take over Earth by infection with an Earth-bourne bacteria which their delicate Martian systems are unable to combat. In the 1997 updated version, *Independence Day*, the invading aliens meet a similar fate. This time, however, the virus has been changed. It is no longer a biological infection, but, instead, a technological one. The hero of *Independence Day* is in fact a computer virus which interferes with the alien’s computer system, forcing them to lower their defenses so that U.S. attack planes can destroy the invading ships.

The transformation of the biological into the technical marks an important point in the evolution of the computer virus. It suggests that the ability (and willingness) to “smash the system” is inherently tied to both human instinct and freedom, but it also suggests that in the late 1990s, human resilience has become tied to technology. Where the Martians in the *War of the Worlds* were defeated by a biological infection, aliens in *Independence Day* are proved vulnerable to technological infection.

In tracing out this technological transformation, I argue that what is revealed is how anxiety about the political domination has been transformed into fears of technological domination.

### **From Biology to Technology: Technological Transformations in the Discourse of Infection**

The discourse of infection is, of course, nothing new. What may be considered new, however, is the technological transformation that such infections have undergone. In the 1980s and 1990s, as the discourse of AIDS and Ebola figured prominently in the popular consciousness, there also arose a discourse of technological infection around the computer virus.

This chapter examines the parallel discourse of biological and technological viruses in an effort to understand how the discourse of technological infection relates to and mirrors the discourses of biological infection. In doing so, I also examine how the computer itself had to undergo a similar transformation, literally becoming a biological organism itself, capable of serving as a host and undergoing processing of both infection and healing as well as a parallel discourse of prevention and “safe computing.”

This chapter also examines the manner in which protection in the form of anti-virus software has been marketed in relation to discourses of other infections, particularly in terms of AIDS.

## II. Viral Style: Technology, Culture, and the Politics of Infection

### The Culture of Virus Production

Within the world of the computer underground there exists a subculture devoted to potentially the most lethal and problematic form of computer programming--the creation and dissemination of viruses. These programs, which range from code which prints humorous messages across the screen to programs which delete or destroy information, are produced and exist in relation to a clearly marked and defined subculture of virus writers. Unlike the image of virus writers as "high tech vandals," these programmers are often very talented and see virus writing as a social, cultural and political project. This essay traces the history and motives of this subculture in an effort to explore both the conditions under which that subculture was born and evolved as well as the shifting cultural and political contexts to which these programmers have responded. Accordingly, I argue that the creation of viruses is not merely a malicious act of vandalism or a senseless act of high-tech destruction, but, instead, functions as a means of subcultural signification and as a strategy for the preservation of a subcultural style in an age of increasing incorporation and commodification of underground computer culture.

As a subculture, virus writers present themselves as what Dick Hebdige identified as *noise*, "as an actual mechanism of semantic disorder: a kind of temporary blockage in the system of representation."<sup>1</sup> Subculture, Hebdige argues, manifests itself in *style* as an intentional form of communication whereby the cultural and social negotiation of signs takes place. Marked as a kind of deviant behavior, subcultural style is often times "incorporated" through the process of commodification which results in the "conversion of subcultural signs into mass-produced objects (i.e. the commodity form)."<sup>2</sup> The result is a "diffusion of the subculture's subversive power," (e.g. the way in which "punk innovations fed back directly into high fashion and mainstream fashion" in the 1970s).<sup>3</sup>

The birth and growth of virus culture can be traced directly to the commodification of the computer in the form of the PC and as a response to the incorporation and dilution of computer culture which accompanied the mass marketing of the PC. In terms of technology, the mass appeal of the personal computer, particularly in the mid-1980s, produced a widespread incorporation of computer culture, taking the essentially subversive "hacker style" which demanded an intimate knowledge of these machines and how they worked and stripping it of its transgressive character for mass consumption. With the introduction and growth of GUIs (graphical user interface), computer users have become increasingly distanced from the machines and software that they use. As a result, the technology has been rendered increasingly opaque, even as it has become easier to use and more "user friendly."



Virus writers are reacting with a kind of digital violence to these transformations which have taken place in computer culture. The dynamics of the production of a commodified opaque technology have created two motives for virus production. First, viruses force the end user to become aware (or at least more aware) of his or her blind reliance or dependence on technology. In doing so, the threat of viral infection forces him or her to take note of the technology itself. The threat of viral infection forces the end user to understand how his or her computer works, to take precautions, to be aware of how viruses are spread and how to protect oneself. As one writer commented about the threat arising from the Microsoft Word Macro viruses: “Control is in your hands. Don’t panic. Take this as an opportunity to learn more about the features of the software you use, to test and verify any security features you plan to utilize and then to configure accordingly.”<sup>4</sup> To the virus writer, the philosophy is simple--there are risks associated with ignorance, especially with ignorance about technology. Typically, virus writers are more hyperbolic in their assessments. As Dark Angel (one of the more vitriolic virus writers) sees it: “Virii are wondrous creations written for the sole purpose of spreading and destroying the systems of unsuspecting fools. This eliminates the systems of simpletons who can’t tell that there is a problem when a 100 byte file suddenly blossoms into a 1,000 byte file. Duh. These low-lives do not deserve to exist, so it is our sacred duty to wipe their hard drives off the face of the Earth. It is a simple matter of speeding along survival of the fittest.”<sup>5</sup> While many, even most, virus writers in the subculture do not harbor such malicious intent, few would disagree with the assessment of the typical computer user as an “unsuspecting fool.”

This split, between programmers and “end users,” opens up the space for the production not only of viruses but of virus culture itself. In this chapter, I examine how the incorporation of computer culture has fostered a new level of subcultural identity and new strategies of technological resistance.

### **Tools of the Trade: VCL and other Goodies**

If virus programmers learned anything from the incorporation of computer culture, it was how they might incorporate it back into their own subculture as well. One of the results of such re-incorporation has been a flood of GUI-based virus programs and “mutation engines.” This software is the point-and-click version of virus programming, which allows *anyone* to not only create viruses for distribution, but also to mutate viruses to make them increasingly difficult to detect and protect against.

These programs are born in an environment which still requires a level of access before they are made available. For example, the most famous program VCL (Virus Creation Laboratory) is itself password protected. Anyone who has spent a fair amount of time reading virus literature has come across the password a number of times. It does, however, prevent a neophyte access to the potentially destructive tool. Since VCL’s distribution there have been a series of similar programs, each designed to stay one step ahead of the anti-virus software on the market. Usually, the viruses these programs create are quickly neutralized by virus detection software, but for the underground they serve a second, more useful purpose. They provide a hands-on introduction to

virus programming from which programmers can learn and begin to experiment. They provide, in essence, tutorials and introductions to the world of virus programming.

Once introduced to the subculture, it becomes a relatively trivial task to find manuals on virus writing, periodicals devoted to the topic, and a wide range of informational resources that enable potential virus writers to learn how to code. It is also the site of discourse where virus writers express social, cultural, and political concerns. One such 'zine, called *40Hex* served for a number of years as both a political soapbox and virus distribution/tutorial forum for its creators, programmers who went by the name *SKISM* (Smart Kids Into Sick Methods). The remainder of the chapter examines the history and evolution of the virus underground through readings of key documents and discussions with participants.

### **Smashing the System: From Boot Sectors to Microsoft Word**

A second motive for virus writing grows out of the split that virus writers see between the expert programmer and the end user. Virus writers see the broader cultural implications of such dependence on technology as well. They see the commodification of the computer and of computer culture as leading to the possibility of technological domination and viruses provide a sense of protection. Accordingly, the targets for viral infection are not just individual users. Instead, the inspiration for such coding comes from precisely the kind of blockage that is characteristic of subcultural style. As one virus writer explains, the typical virus writer “is usually just this angry kid who happens to be very clever, and what’s going through his head when he codes this thing is how fuckin’ cool it’ll be when it starts blowing holes through the infrastructure of some industrial monolith like IBM, where a bunch of drones will start going bugfuck when everything stops working.”<sup>6</sup> The idea of being able to “smash the system” has its roots as far back as John Brunner’s sci-fi classic *The Shockwave Rider*, which tells the story of a computer programmer who liberates society from the tyranny of technology by releasing a virus-like program throughout the government’s network. Like Brunner’s hero, virus writers see themselves as maintaining the fragility of the system, keeping it in a state of precariousness such that its power over us is always limited by our ability to destroy it.

Virus writers have learned an important lesson from the past. Watching the process of commodification, whereby computer style has been commodified and stripped of its potentially subversive force, virus writers have adopted a different sense of style which has made them more resistant to cultural incorporation. Accordingly, viral style represents a break from Hebdige’s initial notion of subculture in an important respect. The culture surrounding viruses is a subculture which demands constant innovation and which accounts for mainstream culture’s (and the industry’s) ability to commodify and incorporate aspects of subversive style. Viral style is a response to a response, a *mutation* of style which is negotiating precisely the moment of incorporation of earlier computer style and culture by the mainstream.

With the mass-production of the personal computer and the widespread incorporation of computer culture, viral style emerged as a style that negotiates a *previously incorporated style*. Cognizant of the dangers and possibilities of incorporation itself, viral style is *self-replicating* and *polymorphic*, continually changing with each iteration. It is a style that enacts its own defense mechanism against incorporation.

This chapter traces out the cultural history and methods of “smashing the system,” beginning with the first “boot sector” viruses and concluding with the innovation of Microsoft Word’s “automacro” feature, which has spawned a new method of virus delivery, previously thought impossible.

### **III. Viral Style: Technology, Culture and the Politics of Infection**

In addition to issues of popular culture and the computer underground, virus programming intersects with specific legal issues. One such issue is the ontological status of the virus itself, another is the issue of attaching responsibility to issues of virus production and virus dissemination, and yet another touches upon issues of the separation of textuality (the code) from the “entity” (the virus itself). Each of these issues raises questions that complicate conventional notions of law, policy, and textuality.

#### **Legal Issues in Virus Coding**

When Robert Morris unleashed his Internet Worm in 1988, the concept of a computer virus was not part of public discussion. Since that time, the public discussion of computer viruses has occurred at the lowest levels, often promoting fear and misunderstanding of what computer viruses are and what can be done to prevent them. Until very recently, cases of virus programming have not intersected with law enforcement. In fact, virus production, which is both an international phenomenon and an activity that is nearly impossible to attach to a single agent, clearly marks a challenge to a number of legal ideals.

There are a number of ways in which virus code can be disseminated. First, and most obviously, the code can be compiled and attached to a program which is then disseminated over a network or through floppy disks. Such a method, if able to be proved, would be subject to laws that already protect property (e.g., trespass, vandalism, and other damage to property). In most jurisdictions knowingly distributing a virus is a criminal offense. The problem arises, however, with the second, more common means of distribution, the computer code itself. Often, virus programmers will distribute the code in an uncompiled form. In doing so, they are not distributing a virus and are not infecting any files. That code, as long as it is marked as virus code and not intended to infect a user’s computer without their knowledge is then protected under the First Amendment of the Constitution. It has status as text, rather than as a virus.

Accordingly, virus code, harmless in its textual form, may have a single point of origin, but its dissemination and eventual transformation into viral form is always multiple, decentralized, and

essentially divorced from its origin. Because the process of compilation of the virus is a transformative process, it fundamentally changes the nature of the text and problematizes the notion of authorship.

### **The Industry and Its Discontents: From Microsoft to McAfee and Associates**

While the legal status of virus programming is ambivalent, the industry response has been much more direct. In the wake of the Robert Morris Internet worm and the growth and spread of computer viruses more generally, Silicon Valley has responded with the introduction of anti-virus software, creating a multi-million dollar business almost instantly. But the relationship between virus writers and anti-virus programmers is a curious dance. Rather than reading anti-virus software as a response to virus programmers, it is important to understand the symbiotic relationship that has developed among them. Virus writers devote their time, almost exclusively, to finding better ways to trick, hide from, or otherwise subvert anti-virus software, which, in turn, promotes the need for continual upgrades, modifications and new software from the anti-virus establishment. Most virus programmers will insist that the competition with anti-virus software writers has functioned to make them *better* virus writers. But it is the Silicon Valley anti-virus software programmers who reap the greatest benefit, particularly financial benefits, from this improved code.

In that sense, the relationship between virus writers and anti-virus programmers is one which allows us to explore what might be considered the “limits of capitalism” in an information culture and forces us to rethink basic notions of commodification and incorporation of subcultural style. Where traditionally such subversion would be seen as a threat to the market, the computer industry has been able to transform that threat into a marketable commodity, so much so that the elimination of the threat itself is more of a problem than the threat itself. The anti-virus industry does not need to transform or incorporate virus subculture. Instead, it relies on virus subculture being *precisely* what it is. As a reaction of subculture, we are witnessing a new form of incorporation which demands that the subculture resist, and in doing so a component of mainstream culture begins to depend on that resistance as a means to create and define itself.

### **Viral Style: Polymorphism and Resistance and the Transformation of Capital**

The book concludes with an assessment of what precisely virus culture and mainstream culture’s response to it tells us about living in an information culture and how notions of subculture and resistance might meaningfully be rethought in the age of information. The creation, production, and dissemination of virus code can be seen as a metaphor for larger issues that are shaping our world and our perceptions. In the discourse of the virus, we witness the fusion of textuality and biology, of machine and the organism, of public anxiety and industry marketing of it, and of the breakdown and reformulation of subcultural incorporation.

In such a series of transformations, it is clear that the transformation from a primarily material culture to a culture of information is going to force us to rethink some of the most central tenets

of contemporary thought. How we consider textuality, authorship, and the construction of the subject will have profound impacts for us, not only in questions of law, but also in the radical transformation of capitalism that will accompany both the shift in mainstream culture and which will underwrite the market for a new economy of information and the anxiety that will doubtlessly accompany it.

Accompanying those transformations will be a new style of subculture, one born from resistance to a new style of capitalism. It will be polymorphic and highly resistant and will, one can hope, serve as a model from which the next generation can learn, adapt, and continue to transform the very possibilities of what it means to live in an information culture and adopt a “viral style.”

### Notes

1. Dick Hebdige, *Subculture: The Meaning of Style*, New York: Routledge, 1979, p. 90.
2. Hebdige, p. 94.
3. Hebdige, p. 95.
4. In Charles Platt, *Anarchy Online: Net Crime*, New York: Harper Collins, 1996, p. 149.
5. Dark Angel, "Dark Angel's Phunky Virus Writing Guide." Text file. n.d. (c. 1992).
6. Platt, p. 145.