IDEOLOGY AS CULTURAL SOFTWARE

*J.M. Balkin**

This essay concerns one of the apparent casualties of the rise of postmodernism—the theory of ideology. In fact, the "theory of ideology" has never been a unitary theory. From its inception the concept of ideology has always been contested, and hence the theory has generated many variations.[1] The most common version asserts that individuals are afflicted with beliefs that in some way mystify or obscure social reality. This mystification in turn works to the advantage of some classes and to the disadvantage of others. This formulation was developed most prominently within the marxist tradition; it has proved a tempting but ultimately unsatisfactory paradigm to explain all sorts of ideological phenomena that bear little resemblance to marxist class struggle.[2]

In recent years postmodern theorists have subjected the marxist theory of ideology to considerable criticism.[3] Some scholars, perhaps inspired in part by Michel Foucault, have abandoned the study of ideology altogether for the study of discourse, hoping that this substitution will permit a better understanding of the many social injustices—like racism, sexism, and homophobia—that transcend the boundaries of economic class.[4] Nevertheless, replacing the study of ideology with the study of discourse does not alter the basic questions that gave rise to the theory of ideology in the first place, nor does it obviate the most troubling problems of this theory.

One reason for this is that the theory of ideology, like the study of discourse associated with postmodernism, has always been part of a larger endeavor. This more general endeavor is the philosophy of culture. The ancient Greek world distinguished be-

---

* Lafayette S. Foster Professor of Law, Yale University. My thanks to Sanford Levinson and Tom Seung for their comments on a previous draft.

1 Compare the variety of definitions offered in TERRY EAGLETON, IDEOLOGY: AN INTRODUCTION 1-2 (1991).

2 *See, e.g.*, CATHARINE MACKINNON, TOWARDS A FEMINIST THEORY OF THE STATE (1989).

3 *See, e.g.*, MICHEL FOUCAULT, POWER/KNOWLEDGE: SELECTED INTERVIEWS AND OTHER WRITINGS 1972-1977, at 118-19 (Colin Gordon ed. & Colin Gordon et. al. trans., 1980).

4 *See, e.g.*, DIANE MACDONELL, THEORIES OF DISCOURSE: AN INTRODUCTION (1986); Michel Foucault, *The Order of Discourse*, *in* LANGUAGE AND POLITICS 108-38 (Michael J. Shapiro ed., 1984).

tween *physis*, the world of nature, and *nomos*, the world of convention, law, and culture. The philosophical study of nomos includes ethics and political theory. However, it also includes culture itself as a philosophical problem and an object of study.

The philosophy of culture has a rich tradition and many illustrious forbearers, of whom Giambattista Vico, Jean-Jacques Rousseau, Immanuel Kant, and Georg W.F. Hegel are perhaps the most prominent. Some of the questions which the philosophy of culture asks are these: What is the relationship of culture to human existence and human history? What role does culture play in producing the faculty of human reason? Is human history, and hence the history of culture, a tragedy or a comedy, or is it a story with no determinate end and no narrative coherence or unity? Later philosophers, like Marx and Nietzsche, emphasized a further question, namely, the question of power. What power do culture and symbolic forms have over individuals? How can individuals recognize this power and what, if anything, can or should they do about it? From this perspective, the study of discourse, like the study of ideology, is merely the latest in a series of approaches to the philosophy of culture. The basic questions it asks are very much the same, and the problems it encounters are very similar.

In this Essay I discuss some of the problems in the philosophy of culture through an extended metaphor—the metaphor of cultural software. I compare certain features of culture, and the way that culture works, to the software that is installed on a computer and that enables a computer to process information.

This metaphor is misleading in several respects. First, I do not believe that the human mind works like any existing computer. Nor do I believe that thinking is primarily a mechanical or algorithmic process. On the contrary, human thinking is distinguished by its symbolic and metaphoric character, and its fundamental motivation in human values. Second, although I shall speak loosely of "biological hardware" and "cultural software," the distinction between hardware and software is not the same for humans as it is for computers. Each individual has a unique brain structure that is the product of genetic inheritance and is shaped and organized by her experiences and activities, especially those in early childhood. Thus, it is highly misleading to think of individu-

als as consisting of identical hardware into which identical copies of software are inserted.[5]

The growth of cognitive science and the search for forms of artificial intelligence have led naturally to comparisons between human beings and computers. One of the most important debates currently raging in the philosophy of mind is the extent to which mind should be defined functionally in terms of information states, like those in a computer. Some philosophers of mind have gone so far as to argue that the human mind is, at basis, indistinguishable from a computer, while others have asserted that the intentional nature of human intelligence makes such comparisons thoroughly inappropriate.[6]

Although these debates are interesting, they are to a large extent peripheral to the concerns of this essay. Unlike most cognitive scientists and most philosophers of mind, my focus is on the mind's relationship to culture and not its ultimate structure.[7] Howard Gardner has noted that although most cognitive scientists "do not necessarily bear any animus against the affective realm, against the context that surrounds any action or thought, or against any historical or cultural analyses, they attempt to factor out these elements to the maximum extent possible."[8] It is quite possible that the computational metaphor of mind has encouraged this trend. For these reasons, Jerome Bruner, himself one of the founders of the cognitive revolution, has recently called for a renewed emphasis on "the concept of meaning and the processes by which meanings are created and negotiated within a community."[9] These concerns lie at the heart of this essay; they motivate my use of the metaphor of cultural software. I am interested in using this metaphor as a metaphor, to illuminate the ways in which human beings are constituted by and express their values within a culture.

The philosophy of culture is also the philosophy of history: it asks how people exist as members of a culture in history. History is a peculiarly human phenomenon; the Grand Canyon changes in

---

5 For an accessible discussion of brain physiology explaining why such a simplistic hardware/software model must be wrong, see GERALD M. EDELMAN, BRIGHT AIR, BRILLIANT FIRE: ON THE MATTER OF THE MIND (1992).

6 *Compare* PHILIP N. JOHNSON-LAIRD, THE COMPUTER AND THE MIND: AN INTRODUCTION TO COGNITIVE SCIENCE (1988) *and* HILARY PUTNAM, REPRESENTATION AND REALITY (1989) *with* JOHN SEARLE, MINDS, BRAINS, AND SCIENCE (1984).

7 I should note that the very attempt to divorce these issues is itself controversial. *See* EDELMAN, *supra* note 5.

8 HOWARD GARDNER, THE MIND'S NEW SCIENCE: A HISTORY OF THE COGNITIVE REVOLUTION 41 (1985).

9 JEROME BRUNER, ACTS OF MEANING 11 (1990).

time, but only human beings have history. Or more correctly, the Grand Canyon has a natural history, but only human beings have a cultural history, which is history proper. Human beings begin to have history only at the moment when they enter into culture, which is also the moment that they begin to create collectively shared and created tools for understanding the world and articulating their values.

Culture is, in this sense, a set of collectively created tools of understanding. This brings us to another basic metaphor: that of the tool. Human beings are toolmakers and tool users. Culture itself is a tool. It is a tool used to make other tools.

Of course tools are always used for something. If culture is a tool, what is it used for? Human beings use culture for at least three interlocking and interrelated purposes. The first is to get about the world, to understand it and make use of it. The second is to interact with others as others, rather than as objects of manipulation. The third is to articulate and express human values.

I specifically distinguish these three purposes because many people think of tools and toolmaking only with respect to the first purpose. They understand tools largely as a way of exploring and mastering the natural world. This mastery can either be material or intellectual—either through controlling and shaping nature or through understanding her. Nevertheless, if this were the only purpose in toolmaking, the conception of culture as a set of tools, and the conception of humankind as toolmaker and tool user, would be significantly impoverished. Human culture would be thoroughly instrumental, nothing more than a means by which human beings mastered their environment. This conception of culture would fail to recognize the existence of other human beings, or in the alternative, it would view them as just another set of objects to be controlled, governed, studied, and mastered. Similarly, the concept of reason developed through culture would be reduced to instrumental rationality. It would not be able to reason about values or ends, but only about means.[10]

Human beings have values, and these values are one of the most important features of human life. Or more correctly, human

[10] We may justly criticize a culture to the extent to which it is primarily or excessively concerned with instrumental rationality at the expense of other forms of reason. Nevertheless, we should not confuse this criticism with a criticism of the metaphor of toolmaking or with the conception of human beings as toolmakers. That criticism is valid only if toolmaking really has no other purposes besides the mastery of objects. The critic's assumption that this is so, may itself be a symptom of living in a culture that has placed too high an emphasis on instrumental concerns and instrumental rationality.

beings value, because value is properly a verb, not a noun. I think this point especially worth stressing. Human beings possess an inexhaustible drive to evaluate, to pronounce what is good and bad, beautiful and ugly, advantageous and disadvantageous. Before culture, human values are inchoate and indeterminate; through culture they become differentiated, articulated, and refined.[11]

Now these purposes—understanding, interaction, and articulation—overlap. Getting about in the world and dealing with others presupposes ends and hence presupposes human values. Conversely, one of the most important human values is the urge to know and understand the world; it is related to the drive to master all problems and obstacles placed before us.

We employ the tools of culture towards these ends. For simplicity's sake, we might identify three basic types of cultural tools that human beings use, while keeping in mind that the three are not easily separated in practice. The first is technology, the second is institutions, and the third is cultural know-how, or what I would like to call cultural software.

What is cultural software? It consists of the associations, heuristics, metaphors, and capacities that we employ in the process of understanding and evaluation. An example of technology is a computer. An example of an institution is a bank. An example of cultural software is linguistic ability. Technology makes tools from materials, institutions make tools from human sociability, and cultural software makes tools from human understanding.

These cultural tools are interdependent and interrelated. For example, the institution of a bank may presuppose technology in the form of buildings, computers, furniture, and a workforce trained in a certain way, with certain understandings and abilities. Nevertheless, different philosophers of culture have emphasized some features more than others. For example, Marx emphasized the role of technology, and Vico emphasized the role of institutions.[12] But the third type of tool—cultural software—is equally important. Without cultural software, our technology lies on the ground, rusted from disuse, and our institutions fall apart. The bib-

---

[11] The phenomenon of articulation of inchoate human values is discussed more fully in J.M. Balkin, *Transcendental Deconstruction, Transcendent Justice*, 92 MICH. L. REV. 1131, 1139-41 (1994). My views have been greatly influenced by Thomas Seung. *See* T.K. SEUNG, INTUITION AND CONSTRUCTION: THE FOUNDATION OF NORMATIVE THEORY (1993).

[12] *See* JON ELSTER, MAKING SENSE OF MARX 143, 267-68 (1985); LEON POMPA, VICO: A STUDY OF THE "NEW SCIENCE" 30-34, 39-45 (2d ed. 1990).

lical story of the Tower of Babel is a good example of what becomes of technology and institutions without cultural software.

The comparison between computer software and cultural software encompasses two basic ideas. The first is that software is an indispensable tool for processing information and performing tasks. The second is that this software is an indispensable part of what we mean by "the computer."

Let me address these two points in turn. First, a computer uses software in order to process information. Without this software it cannot do its job; it cannot interact with the environment around it. If you boot up a computer but do not put any software in it, it just sits there and does nothing. You can type on it all you want, but it will not respond, or at best it will spit out an error message. It cannot process information because it has nothing to process information with. At best its ability to process information is primitive and unhelpful. Only when you install software can it do anything useful, and even then the type of information it can process depends on the kind of software installed. The most massive supercomputer installed with a checkers program still can only play checkers, although it can probably play checkers very quickly indeed. The potential power of the computer remains constant, but its practical power is severely limited. As software becomes increasingly developed so too does the practical power of the computer. In this way the potential power of the hardware is only fully realized through the development of increasingly elaborate software. Thus, we might say in a very loose sense that software empowers hardware.

The second point is that this software is, to a very important extent, constitutive of the computer, or rather, what we unthinkingly call the computer. Often what we mean by "the computer" is really the software together with the computer. Thus I might say that I wrote this essay on my computer, but really I wrote it using a word processing program installed on my computer. Note moreover, that a computer, or more precisely, a program running on a computer, becomes different from what it was before with each input of data. After each keystroke, the state of the computer is different, and how it processes succeeding information is different, based on what went before. Of course, with most software, the basic program remains unchanged each time someone runs it. It returns to a basic state after the program is exited and the computer is turned off. The only thing that changes is the data that has been manipulated. However, in fact, it is possible to design pro-

grams that are partially rewritten each time they are run, in response to the data previously entered, and the tasks previously performed.

This second point has a corollary: in order for hardware and software to interact, both must have a capacity to process information. My word processing program allows my computer to process information, but it can do so only because it is loaded onto another program, an operating system like DOS or UNIX, that allows the computer to run software. Hence, the information processing permitted by the software requires a prior information processor to employ it. Similarly, the operating system only runs because the computer has a program in firmware that allows it to understand and process the commands it receives from the operating system. Finally, this firmware can only operate because the hardware of the computer allows it to process the commands of the firmware at a mechanical level. Thus, the distinction between hardware and software is not a distinction between the part of the computer that processes information and the part that does not—rather, information processing occurs all the way down.

The distinction between hardware and software must be explained differently. In theory, my word processing program could be hard-wired into the computer. It could become part of the hardware. But in practice it is more convenient to allow me to remove it and substitute different programs, or upgrade the program that I have. This is the great advantage of software as an information processing device. It is easily changeable and adaptable; it creates the possibility of many different types of hardware/software combinations, and hence many different types of computers.[13]

Just as computer software allows computers to harness their power, cultural software empowers human beings. The human mind is a marvelous device, but like the most powerful supercomputer, it needs methods of understanding if its power is to be tapped. Our cultural software is the result of a long process of collective accumulation and construction. This process has produced elaborate tools of understanding, which, in conjunction with technology and institutions, can be tremendously empowering.

Of course, cultural software is not only empowering in the sense of allowing us to achieve our goals. It also enables us to re-

---

[13] For an evolutionary argument describing how the capacity to employ "software" might have developed in humans, see DANIEL C. DENNETT, CONSCIOUSNESS EXPLAINED 182-91 (1957).

flect upon and describe what our goals are. Cultural software allows human beings to articulate and concretize their values, to put flesh on the bones of their innate but inchoate urge to value and evaluate. Through cultural software our brute sense of the beautiful is transformed into the many varieties of aesthetic judgment, some of which come into being and fade away at different points in history. Through cultural software the inchoate sense of good and bad is transformed into the many varieties of moral and practical judgment, and the many virtues and vices are articulated and differentiated. Thus cultural software is the great enabling device not only of human understanding, but also of human evaluation. For this reason alone, it is the greatest of human creations, the most powerful and important of human tools.

Nevertheless, the tools of understanding that I call cultural software are different from hammers and nails. Hammers and nails are made by human beings but are separate from them. I can pick up a hammer or put it down. I can carry it with me or leave it at home. Not so with the tools of understanding. The tools of understanding work by becoming part of my apparatus of understanding, which is to say, they work by becoming part of me. Thus, cultural software is not just something that we use to understand and evaluate the world; it is also part of us. Indeed, we might even say that human beings do not become persons until they enter into culture, and hence become imbued with some form of cultural software. To exist as a person is to exist as a person who is part biological hardware and part cultural software. The two together constitute the person, just as what we sometimes mean by "the computer" is both its electronic hardware and the information coded as software. And just as the computer is only able to use software because it already has an information processing capacity, human beings are able to use culture only because they are already toolmakers and tool users. In the case of humans (but not computers) this also means that they already have ends and values, already can get around in the world using their natural tools, and can use tools to assist their natural tools.

An equally important aspect of cultural software, which is not true of all forms of computer software, is the fact that it is collectively created by the individuals who are constituted by it. Language is perhaps the most prominent form of collective cultural software. Like language, the forms of cultural software are constantly being written and rewritten. Just as languages evolve over time, so do the many forms of cultural software. This process of

evolution is one of bricolage—by which I mean starting with the available tools at hand and innovating on that basis. This collective product, the tools of understanding, evolves and mutates over time, and as it changes so, too, do the human beings who are constituted by it. In this way, we can see the history of humanity as the history of an ever changing and mutating set of tools of understanding, which are built on previous versions. To return to our computer metaphor, the history of culture is the history of perpetual upgrades to our cultural software. But unlike computer software, this process is not discrete but gradual, and none of us ever have identical copies.

The theory of cultural software is also a theory of historical existence. To exist in history means to be the bearer of a particular type of cultural software. So historical existence is not merely existence in time, but existence at a time when one is constituted by a particular form of cultural software. A person living in the sixteenth century has a different kind of existence from a person living in the twentieth. The biological hardware may remain roughly the same, but the cultural software is different. As a result, the person, who is the interaction of this biological hardware with cultural software, is different. This explains the difference between a person and the Grand Canyon. The Grand Canyon exists in time, but only people exist in history, because only people are constituted by an evolving, collectively created cultural software.

The theory of cultural software is also a theory of understanding. Human understanding is understanding in history. It is made possible by the tools of understanding, which bear the marks of their historical development. Thus, there is an intimate connection between historical existence and historical understanding, between living in history and understanding in history. This is another way of expressing Hans-Georg Gadamer's point that human understanding is made possible by, rather than hindered by, location within a tradition, and what he calls prejudice or prejudgment.[14] Indeed, you might think of the theory of cultural software as a sort of hi-tech analogue to the Gadamerian concept of tradition. To be human is just to be constituted by some type of cultural software, the product of a certain history of conceptual bricolage and evolution. This is both the meaning of historical existence and the condition of human understanding.

---

[14] *See* HANS-GEORG GADAMER, TRUTH AND METHOD 239-40, 245-53 (Garrett Barden & John Cumming eds. & trans., Crossroad 1975) (2d ed. 1965).

Put still another way, the theory of cultural software takes literally the contemporary chestnut that individuals are socially constructed. People become people only when they enter into culture, which is to say, only when culture enters into them, and becomes them, at the moment when they are programmed with and hence constituted by tools of understanding created by a culture at a certain point in history. People obtain and incorporate cultural tools, which become as much a part of them as their arms and legs.

The notion of cultural construction is often associated with cultural determinism. But the notion of cultural software suggests why this view is misleading. Cultural software empowers individuals even as it creates them. It untaps the potential power of the human mind just as an increasingly complicated and sophisticated software program allows a computer to do more. Therefore, we must understand cultural software as constitutive not only of identity, but of autonomy as well. The confusion of cultural construction with cultural determinism misunderstands what culture does for human beings. Culture is not the source of mechanical obedience but rather the wellspring of what we call freedom. Cultural software, rather than the enemy of human autonomy, forms the very conditions of its possibility.

This point leads us, at last, to a reevaluation of the concepts of ideology and discourse. Although cultural software empowers individuals, it also creates a certain opportunity for power over individuals who are constituted by it. It does both of these things at one and the same time and through the same mechanism. The power that cultural software makes possible is precisely the power that the tools of understanding have over the individuals who are partly constituted by them. This power is of two types. The first arises from the limitations of our conceptual apparatus; it is somewhat like the very powerful computer that only has a checkers program. The second type of power is more subtle. It arises from the nature of information processing itself, and it is never fully eliminated, no matter how sophisticated the software becomes.

Processing information always requires partiality and selectivity. As Heraclitus recognized, the world is in perpetual flux; we cannot comprehend its nature in all of its infinite diversity and differentiation.[15] Without some form of simplification, in the form of categorization, narrative, heuristics, or norms, it is impossible to understand anything at all. Information requires simplification—

---

[15] *See* G.S. KIRK & J.E. RAVEN, THE PRESOCRATIC PHILOSOPHERS: A CRITICAL HISTORY WITH A SELECTION OF TEXTS 197 (1975).

taming the flux for the purpose of understanding; and thus, at the very moment when understanding is made possible, partiality also emerges. I often like to say that the key to information is in formation; it lies in the selection and categorization of the flux of experience into comprehensible categories, events, and narratives. To understand we must establish similarities and differences, categories and narratives, canons and heuristics. These are the basis of all information, and hence the basis of our cultural software. Hence our cultural software limits even as it empowers. It informs us in forming us, which is to say that it informs us in forming our selves as selves endowed with a certain form of cultural software, who see things one way and not another, who are properly "tooled up" for some tasks but not for others. Thus, cultural software has power over us because this power is rooted in the very way that we are able to process information and articulate values.

Individuals within a culture can take advantage of the partiality of our cultural software. They can gain power over us because we, like they, are constituted by the tools of understanding. The most obvious example of this power is the power of symbols and rhetoric. Rhetoric has power because rhetorical features are already lodged in our cultural software. Symbols have power because the associations which make them symbols are already part of us. Hence, the study of rhetoric or the study of semiotics may be thought of as the study of cultural software, or more properly, the study of the traces and effects of cultural software. The study of cultural software is the study of the building blocks of our understanding, and therefore the study of the forms and modes of power exercisable over that understanding. At the same time it is the study of reason itself—the culturally created reason that is the basis of our practical action.

Thus, the theory of cultural software rethinks the traditional conception of ideology in two ways. First, it sees ideological power as the power that cultural software has over the persons who are constituted by it, who are persons because of it. Instead of viewing ideology as false beliefs held by subjects who preexist those beliefs, it locates the source of ideological power in the constitution of subjectivity itself. This subjectivity is not simply the meaning that others assign to you, but the meaning that you assign to the world itself through the shared tools of cultural software.

Second, the theory of cultural software argues that ideology must be understood not only through its negative effects but also through its positive ones. Ideology does not merely obscure; it

clarifies. It does not merely limit the imagination but empowers it as well. The theory of cultural software thus rejects what I like to call the "pathological" view of ideology, which sees ideology as a disease or a decrepit form of human thought. In the theory of cultural software, the mechanisms of ideological thought are the mechanisms of everyday thought. In this theory, truth and falsity, deception and empowerment enter through the same door.

There has always been debate in the literature on ideology as to whether the concept of ideology is necessarily pejorative—that is, opposed to truth or science—or instead may be a neutral description with no pejorative content.[16] A similar question arises for the study of discourse: Do we study discourse neutrally as a general feature of social life, or is our goal to expose unjust relations of power? The theory I propose offers a third conception. It is an ambivalent conception of ideology and discourse. Cultural software both empowers and disempowers. It is both a source of mystification and enlightenment, and these features are not always easily separated from each other. The ambivalent conception allows us to engage in ideological critique of our culture while recognizing how culture nevertheless enables us and enhances our freedom.

In this way, the theory of cultural software has a distinct advantage over both the pejorative and the neutral conceptions of ideology. The pejorative conception founders on what has been called Mannheim's paradox: If all discourse is ideological, how is it possible to have anything other than an ideological discourse on ideology?[17] This problem arises from the uniformly negative connotations of ideology in the pejorative conception. The neutral conception solves this problem by treating ideology as a nonjudgmental term. Yet a neutral conception cannot be fully successful, for, as Mannheim himself pointed out, evaluation eventually works its way into all studies of ideology and discourse.[18]

Mannheim's paradox disappears, however, when it is stated in terms of the theory of cultural software. It may be true that all discourse about cultural software must be stated through the use of cultural software. Yet, this is not an insurmountable obstacle be-

---

16 *See* MICHÈLE BARRETT, THE POLITICS OF TRUTH: FROM MARX TO FOUCAULT 18-19 (1991).

17 *See* CLIFFORD GEERTZ, THE INTERPRETATION OF CULTURES 194 (1973); PAUL RICOUER, LECTURES ON IDEOLOGY AND UTOPIA 157 (Geoge H. Taylor, ed., 1986).

18 *See* KARL MANNHEIM, IDEOLOGY AND UTOPIA: AN INTRODUCTION TO THE SOCIOLOGY OF KNOWLEDGE 88-89 (Louis Wirth & Edward Shils trans., Harvest/HBJ 1985) (1936).

cause the tools of understanding are empowering as well as limiting. They are enabling as well as confusing; indeed, they form the very conditions of autonomy and self-discovery. Cultural software always creates the possibility of a critical engagement with itself. By retracing the origins of the theories of ideology and discourse to the philosophy of culture, we vindicate the possibility of a critical approach to thought and belief. Like a clever entrepreneur, the theory of cultural software sees opportunity in what others regard as hopeless difficulty; where others see ideological determinacy, it sees the promise of critical possibility.