



PROBLEMS AND POSSIBILITIES OF GLOBAL TECHNO-SCIENTIFIC CULTURE: AN INTERVIEW WITH SHELDON RICHMOND



Phil Mullins and Sheldon Richmond

Sheldon Richmond. *A Way Through the Global Techno-Scientific Culture*. Newcastle-upon-Tyne: Cambridge Scholars Publishing, 2020. Pp. 197 + xxiii. ISBN (10): 1-5275-4626-8. ISBN (13): 978-1-5275-4626-4. Hardback £61.99.

Keywords: STS (science technology studies), computing, culture, media ecology, global, internet, socratic social architecture

ABSTRACT

In this interview, Phil Mullins questions Sheldon Richmond about the main ideas developed in his 2020 book A Way Through the Global Techno-Scientific Culture.

Mullins: I have been looking for a fitting general rubric under which to place your new book, *A Way Through the Global Techno-Scientific Culture*, and it seems to be “philosophy and technology.” You did graduate work with Joseph Agassi and have written about aesthetics, Michael Polanyi, and Karl Popper, but you spent more than 30 years as a systems analyst working for a large organization. You were on the job in the period in which what I term “digital culture” began emerging. Your book shows that you have digested much of the rather large and daunting theoretical literature in computer science that charts some of the fateful twists and turns taken leading to the contemporary digital world. But you also analyze digital culture using ideas of Popper, Polanyi, and a host of other philosophical thinkers and social critics. As your title suggests, your book’s agenda is to outline a way to transform what you term “global techno-scientific culture.” Let me give you a chance to respond to my very general characterization of your book and your background as an author.

Richmond: *To be honest, I was unsure what to call our current era and how to title my book so that it would be descriptive of our era. I hit upon the name “global techno-scientific culture” after eliminating various other labels for the era. What is of real importance is the recognition that we are now in an unusual moment in human history: we have developed a socio-technical culture that is global, and monopolistic, and is a misfit for humanity. Worse, we condemn ourselves to serve the computer technology that is bound up with the global techno-scientific culture that we have developed. How do we get through this subservience to our own tools and institutions, and the anti-humanism of tools and institutions of our own making, of our own decisions and choice? That I think is*

the global problem we now face. What can we and must we do now, immediately, to make our way through the global techno-scientific culture?

Before I fell into work in the field of computer technology, I was a philosopher who knew “everything” from a very abstract, impersonal point of view. I was a philosopher who regarded grand theoretical systems as the ultimate answers to the fundamental questions. I wrote essays and a book from this impersonal point of view. But the accidents of life took me to work in IT. During this time, there was no moment of enlightenment, but, through a gradual process of writing about my situation in the giant beehive of a corporate socio-technical structure where I was a worker bee keeping the technical part of the structure functioning, I assembled the picture in this book. I only saw what I thought was a book with a single focus after I exited the IT world: I recognized that I knew the insides of the corporate computer machine, and had helped make it operational and this work, unwittingly and unintentionally, assured that many people, including those at the top, became servants of computer technology.

Mullins: One more preliminary remark on your book’s structure: there are six chapters but from the beginning, you work to overcome the linear organization common in printed literature. Your table of contents is annotated, showing the modules in each chapter and providing a summary. Your Prologue provides a general summary, and, somewhere in this forematter, you invite the reader to skip to the Epilogue which also provides an overview and modules with a general summary written from a slightly different angle of vision. In a word, your book emulates electronic text; it is a hypertext. You assume enough pieces can be picked up by the reader in whatever way he/she wants to approach the book and you suggest a variety of approaches.

Richmond: *Exactly. Hypertext was invented by Ted Nelson who does not get the attention he deserves as a philosopher or as a technology designer. Indeed, hypertext is the core idea of the World Wide Web invented by the engineer and computer scientist, Tim Berners-Lee, who used the concept of the hypertext for developing the protocol called, “HTTP,” as the basis for linking documents on the internet. Also, not enough attention is given to Paul Baran, a RAND engineer who developed the technology of decentralized and distributed networks, a core idea for the structure of the internet. My son Elken did the graphic for the book cover, which is based on the drawing by Paul Baran for representing how decentralized and distributed networks work. The guiding idea for the structure of my book is decentralized, distributed networks. Start anywhere in the network, and you can still achieve your goal: an infinite and recursive network of ideas, always returning to itself.*

Much of the reading I did was during my commutes by bus and subway. I appreciated hard to find books that told their story, even if long and deep, repeatedly, concisely, and in short versions and variations. This, in part, is responsible for how I designed my book. Cognitive psychologist George Miller’s theory of how working memory functions also influenced me: working memory functions best with chunks of seven, plus or minus two, elements. Also important was Claude E. Shannon’s mathematical theory of communication, which boils down to the practical principle that we increase the probability of receiving information by repeating the information. The longer the message is, the less probable it is that it will be received.

Another way of looking at the structure of this book is to say that its architecture mimics the message: distributed, decentralized social and technical architecture is a way through not only the book, but also through the global techno-scientific culture.

Mullins: Given the hypertextual way you think about communication and have organized your book, I turn directly to one of your longer, later chapters, “Philosophers,” where you challenge everyone with any philosophical interest to become engaged in transforming contemporary culture, to work on implementing a “socratic social architecture in all institutions” (77). What does this odd phrase mean? You also suggest that we should develop the approach of “client-server architecture” for social and cultural change. But before you respond, let me also note that in your chapter you roundly indict much mainstream philosophy as not responsibly oriented toward practice. Your sharpest criticisms are directed toward “computational philosophy” where a long-running debate continues about how much human minds are akin to digital computers. This discussion in computational philosophy (including the “transhumanism” spinoffs) is a distraction, in your account, since philosophers need to be attending to the question of “how is what is going on with digital processor technology affecting and transforming humanity, civilization and our humaneness” (103). You argue philosophers must turn to the everyday issues of emerging digital culture where the “failures and frustrations with digital technology” legitimize a techno-elite and trap and “dummify and mechanize techno-subjects” (104).

Richmond: *I will first explain “socratic social architecture” and second, outline my critique of some current philosophers.*

Let me illustrate how socratic social architecture functions by discussing the social architecture of the university. A lecture is a top-down situation where students transcribe notes on “smart” devices these days. Very little time is left for questions.

But a seminar or open discussion situation as opposed to the lecture is basically distributed and decentralized. The students often read their papers, and/or raise questions, and the professor and other students comment and may jump in with answers. Plato’s Symposium is the model for the modern-day social architecture of a decentralized distributed interactive teaching and learning system.

However, in the computer world, the von Neumann architecture is the basic top-down technical architecture for digital machines. The CPU (central processing unit) runs the show, through following the instructions or program(s) stored in memory. The CPU, running a stored program, transforms the input-data, and sends out the transformed-data to a printer, or screen. The person sitting at the screen is a passive receiver of the information, the output of the data. Where the interaction may occur between the human and the machine is the human acting as an input device for the machine: clicking on virtual buttons on the display, touching areas of a touch screen, tapping the keys of a keyboard, talking to the machine that uses voice-recognition software. Even those who play games on computers are input devices for the game software. Also, as users of digital computers, we are very versatile: people can be output devices as well. As output devices, we passively receive information on the monitor, or from the audio, or from the printer, and transfer the information to our brains, and may use that stored information for further computer-interaction, inputting other data for the computer. People may also act as communication devices for computers and transfer the information received from the computer to other people by speech, by email, by social media, by messaging apps; those people who have received the information or data from others, then use that information as input-data while functioning as input-devices for computers.

I see this form of computer-interaction as top-down where the CPU, with its software or apps, acts as our “lecturer” and we react to the “lecture”: our interaction is very passive, following the rules for interaction as an

input-device—posing “questions” or inputting data to the computer that the CPU and software transform into output-data, giving us our “answers” as transformed-data. Also, people follow the rules for interaction with the CPU and software, as an output-device. The sum and total of it is, concerning our interaction with “smart devices,” that we are basically peripheral devices for computer technology: both as input and output devices.

Fundamentally opposed to the top-down, centralized functioning of the technical architecture for computer technology, is the technical architecture and functioning of computer networks, such as the internet. The internet is distributed and decentralized and is akin to socratic social architecture. One name for distributed and decentralized computer architectures is “client-server” where your home or office PC is a “client” PC on the network. Both the internet and private networks and PCs from which you access information on the network are called “servers.” Another name is “peer-to-peer”, where your home PC and other people’s home PCs function both as “servers” and “clients,” distributing information, files, and apps to each other on a distributed and decentralized network.

Basically, I propose that we refashion our social institutions, including our educational institutions, to mirror the Socratic Symposium where everyone takes turns as the lecturer, and fields comments and questions from the other students. This is what I call “socratic social architecture.” The internet and those who designed the internet likely did not have implementing “socratic social architecture” in mind; rather they were thinking of the most efficient and fail-safe method of communicating data and accessing information and programs from computers.

Secondly, my critique of some philosophers: I am a philosopher, but I have a very traditional, socratic idea of philosophy, its practice and teaching. Namely, I think philosophy is a critical enterprise: Socratic philosophers are critical interpreters of ideas (and disciplines) developed by others, following Plato’s version of Socrates going into the marketplace to join with people in open discussion. Open discussion is hindered when philosophers use highly specialized and technical jargon, where most of the time spent in discussion with others who are non-philosophers involves explaining the jargon. Ordinary language philosophers, and Wittgenstein and his followers, argued that philosophical language is “language on holiday.” Rather than solve philosophical problems, philosophers have created a language that disguises, distorts, and deepens those very problems. I picked on the computational philosophers of mind as the prime example of such philosophers. Many of those philosophers, for the most part studied Wittgenstein and ordinary language philosophy, as well as analytic philosophy, where the analysis of language, and more lately, the use of language in an exact and precise manner, almost as mathematical formulae, is thought to be the tool for solving or dissolving philosophical problems. Ironically, these analytic and neo-Wittgensteinian philosophers have created their own jargon that keeps others not in their “school” in the dark. This jargon and mystique side steps the impact of computers by identifying minds with computers. The identification of minds with computers allows for the treating of humans by those in the technological elite (i.e., the “techno-elite”), as functions of computer systems. I think those philosophers who promulgate computational philosophy have become unwitting ideologues for the computer-machine and its domination of all cultures, turning our variety of cultures, globally, into off-shoots and subsidiary sub-cultures of the new global and monopolistic culture of digital technology.

Moreover, identifying mind with computation is based on three well-known philosophical errors or fallacies. The first error is a semantic error: philosophers who use predicates (properties) that apply to humans, to describe certain operations of machines, are applying predicates to the wrong category. The second error involves wrongly identifying certain behaviours of an object with the identity of an entity or object. It is the quack-and-walk-like-a-duck-is-a-duck error. Because we can get computers to simulate certain cognitive functions does not mean that they are cognitive entities; they are just machines that have certain functions or behaviours similar to cognition.

Simulating even “learning,” does not mean that the machine “learns” and thus is a “learning-machine.” The third error identifies a social decision with a real or natural happening: to treat how computer technology has developed and is used as something natural rather than as the output of various social decisions is a very old and deep-seated error that humans often make and have made with various cultural and social arrangements. For instance, bosses or managers are not required to control the operations of an organization in all cases, for all organizations. In other words, the top-down, hierarchical structures of organizations are the result of social decisions and not part of so-called “human nature”—not a result of the biological evolution of humans as supposed “naked apes.” Hierarchical institutions are not natural, but are the result of social decisions that are no different than the social decision to drive on the right side of the road; no different than the social decision to have lecture-based teaching and learning in universities, rather than to have symposium-based teaching and learning in universities.

Mullins: You argue a certain mystique about computer technology operates in emerging digital culture and this mystique has had social and political fallout. Please comment on this mystique and briefly sketch the contours of its fallout.

Richmond: *The mystique of computer technology and its institutions operates to keep people ignorant and subservient. Look at the language we use to talk about computers. I don’t mean the technical language and the acronyms used as short forms, such as RAM (Random Access Memory), CPU (Central Processing Unit), VLSI (Very Large Scale Integration) and so forth. Rather, I am talking about using the words “smart,” “intelligent,” and cognates related to thinking and intelligence, concerning computer technology. Calling mobile phones “smart phones,” has become the least of it since we also speak of “smart cities” and “smart watches.” Now there are so-called smart devices of all sorts. What the mystique does is disguise what is going on: our intelligence is not merely transferred to computers (such as doing complex financial calculations), but our intelligence is removed from us, in that decision making, and many other activities requiring intelligence are being progressively transferred to computer systems, eventually without the intervention of humans because we have begun to think those machines are “smart” (even without yet having “artificial intelligence”). Moreover, many philosophers of mind and computer scientists with their computational theories of mind, uncritically talk about how computers form “mental models”, “think”, “learn”, have “sensory-input” or “perceive”, and “recognize” faces. Those philosophers and computer scientists function unintentionally as ideologues for computer technology, and also become subservient functionaries of the global computer-machine.*

These processor-embedded devices are not “super-intelligent” machines. They have no intelligence whatsoever. They are just ordinary machines that operate as all machines do: “wind them up” and they go according to human design, such as ordinary self-winding watches, which mechanically wind when one moves one’s arm during normal activity. Computer technology is no smarter than self-winding watches; no more “intelligent” than humans even when we allow them to make decisions for us. We just transfer a function to them, that we think is intelligent, and we label the machine—if it has micro-chips that use algorithms and data—“smart”.

The outcome is to keep people in the dark about what is going on globally with computers: we serve computers, including the techno-elite who benefit from our ignorance due to the mystique of computers. In other words, we are undergoing a role-reversal between us and our own technologies. We have ended up increasingly serving our computer technology and the institutional systems surrounding our computer technology.

Mullins: Your analysis adapts both C. P. Snow's "two cultures" account, and Polanyi's modified account positing a continuity between science and the humanities.

Richmond: *Snow was insightful in realizing that science is a culture, with values, with tradition, with special rites and rituals, with a language or dialect or vocabulary or jargon, with a process for teaching the values, tradition, rites and rituals to those who enter the culture; and the Humanities are also a culture. However, as Polanyi argues, the culture of science is also humanistic, even though the Humanities are foreign to many scientists. Moreover, the culture of science with its humanistic aspects, along the lines most keenly recognized by Polanyi, has been transformed. The social conditions have changed to the degree that the humanistic aspects of science, and of technologies previous to computer technology, have shifted. The shift involves a complete transformation of science into an instrument, an app if you will, for computer technology. The humanistic aspect of science that Polanyi emphasized has been excised and replaced with the practice of reducing scientific understanding to algorithmic processes and unexplained formulae that even scientists do not understand. Nobody understands quantum mechanics, according to Richard Feynman.*

Mullins: So-called information has proliferated, and significant human understanding seems to be declining in global techno-scientific culture. You provocatively suggest computer technology is making knowledge extinct. Real knowledge is about something other than itself (and you sometimes dub this "objective knowledge") but in the information economy we generate primarily "nominal knowledge" (31), which is preoccupied largely with massaging and manipulating symbols currently valent in the milieu, and nominal knowledge is indifferent to the truth. With this general cultural diagnosis (which draws on figures like C. P. Snow but also Neil Postman), you combine analysis and criticism of the social choices made in modern society in developing computer tools. You argue that machines are really not intelligent, but the social choice has been made to regard them as smart. Further, the average user is now under the control of IT staff. What you bring together into this broad macroscopic account of our cultural situation is shocking. Can you unpack a bit more your claims about the extinction of knowledge and the enslavement of most end-users of computers?

Richmond: *Shocking but not new! Reading Plato's account of Socrates on the technology of writing (which I did as an undergrad, a graduate student, and even taught) but shoved into the back of my mind, this again became focal to my understanding of how computer technology has transformed knowledge. I am talking about knowledge that is about something, that has real-world, real-life reference, and I label that knowledge "objective" knowledge that is about something or other. When studying say psychology, physics, or philosophy, or any subject matter, we think we are learning something about humans, the world, the conditions of our lives, and about past lives, and past societies, and even about what people in the past thought, and how they experienced the world. When we have gained this knowledge, whether we call it as I have "objective" knowledge, or "substantive" knowledge, or "real/genuine" knowledge, we think we know about something other than the symbols we use when discussing our knowledge, or teaching our knowledge to others. Instead, we are doing what Socrates said that technology would do: the use of writing technology, according to the Socratic critique, gives people a semblance of knowledge through using the words, symbol systems, without even realizing to what those symbols refer. This is what our use of computers does: manipulates symbols without reference to anything outside symbols, as mere tokens. It is a game without meaning: even chess as a game with arcane rules for moving the chess pieces has a reference (purportedly, war with battles, and strategies). Once we buy into the pretense that computers produce knowledge, as I realized*

by reflecting on and extending to our current writing technology the Socratic critique of writing-technology, we will lose “authentic” knowledge: knowledge about something other than how to play the games of symbol-system manipulation. Losing authentic or objective knowledge is part and parcel of our role reversal with computer technology: we serve computers and that is what I see as our enslavement to computers. We think computers have the knowledge, and we seek knowledge from computers. By seeking knowledge from computers, we get nominal knowledge, not objective knowledge, and we entrench ourselves in slavery, as servants to computer technology. As slaves to computers, we exile ourselves not only from what is central to humanity, objective knowledge, but also we exile ourselves from our own humanity. That is indeed shocking.