SOFTWARE AND MIND

Andrei Sorin

EXTRACT

Chapter 8: *From Mechanism to Totalitarianism*

This extract includes the book's front matter and chapter 8.

This chapter examines the totalitarian tendencies of the mechanistic philosophy, and particularly their manifestation in our mechanistic software culture.

The entire book, each chapter separately, and also selected sections, can be viewed and downloaded free at the book's website.

www.softwareandmind.com

SOFTWARE AND MIND

The Mechanistic Myth and Its Consequences

Andrei Sorin

ANDSOR BOOKS

Published by Andsor Books, Toronto, Canada (www.andsorbooks.com)
First edition 2013. Revised 2019.

For disclaimers see pp. vii, xvi.

Designed and typeset by the author with text management software developed by the author and with Adobe FrameMaker 6.0. Printed and bound in the United States of America.

Acknowledgements

Excerpts from the works of Karl Popper: reprinted by permission of the University of Klagenfurt/Karl Popper Library.

Excerpts from *The Origins of Totalitarian Democracy* by J. L. Talmon: published by Secker & Warburg, reprinted by permission of The Random House Group Ltd.

Excerpts from *Nineteen Eighty-Four* by George Orwell: Copyright ©1949 George Orwell, reprinted by permission of Bill Hamilton as the Literary Executor of the Estate of the Late Sonia Brownell Orwell and Secker & Warburg Ltd.; Copyright ©1949 Harcourt, Inc. and renewed 1977 by Sonia Brownell Orwell, reprinted by permission of Houghton Mifflin Harcourt Publishing Company.

Excerpts from *The Collected Essays, Journalism and Letters of George Orwell*: Copyright ©1968 Sonia Brownell Orwell, reprinted by permission of Bill Hamilton as the Literary Executor of the Estate of the Late Sonia Brownell Orwell and Secker & Warburg Ltd.; Copyright ©1968 Sonia Brownell Orwell and renewed 1996 by Mark Hamilton, reprinted by permission of Houghton Mifflin Harcourt Publishing Company.

Excerpts from *Doublespeak* by William Lutz: Copyright ©1989 William Lutz, reprinted by permission of the author in care of the Jean V. Naggar Literary Agency.

Excerpts from *Four Essays on Liberty* by Isaiah Berlin: Copyright ©1969 Isaiah Berlin, reprinted by permission of Curtis Brown Group Ltd., London, on behalf of the Estate of Isaiah Berlin.

Library and Archives Canada Cataloguing in Publication

Sorin, Andrei
Software and mind : the mechanistic myth and its consequences / Andrei Sorin.

Includes index.
ISBN 978-0-9869389-0-0

1. Computers and civilization. 2. Computer software – Social aspects.
3. Computer software – Philosophy. I. Title.

QA76.9.C66S67 2013 303.48'34 C2012-906666-4

Don’t you see that the whole aim of Newspeak is to narrow the range of thought?… Has it ever occurred to you … that by the year 2050, at the very latest, not a single human being will be alive who could understand such a conversation as we are having now?

George Orwell, *Nineteen Eighty-Four*

Disclaimer

This book attacks the mechanistic myth, not persons. Myths, however, manifest themselves through the acts of persons, so it is impossible to discuss the mechanistic myth without also referring to the persons affected by it. Thus, all references to individuals, groups of individuals, corporations, institutions, or other organizations are intended solely as examples of mechanistic beliefs, ideas, claims, or practices. To repeat, they do not constitute an attack on those individuals or organizations, but on the mechanistic myth.

Except where supported with citations, the discussions in this book reflect the author's personal views, and the author does not claim or suggest that anyone else holds these views.

The arguments advanced in this book are founded, ultimately, on the principles of demarcation between science and pseudoscience developed by philosopher Karl Popper (as explained in "Popper's Principles of Demarcation" in chapter 3). In particular, the author maintains that theories which attempt to explain non-mechanistic phenomena mechanistically are pseudoscientific. Consequently, terms like "ignorance," "incompetence," "dishonesty," "fraud," "corruption," "charlatanism," and "irresponsibility," in reference to individuals, groups of individuals, corporations, institutions, or other organizations, are used in a precise, technical sense; namely, to indicate beliefs, ideas, claims, or practices that are mechanistic though applied to non-mechanistic phenomena, and hence pseudoscientific according to Popper's principles of demarcation. In other words, these derogatory terms are used solely in order to contrast our world to a hypothetical, ideal world, where the mechanistic myth and the pseudoscientific notions it engenders would not exist. The meaning of these terms, therefore, must not be confused with their informal meaning in general discourse, nor with their formal meaning in various moral, professional, or legal definitions. Moreover, the use of these terms expresses strictly the personal opinion of the author – an opinion based, as already stated, on the principles of demarcation.

This book aims to expose the corruptive effect of the mechanistic myth. This myth, especially as manifested through our software-related pursuits, is the greatest danger we are facing today. Thus, no criticism can be too strong. However, since we are all affected by it, a criticism of the myth may cast a negative light on many individuals and organizations who are practising it unwittingly. To them, the author wishes to apologize in advance.

Contents

Preface

This revised version (currently available only in digital format) incorporates many small changes made in the six years since the book was published. It is also an opportunity to expand on an issue that was mentioned only briefly in the original preface.

Software and Mind is, in effect, several books in one, and its size reflects this. Most chapters could form the basis of individual volumes. Their topics, however, are closely related and cannot be properly explained if separated. They support each other and contribute together to the book's main argument.

For example, the use of simple and complex structures to model mechanistic and non-mechanistic phenomena is explained in chapter 1; Popper's principles of demarcation between science and pseudoscience are explained in chapter 3; and these notions are used together throughout the book to show how the attempts to represent non-mechanistic phenomena mechanistically end up as worthless, pseudoscientific theories. Similarly, the non-mechanistic capabilities of the mind are explained in chapter 2; the non-mechanistic nature of software is explained in chapter 4; and these notions are used in chapter 7 to show that software engineering is a futile attempt to replace human programming expertise with mechanistic theories.

A second reason for the book's size is the detailed analysis of the various topics. This is necessary because most topics are new: they involve either entirely new concepts, or the interpretation of concepts in ways that contradict the accepted views. Thorough and rigorous arguments are essential if the reader is to appreciate the significance of these concepts. Moreover, the book addresses a broad audience, people with different backgrounds and interests; so a safe assumption is that each reader needs detailed explanations in at least some areas.

There is some deliberate repetitiveness in the book, which adds only a little to its size but may be objectionable to some readers. For each important concept introduced somewhere in the book, there are summaries later, in various discussions where that concept is applied. This helps to make the individual chapters, and even the individual sections, reasonably independent: while the book is intended to be read from the beginning, a reader can select almost any portion and still follow the discussion. In addition, the summaries are tailored for each occasion, and this further explains that concept, by presenting it from different perspectives.

❖

The book's subtitle, *The Mechanistic Myth and Its Consequences*, captures its essence. This phrase is deliberately ambiguous: if read in conjunction with the title, it can be interpreted in two ways. In one interpretation, the mechanistic myth is the universal mechanistic belief of the last three centuries, and the consequences are today's software fallacies. In the second interpretation, the mechanistic myth is specifically today's mechanistic *software* myth, and the consequences are the fallacies *it* engenders. Thus, the first interpretation says that the past delusions have caused the current software delusions; and the second one says that the current software delusions are causing further delusions. Taken together, the two interpretations say that the mechanistic myth, with its current manifestation in the software myth, is fostering a process of continuous intellectual degradation – despite the great advances it made possible.

The book's epigraph, about Newspeak, will become clear when we discuss the similarity of language and software (see, for example, pp. 409–411).

Throughout the book, the software-related arguments are also supported with ideas from other disciplines – from the philosophies of science, of mind, and of language, in particular. These discussions are important, because they show that our software-related problems are similar, ultimately, to problems that have been studied for a long time in other domains. And the fact that the software theorists are ignoring this accumulated knowledge demonstrates their incompetence.

Chapter 7, on software engineering, is not just for programmers. Many parts (the first three sections, and some of the subsections in each theory) discuss the software fallacies in general, and should be read by everyone. But even the more detailed discussions require no previous programming knowledge. The whole chapter, in fact, is not so much about programming as about the delusions that pervade our programming practices, and their long history. So this chapter can be seen as a special introduction to software and programming; namely, comparing their true nature with the pseudoscientific notions promoted by the software elite. This study can help both programmers and laymen to understand why the incompetence that characterizes this profession is an inevitable consequence of the mechanistic software ideology.

The book is divided into chapters, the chapters into sections, and some sections into subsections. These parts have titles, so I will refer to them here as *titled* parts. Since not all sections have subsections, the lowest-level titled part in a given place may be either a section or a subsection. This part is, usually, further divided into *numbered* parts. The table of contents shows the titled parts. The running heads show the current titled parts: on the right page the lowest-level part, on the left page the higher-level one (or the same as the right page if there is no higher level). Since there are more than two hundred numbered parts, it was impractical to include them in the table of contents. Also, contriving a short title for each one would have been more misleading than informative. Instead, the first sentence or two in a numbered part serve also as a hint of its subject, and hence as title.

Figures are numbered within chapters, but footnotes are numbered within the lowest-level titled parts. The reference in a footnote is shown in full only the first time it is mentioned within such a part. If mentioned more than once, in the subsequent footnotes it is abbreviated. For these abbreviations, then, the full reference can be found by searching the previous footnotes no further back than the beginning of the current titled part.

The statement "italics added" in a footnote indicates that the emphasis is only in the quotation. Nothing is stated in the footnote when the italics are present in the original text.

In an Internet reference, only the site's main page is shown, even when the quoted text is from a secondary page. When undated, the quotations reflect the content of these pages in 2010 or later.

When referring to certain individuals (software theorists, for instance), the term "expert" is often used mockingly. This term, though, is also used in its normal sense, to denote the possession of true expertise. The context makes it clear which sense is meant.

The term "elite" is used to describe a body of companies, organizations, and individuals (for example, the software elite). The plural, "elites," is used when referring to several entities within such a body.

The issues discussed in this book concern all humanity. Thus, terms like "we" and "our society" (used when discussing such topics as programming incompetence, corruption of the elites, and drift toward totalitarianism) do not refer to a particular nation, but to the whole world.

Some discussions in this book may be interpreted as professional advice on programming and software use. While the ideas advanced in these discussions derive from many years of practice and from extensive research, and represent in the author's view the best way to program and use computers, readers must remember that they assume all responsibility if deciding to follow these ideas. In particular, to apply these ideas they may need the kind of knowledge that, in our mechanistic culture, few programmers and software users possess. Therefore, the author and the publisher disclaim any liability for risks or losses, personal, financial, or other, incurred directly or indirectly in connection with, or as a consequence of, applying the ideas discussed in this book.

The pronouns "he," "his," "him," and "himself," when referring to a gender-neutral word, are used in this book in their universal, gender-neutral sense. (Example: "If an individual restricts himself to mechanistic knowledge, his performance cannot advance past the level of a novice.") This usage, then, aims solely to simplify the language. Since their antecedent is gender-neutral ("everyone," "person," "programmer," "scientist," "manager," etc.), the neutral sense of the pronouns is established grammatically, and there is no need for awkward phrases like "he or she." Such phrases are used in this book only when the neutrality or the universality needs to be emphasized.

It is impossible, in a book discussing many new and perhaps difficult concepts, to anticipate all the problems that readers may face when studying these concepts. So the issues that require further discussion will be addressed online, at *www.softwareandmind.com*. In addition, I plan to publish there material that could not be included in the book, as well as new ideas that may emerge in the future. Finally, in order to complement the arguments about traditional programming found in the book, I have published, in source form, some of the software I developed over the years. The website, then, must be seen as an extension to the book: any idea, claim, or explanation that must be clarified or enhanced will be discussed there.

CHAPTER 8

From Mechanism to Totalitarianism

The End of Responsibility

The irresponsibility that characterizes the world of programming, and the apathy of the rest of society, form an extraordinary phenomenon – a phenomenon that warrants closer analysis. Why are we ready to tolerate, in the domain of programming, failures that in other domains would be easily recognized to be due to incompetence or corruption? In the present section, I will try to explain this phenomenon by showing that it is an inevitable consequence of our mechanistic culture.

Software Irresponsibility

1

"No one ever got fired for buying IBM." This famous saying illustrates perfectly the irresponsibility that defines corporate computing. The saying dates from the 1970s, and perhaps even earlier, so it also serves to remind us that corporate computing was always controlled by bureaucrats.

What the saying implied was that buying IBM hardware and software was a *safe* decision. If the project failed, no one would be blamed; the conclusion would be that everything humanly possible had been done, and the project would also have failed with any other kind of hardware or software. Buying another brand, on the other hand, was a *risky* decision. A failure in this case might have led to the conclusion that the project would have succeeded had IBM been chosen instead. (In reality, the saying exaggerates IBM's role in absolving decision makers from their responsibility: there were many cases where managers did *not* buy IBM, the project failed, and still no one got fired. Hardly anyone in the world of corporate computing is blamed for failures.)

The saying is about IBM because it was coined at a time when IBM was dominating the computer industry. Here we are concerned, however, with the mentality behind it – a mentality that has not changed.

Because of their ignorance, software practitioners perceive application development as a dangerous phenomenon, over which human beings have little control. Primitive people, when recognizing their own ignorance, invent superstitions and magic systems to cope with difficult situations. Similarly, managers and programmers have created a rich culture of *computer* beliefs, myths, and magic systems. (We discussed this in "Anthropology and Software" in the introductory chapter.) They attribute the success of computing projects, not to knowledge and skills, but to selecting the correct systems and performing the proper acts. Their ignorance has given rise to the idea that the success of a project is due to some unexplainable power, which appears to inhere in certain types of hardware or software but not in others. The bureaucratic mentality is similar to the primitive mentality: both stem from ignorance, and both result in intellectual stagnation. Instead of expanding their knowledge to deal with difficult problems, the software bureaucrats, like the primitives, develop elaborate systems of belief.

IBM's high profits in the early period (profits which allowed it to establish its monopolistic position) were due in large measure to the systematic exploitation of this bureaucratic mentality. IBM was aware of the ignorance of its customers, and their perception of computers not as business tool but as status symbol. Thus, as corporate decision makers could not assess rationally the cost and benefits of business computing, IBM was able to maintain arbitrarily high prices. Instead of helping companies to develop expertise in the programming and use of computers, IBM fostered ignorance and irresponsibility by encouraging everyone to view the perpetual adoption of new equipment, regardless of cost, as a business necessity.[1]

[1] Joan M. Greenbaum, *In the Name of Efficiency: Management Theory and Shopfloor Practice in Data-Processing Work* (Philadelphia: Temple University Press, 1979), pp. 133–135. These issues were brought up at IBM's antitrust trial.

Thus, the attitudes we see in today's decision makers were born in the first years of business computing. Recognizing that customer ignorance is more profitable than expertise, the computer companies played an important part in the evolution of the software bureaucracy. It was against their interest that software practitioners be knowledgeable, responsible, and creative, so they did all they could to reduce business computing to trivial tasks – tasks that appealed *only* to people with a bureaucratic mentality. This ensured that no talented individuals, no true professionals, could remain in the world of corporate computing. In particular, the computer companies contributed greatly to the redefinition of programming expertise as expertise in ways to *avoid* programming, and to depend instead on complicated development tools and ready-made pieces of software.[2]

As a result of this scheme, the typical programmer or data-processing manager emerging from that period is a person who knows very little about computers, programming, or business, but who enjoys a position of prestige in society thanks to the propaganda conducted on his behalf by the elites. These incompetents induce their employers to spend large amounts of money on hardware and software novelties, and in exchange, the elites flatter them by depicting them (with the assistance of the media) as skilled professionals whose expertise is vital to their company's success. Everyone gets to respect them and to depend on them, and they themselves are convinced that what they are doing is important and difficult, when in reality they are puppets manipulated by the elites.

Although IBM no longer dominates the computer industry, the saying can still be used to describe the software bureaucrats. Today no one gets fired for buying Microsoft, or for buying client/server, or object-oriented, or 4GL, or relational, or data warehouse, or cloud, or artificial intelligence, or anything else deemed to be the correct choice.

Take CASE, for example – the idea that it is possible to develop business applications without programming. For many years, CASE was promoted by the software companies, by professors and gurus, by the business and computer media, and by professional associations, as the indisputable next

[2] It is an old maxim in the computer industry that good software sells hardware. But, even if the promotion of various types of software to help sell hardware started as an honest business strategy, the computer companies realized very quickly that, while good software indeed sells hardware, *bad* software sells even more hardware. Most hardware expenses are induced by bad software; and since bad software is the result of programming incompetence, it is not surprising that the computer companies never encouraged programming expertise. There are many ways for programming incompetence to demand more expensive hardware; for example, a more powerful computer is needed to compensate for the inefficiency of an inferior application, or to permit the use of large development environments, database systems, and packaged applications – all intended as substitutes for programming expertise.

stage in "software engineering." We will never know how many billions of dollars were wasted by businesses on this idea, before abandoning it. And the fact that they could not recognize its absurdity just by looking at it demonstrates the ignorance that managers and programmers suffer from. CASE had to be actually tried in thousands of projects before they realized that it didn't work. Even then, they could not see why the idea was fundamentally mistaken, so they learned nothing from this experience. (We discussed the CASE fallacy in chapter 6; see pp. 465–469.)

Needless to say, no one got fired for buying CASE. Nor did its failure affect the credibility of the companies that sold CASE systems, or the experts that advocated it, or the publications that promoted it. They simply went on inventing and promoting other concepts and expedients, for the same managers and programmers who had bought CASE earlier. Like the primitives, the software bureaucrats are not perturbed by the failure of one magic system. Their belief in software magic unshaken, they are trying now other systems. And no one will get fired for buying those systems either.

2

Partial or total software failures are such a common spectacle that they are now taken for granted. Actually, we no longer think of these occurrences as failures: the failing application, or utility, or methodology, or theory, or development system is accepted anyway, or else is abandoned and another one tried, and all this is seen as a normal software-related activity. Anyone close to the world of "information technology" is familiar with disappointing projects – requirements that cannot be satisfied, applications that cannot be kept up to date, projects that take far longer and cost far more than anticipated – but these situations are not seen as failures. New software products are installed every year in millions of places without being seriously used – presumably because they are not, in fact, the "solutions" they were said to be – but they are not seen as failures either. Nor are seen as failures those software projects that were approved on the promise of increased profits, or savings, or efficiency, or productivity, or strategic advantages, or return on investment, when these promises are not even remembered, much less verified, years later.

More and more organizations are facing software-related problems, but no one is referring to them as software failures. If the problem is discussed at all, the next software change is described as an upgrade, or as a migration, or as deploying a more advanced application, or as rearchitecting the system, or as strategic business transformation, or simply as investing in technology. For obvious reasons, it is in no one's interest to dwell on a failure; besides, neither the programmers, nor the managers, nor the consultants, nor the software companies appear to be responsible for it. So the whole affair is quietly forgotten, and the next project is initiated.

The only events reported in the media are the "success stories": isolated situations carefully selected from the thousands available, and described so as to create in the public eye a bright image of the world of computers and software. In reality we may well have a hundred failures for each success, but one wouldn't think so from reading business and computer periodicals. Only when the loss is in the tens or hundreds of millions of dollars is a software failure likely to be made public. And even then, no one is seriously reproved: after a brief investigation – conducted largely for the sake of appearances – the failure is forgotten.

❖

We are *surrounded* by software failures, but the software propaganda has succeeded in convincing us that these are not failures but normal occurrences. Thus, since they are not failures, no one is to be blamed. Software is designed and programmed by people, is sold and bought by people, is installed and used by people, but when it fails no one is to be blamed. In most cases we know the actual individuals involved in its development, purchase, or installation; but we don't feel that these individuals must be reprimanded, that they are accountable for their work in the same way that physicians, pilots, or engineers are for theirs.

In other professions we have the notions of incompetence, negligence, and malpractice to describe performance levels that fall below expectations. In software-related matters, and particularly in programming activities, these notions do not exist. In other professions we have created codes of expertise based on high performance levels – levels usually attained only after much training and practice; and we have associations safeguarding these codes, watching over the practitioners' conduct, and issuing and revoking licences. No one expects equivalent codes and licences for software practitioners.[3]

For most products, we expect solid warranties regarding quality and

[3] Some attempts *have* been made to establish codes for programming expertise. What these codes promote, though, is not the difficult skills required to solve real problems, but the trivial skills required to use the programming aids prescribed by the software elites. And the purpose of these aids, we saw, is to act as *substitutes* for programming expertise. Such codes, therefore, while appearing to promote professionalism, are in reality a fraud: they are invented by the elites as part of their effort to *prevent* expertise, and to create a dependence on development systems which they control. These are codes befitting a software bureaucracy.

performance; for software, the only warranty we get is that the bytes in the box will be read successfully into our computer. Instead of warranty we get *disclaimers*, and we sign a software agreement which specifies that, regardless of the product's performance, we alone are responsible for the consequences of its use. When other products do not work as promised, we are outraged, complain, return them, and even consider a lawsuit. When software products have deficiencies, or fail to provide the promised "solutions," we gladly expend time and effort dealing with their shortcomings, pay for "technical support," and look forward to the next version – which, we are told, will solve the problems created by the present one.

What is extraordinary, again, is not so much the incompetence and irresponsibility of the software practitioners, as the apathy of the rest of society: our acceptance, our belief that what they are doing represents the utmost that human beings can accomplish in the domain of programming.

If this incompetence and irresponsibility are found today mainly in programming, it is safe to predict that, as our dependence on software is growing, the same incompetence and irresponsibility, and the same tolerance and apathy, will spread into other fields. At issue, therefore, is more than just the current exploitation of society by the software bureaucrats. The corruption we are witnessing today in only one field may be the reflection of a trend that is affecting the entire society, so the study of software irresponsibility may reveal important truths about ourselves and our social future.

Determinism versus Responsibility

1

The idea of responsibility, and the closely related ideas of free will and determinism, are subjects studied by that branch of philosophy known as ethics, or moral philosophy. The *conflict* between free will and determinism, in particular, has troubled thinkers for centuries, and is still being debated.

The problem is simply this: Are human actions, choices, and decisions free, or are they causally determined? Usually, we feel that what we did on a certain occasion was the result of our own volition, that we could have acted differently had we chosen to. This perception is perhaps gratifying, but it is hardly sufficient when our acts affect other persons – when what we do has moral or legal significance. We need, therefore, a dependable method of determining, for a given act, whether it was performed freely or not. When a person performs a wrongful act, must he be blamed? Can we be sure that he could have acted differently and chose the wrongful act freely? And conversely, are a person's right acts always worthy of praise? Can we be sure that he could have acted differently and made the right choice freely?

It is obvious, then, why the idea of responsibility is related to the conflict between free will and determinism: If we believe in free will, we admit that we are free to conduct ourselves according to our own will; so we must be held responsible for our acts. If, however, we believe in determinism, then our acts, like any event in the universe, are the result of previous events, over which we may have no control; so we must not be held responsible.

Determinism is the thesis that the future state of a system can be determined from its present state, because all events are necessarily caused by some other events. And a belief in mechanism entails a belief in determinism. The mechanistic doctrine is founded upon the principles of reductionism and atomism: it maintains that a complex phenomenon can be seen as the result of simpler phenomena, which in their turn can be broken down into even simpler ones, and so on, reaching ultimately some elementary phenomena that can be understood intuitively. Mechanism, thus, holds that everything can be explained: from phenomena involving objects and physical processes, to phenomena involving human beings and mental processes.

Every entity or event in the present is a result of past phenomena, so mechanism assures us that from a knowledge of the past we can explain fully and precisely the present. Similarly, since present phenomena are the cause of future ones, a knowledge of the present should allow us to predict fully and precisely every entity and event in the future. This fantastic claim is the essence of determinism, and a *necessary consequence* of mechanism. To put it differently, a person who believes in mechanism cannot believe at the same time in *indeterminism* without contradicting himself.

The best-known expression of determinism is the one formulated by Laplace some two centuries ago: The whole universe, and every occurrence in the universe, can be seen as a system of particles of matter acting upon one another according to Newton's law of gravitation. Thus, an infinitely intelligent being, capable of observing the state of all the particles at a particular instant and capable also of solving the pertinent equations, could determine their state – and hence the state of the universe – for any other instant in the past and in the future. This being (known as Laplace's demon) could, therefore, predict every future occurrence in the universe.

Note that it is irrelevant whether such a being exists; only the *idea* matters. The mechanists merely claim that it is possible to reduce to particle mechanics all physical, chemical, biological, psychological, and social phenomena, and hence omniscience – complete understanding of the past and present, and complete knowledge of the future – is within our reach. The mechanists admit that we are far from having achieved this reduction. Still, they say, we are making progress, and one day we may well become omniscient. This too, however, is irrelevant: even if human beings are incapable of omniscience, the mechanists say, and even if no omniscient beings exist anywhere, this does not alter the fact that the future can be completely determined from the present. The particles and the law of gravitation continue to exist even if Laplace's demon is no more than an idea.

Thus, the mechanists conclude, even if no real or hypothetical being can know everything in the world, it is still true that the world can be explained; it is still true that the state of the universe at any instant is the result of earlier states, that every occurrence is caused by previous occurrences. And, since we are part of the universe, it must also be true that every decision and action displayed by human beings is the result of past occurrences. But most occurrences in the universe are beyond our control; so perhaps all our thoughts, feelings, and acts are due in fact to external causes, not our own volition. The idea of free will is therefore an illusion. And it is not only the possibility of free will that is an illusion, but also the possibility of original deeds, and hence of individual responsibility. Human beings are just mechanical parts in the great machine that is the universe, and all aspects of human existence can be explained and predicted, in principle, with the same equations that describe the motion of bits of matter.

This view, fallacious and distasteful as it is, has dominated science and philosophy since the seventeenth century. The mechanistic doctrine has been a complete failure in the human sciences, where it cannot explain even the simplest phenomena, let alone thoughts and feelings. But its spectacular success in the natural sciences has inspired scientists and philosophers with confidence, and few doubt, even today, that reductionism and atomism will eventually prove to be just as successful in explaining *human* phenomena. Thus, even though mechanism has been shown to work only in a narrow range of physical phenomena, researchers in *all* disciplines see it as their professional duty to adhere faithfully to the mechanistic doctrine. At the same time, because determinism creates such a demeaning view of humanity, many scientists and philosophers are now reluctant to call themselves determinists; and some even deny that they are mechanists.

As I have pointed out, today's mechanistic theories only *appear* less naive than the earlier ones (see pp. 76–78). In reality, current academic research is grounded, just like seventeenth-century science, on the assumption that all phenomena can be reduced to particle mechanics. The theories have not changed, and are as fallacious as ever; but because determinism is no longer fashionable, the bold and straightforward claims of earlier times have been replaced with a mass of sophisticated rhetoric.

Today's mechanists are in a difficult position. They want to give up the idea of determinism, but to continue to practise mechanism. Mechanism, however, entails determinism, so the mechanists are caught in a self-contradiction. The only way out is to claim that mechanism can be reconciled with indeterminism, that we can have both. And indeed, countless theories have been advanced in the last one hundred years in an attempt to show that mechanism is, in fact, compatible with indeterminism. This notion is nonsensical, of course, but in a culture where so many absurdities have already been claimed in the name of mechanism, one more absurdity makes little difference. This is how Karl Popper puts it: "I personally find Laplacean determinism a most unconvincing and unattractive view.... But it is, perhaps, worth stressing that Laplace does draw the correct conclusions from his idea of a causally closed and deterministic [world]. If we accept Laplace's view, then we must not argue (as many philosophers do) that we are nevertheless endowed with genuine human freedom and creativity."[1]

2

The delusion of determinism is easy to understand if we recall the concept of simple and complex structures. All phenomena are in reality non-mechanistic, the result of many interacting processes; so they can be represented accurately only with *complex* structures. Still, some phenomena can be *approximated* with simple structures. If a phenomenon requires a system of interacting structures, we can usefully approximate it with one structure when the links between structures are weak enough to be ignored. But a simple structure provides both mechanistic and deterministic qualities; consequently, a phenomenon seen as mechanistic will be seen at the same time as deterministic. The determinism of a phenomenon, though, just like its mechanism, is an illusion, an *approximation* of reality. We find a deterministic approximation useful, presumably, for the same phenomena for which we find a mechanistic approximation useful.

We have discovered useful mechanistic approximations for many phenomena – physical ones, in particular; but these phenomena form only a small part of our world. The most important biological processes, and practically all mental and social processes, are complex phenomena; and for them no useful mechanistic representations have been found. Even some physical phenomena are only poorly approximated by mechanistic models (the apparently simple three-body system, for instance, see pp. 107–108). And, most

[1] Karl R. Popper, *The Open Universe: An Argument for Indeterminism* (London: Routledge, 1988), p. 124.

surprisingly perhaps, mechanism has not kept up with the discoveries of modern physics in the field of *elementary* particles, precisely where one would expect reductionism and atomism to work best.

Mechanistic approximation, thus, fails at both low and high levels of complexity: for elementary physical phenomena as much as mental and social phenomena. Ultimately, only a narrow range of phenomena are amenable to mechanistic approximation. So, if the others are non-mechanistic, they must also be indeterministic.

❖

Recall the theories we studied in chapter 3, and their consequences. Those scientists claimed that all human phenomena can be explained with precision, so human beings are in reality deterministic systems. Scientists today are still working on this type of theories; but because the academic fashion has changed, we don't hear the claim that human beings are deterministic systems as often as we did in the past.

A recapitulation of these delusions will help us to understand how they affect our conception of individual responsibility. What I want to show is that, by degrading the concept of indeterminism, the theories of mind – and now also the theories of software and programming – are still claiming, in effect, that human beings are deterministic systems. The conclusion that we are not responsible for our acts follows then logically. Thus, there is no real difference between a theory that *denies* the existence of free will and creativity, as the older theories did, and one that *accepts* these qualities but *redefines* them to match a mechanistic model.

Minds and societies give rise to indeterministic phenomena, so mechanistic theories cannot explain them. The faith in mechanism, however, prevents the scientists from recognizing these failures as falsifications. So, instead of doubting and severely testing their theories, they end up *defending* them. The only way to defend a fallacious theory is by turning it into a pseudoscience: tests are restricted to situations that confirm it; and the theory is repeatedly expanded, by incorporating the falsifying instances and making them look like new features. Eventually, the theory becomes unfalsifiable, and hence worthless. But the scientists do not consider this activity to be dishonest, or unprofessional. On the contrary: because the principles of reductionism and atomism are accepted implicitly as the universal method of science, defending a mechanistic theory is perceived as important scientific work.

Then, even though their theories keep failing, the scientists draw demeaning conclusions about human beings and human societies – conclusions that would be warranted only if their theories were successful. They conclude, in particular, that all manifestations of human knowledge and behaviour can be deduced from some innate faculties, which constitute a sort of alphabet of human capabilities; so the knowledge and behaviour displayed by each one of us can be described with precision as a function of these innate faculties, just as the operation of a machine can be described as a function of its parts.

Mechanistic theories of mind and society, thus, *inevitably* lead to the view that human beings are deterministic systems: we have no control over our innate capabilities, of course; and if, in addition, our knowledge and behaviour can be precisely deduced from these capabilities, then we can have no greater control over our decisions and actions than machines have over theirs. It is impossible to separate mechanism from determinism.

But, to adhere to the current academic fashion, the scientists claim that mechanistic theories do *not* preclude indeterminism in human affairs. Specifically, they claim that it is possible to explain mathematically all human acts, and all aspects of a human society, without denying free will and creativity. They avoid the contradiction by *redefining* these concepts: from the absolute qualities we take them to be, to some relative qualities, which match their theories. They continue to use the terms "freedom" and "creativity," but instead of *unpredictable* acts, the terms mean now just the freedom and creativity to select any act from a range of known alternatives. Still, the scientists argue, if the number of alternatives is sufficiently large, the new definition does not deny indeterminism. *Their fallacy is to misinterpret indeterminism as a large number of alternatives.* (See also the related discussion in chapter 3, pp. 281–284.)

❖

Mechanistic models are logically equivalent to simple hierarchical structures, as we know. Mechanistic theories, therefore, depict human phenomena as simple structures. Depending on the theory, the starting elements are various bits of knowledge or behaviour, or various propensities, while the values of the top element are the many alternatives displayed by a mind or by a society: all possible knowledge, behaviour patterns, language uses, social customs, and so forth. The mechanistic *software* theories, for their part, depict as simple structures *software* phenomena – the development and use of software. The starting elements are various software entities and various bits of software-related knowledge, while the values of the top element are all the alternatives possible for software applications and their use.

Now, it is quite easy to design a mechanistic model that displays an infinite number of alternatives. And the scientists misinterpret this trivial quality of mechanistic models as indeterminism: they believe that these alternatives are the same as the infinity of alternatives that make up the actual, complex phenomenon. So, they conclude, a model has been discovered that is both mechanistic and indeterministic.

Mechanistic theories, thus, assume that human minds and societies are akin to devices, in that they can be in a number of states, or can generate a number of alternatives – states and alternatives that we can account for. A device like a die, for example, behaves unpredictably when rolling; but we know that it will display, when stopped, one of six given symbols. Human phenomena are thought to be essentially the same, except for displaying many more alternatives, perhaps even an infinite number of alternatives. Let us use the analogy of the die to examine this fallacy.

A rolling die constitutes a complex phenomenon, the interaction of several phenomena; this interaction is what determines its movement and, consequently, which symbol will be displayed when it stops. The six alternatives form, in fact, only a small part of the actual phenomenon. To describe the complex phenomenon, we would have to take into account many details of the die itself, its movement when rolling, and its environment: its size, its material and weight, the shape of its edges and corners, its starting orientation in space, the form and texture of the surfaces it touches, how variables like the ambient temperature and humidity affect its movement, and the direction and magnitude of the force that sets it in motion. Also, the final state will be, not just the symbol displayed, but the exact position and orientation of the die in space when the rolling stops; that is, not one of six, but one of an infinite number of alternatives. The actual phenomenon, thus, consists of a large number of factors, which can display an infinite number of values, and lead to an infinite number of possible results.

It is not this infinity of alternatives that prevents us from predicting the final state, though, but the fact that the phenomenon is non-mechanistic: we cannot describe with precision how the final state depends on the various factors. Still, nothing stops us from representing the phenomenon with a mechanistic model: all we have to do is depict it with a simple structure where the starting elements are the six symbols, and the top element is the symbol displayed when the die stops rolling. The phenomenon can then be said to be both mechanistic (because we can account for all possible values of the top element from a knowledge of the starting elements) and indeterministic (because we don't know which symbol will actually be displayed).

What we did, obviously, is *simplify* the phenomenon: our model is an *approximation* of the actual die phenomenon. Rather than a complicated movement and a final state determined by the interaction of many factors, we described the phenomenon simply as the selection of one symbol out of six. And "indeterminism" is now, not the uncertainty of the actual phenomenon, but the uncertainty of not knowing which symbol will be selected. In other words, if we degrade the notion of indeterminism to mean just the uncertainty of not knowing which alternative will be selected from a known range of alternatives, we can claim that the phenomenon of the die is both mechanistic and indeterministic.

Scientists attempt to represent minds and societies with mechanistic models because they believe the indeterminism of human phenomena to be like the indeterminism of the *simplified* die phenomenon. The only difference, they say, is the larger number of alternatives. The indeterminism of human phenomena, however, is like the indeterminism of the *actual* die phenomenon. Minds and societies give rise to complex phenomena, and hence to *true* indeterminism, not the weak indeterminism displayed by a mechanistic model.

We can perhaps delude ourselves that we understand the die phenomenon when we depict it as the selection of one symbol out of six; but this delusion comes to an end when we study the great complexity of the *actual* phenomenon. Clearly, the unpredictability of the final state of the die as a result of all those factors is of a higher level than the unpredictability of selecting one symbol out of six. Thus, the simplified, mechanistic model is useless if what we need is to describe the relationship between those factors and the die's final state.

Similarly, the scientists keep simplifying the complex structure that is a human phenomenon until they manage to depict it with a mechanistic model. But this approximation is too crude to be useful, because it can display only a fraction of the alternatives that make up the actual phenomenon. The scientists stop their simplification at a point where there is still enough indeterminism left to make the mechanistic model somewhat unpredictable; and they misinterpret this unpredictability – which is just the trivial process of selecting one alternative out of many – as the indeterminism of the original, complex phenomenon.

In our analogy, the actual die phenomenon stands for a human phenomenon, and the simplified model stands for the mechanistic model of the human phenomenon. But unlike the six alternatives of the die, the mechanistic model of the human phenomenon, even though a simplified version of the actual phenomenon, still displays an infinite number of alternatives. And this is the source of the confusion. For, the infinity of alternatives found in the mechanistic model is only a fraction of the infinity found in the actual, complex phenomenon; and consequently, the indeterminism of the mechanistic model is only a weak version of the indeterminism displayed by the actual phenomenon. The number is infinite in both cases, but we are witnessing in fact different *kinds* of phenomena: one can be represented mathematically, while the other cannot.

It is the infinity of alternatives displayed by their mechanistic models, therefore, that confuses the scientists: they mistakenly conclude that, being infinite, these must be all possible alternatives, so their models can represent human phenomena.

Mechanistic models fail because they are incapable of displaying the same indeterminism as the actual phenomena. True indeterminism is, of course, the highest form of indeterminism. It is only by choosing a weaker definition that the scientists manage to contrive models that are both mechanistic and indeterministic. But if their goal is to explain human phenomena, degrading the concept of indeterminism cannot help them: all they can discover then is theories that do not work, models that do not approximate the human phenomena closely enough to be useful.

In conclusion, the difference between true indeterminism and its weaker version is not as mysterious as it appears. When attempting to represent a complex human phenomenon with a simple structure, the scientists are committing the two mechanistic fallacies, abstraction and reification: they start from levels that are too high, and they ignore the links with the *other* structures that are part of the phenomenon. Their model, as a result, cannot account for all the alternatives: missing are those caused by interactions at levels lower than the level of the starting elements. The difference between the alternatives generated when starting from low-level physiological elements, and those generated by starting from the higher-level elements, is the difference between true indeterminism and its weaker version.

3

Just like the mechanistic theories of mind and society, the mechanistic *software* theories praise creativity while degrading this concept to match the weak version of indeterminism. Creativity in software-related matters, according to these theories, does not mean the utmost that human minds can accomplish, but merely the selection and combination of bits of knowledge within a predetermined range of alternatives.

Software mechanism holds that software-related phenomena can be represented as simple structures, and that it is possible to account for all the values of the top element in these structures. These values are, in effect, all the applications that can be implemented with software, and all aspects of software use. Software mechanism claims, thus, that it is possible to account for all the alternatives displayed by human minds when engaged in software-related activities. Accordingly, it should be possible to identify the elements that lie just a few levels below the top element. It is these elements, clearly, that give rise to all the alternatives, so there is no need to deal with lower-level ones. We should treat *these* elements as starting elements, and incorporate them in software devices. Then, by operating such a device, anyone will be able to generate all the values of the top element – all conceivable alternatives in software-related phenomena – simply by selecting and combining built-in elements.

What this means in practice is that we should be able to replace the knowledge and experience needed to create software applications, with the simple skills needed to operate software devices; that is, the skills needed to make selections. Software devices, thus, materialize the belief that what an experienced mind does is merely select and combine alternatives, so the indeterminism we observe in software-related phenomena is just the uncertainty caused by the large number of alternatives.

This interpretation also explains why the software devices provide their operations in the form of selections (menus, buttons, lists, etc.), and selections within selections, instead of allowing us to use them freely – as we use natural languages, or traditional programming languages. Their designers, obviously, are attempting to implement a simple hierarchical structure: the structure of knowledge that, according to their theories, exists in the mind of an experienced person.

So there is no longer a need for each one of us to develop structures in our mind starting from low-level bits of knowledge. Since the operations provided by devices can directly replace high-level knowledge elements, by combining these operations we will be able to generate any one of the knowledge alternatives that human minds can display (or, at least, the important alternatives) without having to possess that knowledge ourselves. The easy skill of selecting and combining those operations, the software mechanists assure us, can be a substitute for the complex knowledge developed in a mind by starting from low levels. The only thing that human beings need to know from now on is how to select and combine operations within the range of alternatives provided by software devices.

Like the mind mechanists, the software mechanists commit the two fallacies, reification and abstraction: they take into account only one structure, ignoring the other relations that exist between the same elements; and they start from levels that are too high. As a result, they lose the alternatives arising from the interactions occurring at levels lower than their starting elements. They note the infinity of alternatives possible even when starting from higher levels, and conclude that, since their software devices can generate these alternatives, they have attained their goal: a mechanistic model that can emulate indeterministic phenomena. Their infinity, however, is only a fraction of the infinity of alternatives found in the actual, complex phenomenon.

By restricting ourselves to simple structures, we are becoming a closed, deterministic society, where only certain alternatives can exist. The danger posed by our software ideology, therefore, is not just the loss of alternatives in software-related matters, but the degradation of minds. Our non-mechanistic capabilities do not simply exist – they develop; and they can develop only when we are exposed to low-level elements, because this is the only way to create all possible alternatives in our minds. If we restrict ourselves to mechanistic knowledge – to simple knowledge structures and high-level starting elements – our minds cannot develop above the intellectual level of machines.

❖

Why is it so easy for the software elites to convince us that our minds are inferior to their devices? Our willingness to accept this notion, despite its obvious fallaciousness, may well be a symptom, an indication of how advanced our mental degradation *already* is. We must use our minds to judge the usefulness of software devices, but it is these very minds that remain undeveloped when we agree to depend on devices. So, if we trust the elites and get to depend on their devices, we are *bound* to lose our ability to recognize how useless these devices actually are. We will believe that the devices are superior to our minds, so we will continue to depend on them; our minds will then be further degraded, in a process that feeds on itself.

This collective mental degradation is an amazing spectacle. We are willingly renouncing our natural mental capabilities, and our responsibility as individuals, and replacing them with a dependence on devices and on the elites behind them. We are degrading our status from creative and responsible individuals to operators of devices. This process, which may well be irreversible, is the ultimate consequence of our mechanistic culture: when we agree to limit ourselves to mechanistic performance, and hence to a fraction of our mental capacity, what we do in effect is fulfil the wish of those scientists who have been telling us for a long time that our minds are nothing but complicated machines.

4

We must not be confused by words. No matter what the mechanists say, their theories necessarily lead to the view that there is no indeterminism in human phenomena. The scientists, and now also the software theorists, may use words like "indeterminism," "freedom," and "creativity," but they are degrading these concepts to mean the selection of acts from a known range of alternatives. This is how they manage to have mechanism and indeterminism at the same time.

Whether their theories describe minds and societies, or software development and use, if they claim that it is possible to account for all the alternatives in advance then what they are promoting is *deterministic* theories; and if they attempt to represent minds and societies with mechanistic models, or to replace knowledge with software devices, then what they are saying is that human beings are deterministic systems.

If the mechanists believe that we are deterministic systems, their conclusion *must* be that we are not free agents, and that we cannot be held responsible for our acts. The theories of mind and society manage to avoid this conclusion by maintaining that it is possible to have both mechanism and indeterminism. And, since these theories do not work anyway, few notice the absurdity of the claim. We didn't change the way we think or behave or speak, to match the mechanistic theories; so the fact that they are self-contradictory can be overlooked. The *software* theories make claims similar to the traditional mechanistic claims, but now we *are* modifying our conception of intelligence and creativity: we are mutating into deterministic beings.

I remarked at the beginning of this chapter that the incompetence and irresponsibility displayed by the software practitioners, and the tolerance and apathy of the rest of us – our belief that they need not be accountable for their deeds as other professionals are for theirs – constitute an extraordinary phenomenon. We are now in a position to explain this phenomenon.

These attitudes are a natural consequence of the mechanistic software culture. It is impossible to have at the same time a mechanistic, and hence deterministic, culture *and* responsibility. If people are seen as deterministic systems, we cannot expect them to possess non-mechanistic knowledge. So, if all they can do is follow theories and methodologies, or operate software devices, we must limit their accountability to the performance of these acts; we cannot hold them responsible for the *results*.

Increasingly, as programmers and as users, we depend on software devices that are based on mechanistic theories. And when we agree to depend on these devices, we agree in effect to modify our conception of human intelligence to correspond to the mechanistic dogma. We agree, in other words, to forgo our non-mechanistic capabilities: we restrict ourselves to mechanistic knowledge, and thereby keep our performance at the level of novices, or machines. This, clearly, is a new development, a new manifestation of the mechanistic myth.

The traditional mechanistic theories tried to explain and predict human acts; and they failed, because we ignored them and remained indeterministic systems. The software theories, when viewed as theories of human capabilities, appear to work; but this is because we have now agreed to restrict our performance to mechanistic levels. Thus, the reason why irresponsibility is especially noticeable in software-related activities is that, in our capacity as programmers or software users, our transformation from indeterministic to deterministic systems is almost complete. In our software pursuits, we have renounced our non-mechanistic capabilities; so the world of software constitutes, in effect, a social system founded upon mechanistic principles. In this system, people have been absolved from accountability for the consequences of their acts, and the idea of responsibility has been degraded to mean simply adherence to the official ideology. Attitudes that used to exist only in fictional societies, or in totalitarian societies, can now be found in our software pursuits.

Today we can observe this phenomenon mainly in our software-related affairs – programming, in particular – because the mechanistic software theories affect our software pursuits directly. But we should expect to see the same irresponsibility and the same acceptance emerge in other domains, as these domains become dependent on mechanistic software notions. We cannot retain the concept of individual responsibility in a society where we consider individuals to be deterministic systems. Indeterminism is associated with free will, and hence responsibility; so determinism *logically* entails irresponsibility.

5

As our dependence on software is growing, our society is becoming increasingly totalitarian; and this progression can be observed in our attitude toward individual responsibility. Under totalitarianism, the responsibility of the individual is redefined to mean the responsibility of obeying the elite. Terms like "freedom," "creativity," "expertise," and "intelligence" are still used, but their meaning is degraded: individuals are expected to possess these qualities, but the qualities mean now merely the selection of certain acts from a predetermined range of alternatives. Whereas in their true sense the terms describe *absolute* values – the ability to select *any* alternative that can be conceived by human minds – under totalitarianism the terms describe *relative* values: the freedom to perform any activity sanctioned by the elite, the creativity to accomplish a task with the means supplied by the elite, expertise in the kind of knowledge taught by the elite, and the intelligence to appreciate the totalitarian ideology.

Note how similar these concepts are to the mechanistic *academic* ideology: the theories of mind claim that intelligence, creativity, and expertise mean the selection of acts from a predetermined range of alternatives – the range for which human minds are biologically wired; and the responsibility of scientists is redefined to mean the obligation of pursuing only this type of theories. Note also how similar these concepts are to the mechanistic *software* ideology: expertise in programming, and in software-related activities generally, has been redefined to mean familiarity with the mechanistic software theories, and with the development tools provided by the elite; and responsibility means simply the challenge of keeping up with these theories and tools.

But this is no coincidence. Recall the inevitable progression – mechanism, scientism, utopianism, totalitarianism. Mechanistic beliefs lead to scientistic theories – mechanistic theories of mind and society – which then lead to the utopian vision of a perfect society. The perfect society must be founded upon strict mechanistic principles, but human beings are currently non-mechanistic systems, undisciplined and unpredictable. They must be coerced, therefore, in the name of progress, to abandon the old-fashioned ideas of freedom and creativity, to admit that they are merely small parts in a great deterministic system, and to restrict themselves to mechanistic performance. It *is* possible to create a perfect society, but, unfortunately, only through totalitarianism. This is a passing phase, though. The coercion is only necessary until everyone gets to appreciate the benefits of mechanistic thinking; that is, until every person becomes a deterministic system, an automaton.

Mechanism and determinism are logically related to irresponsibility, as we saw. Thus, one way of judging how close a society is to being "perfect" is by studying its conception of individual responsibility: how people think of themselves, of their rights and responsibilities, is an indication of the stage their society has reached in the progression from a human system to a mechanistic one. What we notice is a shift: from the idea that we are directly responsible for our acts as individuals and as professionals, to the idea that we are only responsible for conforming to a certain ideology, while its validity or morality need not concern us. If it is mechanistic, and hence "scientific," the ideology is believed to embody the absolute truth.

Under Nazism, for example, countless individuals were involved in atrocious crimes, or were watching passively as crimes were being committed all around them, convinced that they were serving their country. By modifying the idea of responsibility, perfectly normal people – people with family and friends, and an appreciation of life, art, and logic – were induced to do almost anything. For an entire society, the idea of responsibility was redefined in just a few years to mean, simply, adherence to the ideology established by their leaders. Pseudoscientific theories on social, racial, and political matters, invented by a small elite, became unquestionable principles. People no longer saw themselves as independent human beings, but as parts of a great mechanistic system, a great historical process that was as inevitable as evolution itself.

When the mechanistic dogma is accepted implicitly, the shift in the meaning of responsibility follows logically. If we believe in mechanistic social notions – if, in other words, we hold certain developments to be historically inevitable – we cannot at the same time consider ourselves to be free agents. All our thoughts and acts must then conform to these notions, and any doubts seem as absurd as doubting the laws of nature. The way mechanism engenders irresponsibility, then, is by assuming that individuals and societies are parts of a great deterministic system, so everything can be explained precisely and completely – in principle, at least. Our common-sense conceptions of free will, intelligence, creativity, and responsibility are therefore mere illusions. In reality, we *cannot help* thinking, feeling, and acting as we do. Our knowledge, beliefs, and capabilities are determined largely by forces beyond our control. Although we do not yet understand that great system of which we are part, there can be no doubt that it is deterministic; so the conclusion that we are not really free, and hence cannot be held responsible for our acts, remains valid.

6

Nazism is only an extreme example of the progression of a society from liberalism to mechanistic thinking and irresponsibility. Because mechanistic theories of mind end up degrading the notion of responsibility, we can observe this progression, in a milder form, in our own society. Recall the mechanistic delusions we studied in chapter 3. While society has yet to reach the level of degradation depicted by these theories, we can already observe it in the attitude of the scientists themselves. They have degraded the definition of scientific responsibility, from discovering sound and useful theories, to discovering *mechanistic* theories.

Then, since what matters to them is not whether the theory is valid but whether it is mechanistic, the scientists draw from their *failing* theories the kind of conclusions that would be warranted only if the theories were successful. Specifically, they claim that human beings are nothing but complicated machines; that human acts are a function of innate propensities and can be explained with mathematical precision; and that, based on these facts, it is possible to design and implement a perfect society.

Thus, in addition to turning their disciplines into pseudosciences, the scientists promote theories that lead to totalitarianism. But they do it out of a sense of duty: their responsibility, they say, is to promote science, and science means mechanism. They perceive themselves as pioneers, as experts who understand what the rest of us cannot yet see, and who alone have the courage to accept the implications of these discoveries; namely, to accept the fact that human beings are in reality automatons. So it is their duty to enforce these ideas upon us, the ignorant masses. But there is nothing wrong in this, the scientists say; all they are doing is helping to speed up a process of social evolution that was going to take place anyway.

Since mechanism has distorted their own sense of responsibility, it is not surprising that their vision of the perfect society is a deterministic social system where everyone has been degraded just as they have degraded themselves. The only reason we have not yet been turned into automatons is that these scientists lack the power to implement their vision. But it is important to study their mechanistic obsessions and totalitarian attitudes, because the same obsessions and attitudes are now being displayed by our *software* elites; and these elites do have the power to put their ideas into practice – through software concepts and devices.

❖

Recall the linguistic theories we studied in chapters 3 and 4. Discussing the same theories, Roy Harris[2] traces their evolution over the last three centuries, and notes that their foundation on the mechanistic dogma has turned the study of language and mind into a system of belief, a mythology. The current theories, which are merely the latest in our tradition of mechanistic language delusions, claim that human beings have an innate language faculty and that this faculty can be explained with deterministic models. What the mind does when we communicate through language, then, must be akin to what a machine does. So we can view the language faculty as a sort of language machine that runs in the mind.

Because they are founded on mechanistic delusions, our theories end up resembling the mystical conceptions of language held by earlier civilizations. "But no other civilization than ours," Harris observes, "has envisaged language as the product of mysterious inner machinery, run by programs over which human beings have no control. That, it will be said, is just the mythology one might expect of a computer-age society; and so it is.... What is significant is that the new view of language promoted is not a conceptual enrichment of what preceded, but a conceptual impoverishment. A society whose academic establishment accepts with alacrity and even with enthusiasm the prospect of being able to treat verbal communication as a complex form of data-processing is a society which proclaims its linguistic immaturity."[3]

This mechanistic model of mind, Harris continues, leads *unavoidably* to the conclusion that human beings are not free and responsible agents. The ultimate consequence of language mechanism, thus, is not the promotion of invalid linguistic theories, but the undermining of the idea of individual responsibility. Because language fulfils such an important function in society, if we believe

[2] Roy Harris, *The Language Machine* (Ithaca, NY: Cornell University Press, 1987).

[3] Ibid., pp. 171–172.

that we have little control over what we say – if we feel that all we do in reality is operate a language machine – we will necessarily conclude that we cannot be held responsible for any acts involving language. Our speech, our knowledge, our decisions, our social and professional performance, are then largely the result of factors beyond our control.

Such conclusions, like the language theories themselves, "are aberrations which the myth of the language machine unavoidably promotes. Unavoidably, because the workings of the machine are envisaged as totally independent of any criteria of values entertained by the machine's human operators. Thus a form of discourse about language is created which serves either to disengage language from human motives and intentions, or to disguise the extent and nature of that engagement.... The myth of the language machine is a convenient myth because it absolves us from our day-to-day duties as language-makers, and blankets out for us all awkward questions concerning the exercise of authority through language. We purchase this linguistic security at cost price: and the cost is the removal of language from the domain of social morality altogether."[4]

We have already determined the similarity of language and software, so the preceding arguments can also be read by substituting software for language. What we note then is that this analysis describes our software culture even more accurately than it does our conception of language. Our software theories are degrading the notion of individual responsibility just like the language theories. Like the myth of the language machine, the software myth "serves either to disengage [software] from human motives and intentions, or to disguise the extent and nature of that engagement." It "absolves us from our day-to-day duties as [software] makers, and blankets out for us all awkward questions concerning the exercise of authority through [software]." As in the case of language, human beings are perceived as mere operators of software machines; so our responsibility is limited to knowing how to operate these machines. The cost of this security is "the removal of [software] from the domain of social morality altogether."

Unlike the language machine believed to run in the mind, though, the software machines are real: these are the devices supplied by the software elites. Thus, while our degradation through language is a relatively slow process, the same degradation is taking place, at a much faster rate, through software. Since we have agreed to depend on software in everything we do, and have also agreed to restrict ourselves to the kind of applications that can be created with software devices, we are asserting in effect that our knowledge, our capabilities, and our performance are the result of factors beyond our control.

[4] Ibid., p. 162.

Isaiah Berlin[5] shows that, despite their differences, most social and political theories are founded on the premise that social evolution is a deterministic process; that all historical developments are, in fact, inevitable and predictable; and that the only reason we cannot explain them is our limited knowledge.

Not surprisingly, these theories conclude that the ideas of free will and responsibility are illusions, reflections of our present ignorance. Science will allow us, one day, to control minds and societies as effectively as we control machines: "All these theories are, in one sense or another, forms of determinism, whether they be teleological, metaphysical, mechanistic, religious, aesthetic, or scientific. And one common characteristic of all such outlooks is the implication that the individual's freedom of choice … is ultimately an illusion, that the notion that human beings could have chosen otherwise than they did usually rests upon ignorance of facts.… The more we know, the farther the area of human freedom, and consequently of responsibility, is narrowed. For the omniscient being, who sees why nothing can be otherwise than as it is, the notions of responsibility or guilt, of right and wrong, are necessarily empty."[6]

To be as successful as are the theories of the natural sciences, the theories of mind and society would have to explain social phenomena through reductionism and atomism; specifically, they would have to show how social phenomena derive from the knowledge and acts of individual persons. Although they cannot discover such theories, the mechanists continue to believe in the possibility of exact explanations of social phenomena. So they claim that social evolution is brought about, not by people, but by *social forces*. While manifesting themselves through the acts of people, these forces have a character of their own, and are more important than the people themselves.

Depending on the theory, the forces may be described as social classes, political institutions, religions, cultures, economic conditions, or technological changes. All these concepts, however, serve in the end the same purpose: because the mechanists cannot explain social evolution in terms of individual persons, they attempt to explain it by replacing the persons with an abstract entity – a "whole." The theories appear then to explain the evolution, but this is an illusion, because the "whole" is merely an invention of the scientists. Those social forces remain unexplained, and their existence remains unproved. The forces are said to cause social evolution, but they are themselves described in terms of this evolution; so, being circular, the theories are fallacious.

[5] Isaiah Berlin, "Historical Inevitability," in *Four Essays on Liberty* (Oxford: Oxford University Press, 1969).

[6] Ibid., pp. 58–59.

All deterministic theories, thus, are alike: "Different though the tone of these forms of determinism may be … they agree in this: that the world has a direction and is governed by laws, and that the direction and the laws can in some degree be discovered by employing the proper techniques of investigation: and moreover that the working of these laws can only be grasped by those who realize that the lives, characters, and acts of individuals, both mental and physical, are governed by the larger 'wholes' to which they belong."[7]

If we believe in the existence of some great social forces that control our life, the necessary conclusion is that it is not we, the individuals, but those forces, that are responsible for what happens in the world: "What the variants of either of these attitudes entail, like all forms of genuine determinism, is the elimination of the notion of individual responsibility.… If the history of the world is due to the operation of identifiable forces other than, and little affected by, free human wills and free choices (whether these occur or not), then the proper explanation of what happens must be given in terms of the evolution of such forces. And there is then a tendency to say that not individuals, but these larger entities, are ultimately 'responsible.'"[8]

These theories are popular because they are comforting. It is reassuring to hear that we must not blame ourselves for failures, that our knowledge and accomplishments are in reality part of a master plan, so whatever we do is necessarily right: "The growth of knowledge brings with it relief from moral burdens, for if powers beyond and above us are at work, it is wild presumption to claim responsibility for their activity or blame ourselves for failing in it.… And so individual responsibility and the perception of the difference between right and wrong choices, between avoidable evil and misfortune, are mere symptoms, evidences of vanity, of our imperfect adjustment, of human inability to face the truth. The more we know, the greater the relief from the burden of choice.… We escape moral dilemmas by denying their reality; and, by directing our gaze towards the greater wholes, we make them responsible in our place. All we lose is an illusion, and with it the painful and superfluous emotions of guilt and remorse. Freedom notoriously involves responsibility, and it is for many spirits a source of welcome relief to lose the burden of both."[9]

Until recently, the transition from determinism to irresponsibility could be observed only in totalitarian societies and in the mechanistic theories concocted by academics. In our time, however, a third possibility has emerged: the software culture. The basic tenets of the deterministic *social* theories can now be found in our *software* theories: the blind faith in mechanism; the belief in historical inevitability; the idea of the "whole" – the mighty forces of progress that are beyond our control even though we are parts of them; and the

[7] Ibid., p. 62.

[8] Ibid., pp. 63–64.

[9] Ibid., pp. 77–81.

comfort of knowing that, in the final analysis, we are not really responsible for our acts, nor for our failures.

The mechanistic software theories started by affecting only the software practitioners, but they are now spreading into all software-related activities. We have already discussed these theories, and how they have degraded our notions of knowledge, freedom, and creativity. What I want to show here is that, while concerned with software, these theories belong in fact to the same category as the deterministic social theories we have just examined.

The software counterpart of the social forces that control the history of humanity are the forces of *software evolution*. Although we take part in this evolution (by performing software-related activities), only the software elites actually understand it. It is their task, therefore, to help us assimilate software changes; and they do it by inventing concepts and devices that embody the power of software while remaining simple enough for us to use.

Our role in the process of software evolution, then, is to adopt the latest software devices provided by the elites – the latest theories, methodologies, development environments, and applications. And we must not attempt to judge these innovations with our naive standards of good and bad, of success and failure. If, for example, a software device does not provide the promised benefits, or if it creates several new problems for each problem it solves, or if we have to alter the way we run our affairs in order to use it, or if it demands more of our time than we had planned, or if it makes us unduly dependent on the elites, we must not think of these issues as shortcomings, but as a manifestation of software evolution. We must not blame the device, nor its maker, nor ourselves.

Each version of a device is a newer thing, so it constitutes software progress. This evolution is historically inevitable, so we must conform to it even if we do not understand it. Founded as they are on mechanistic principles, the devices are expressions of software science, and their perpetual changes reflect the never-ending process of software evolution. It is ignorance and vanity that prompt us to doubt the innovations, and tempt us to pursue personal ideas. It is presumptuous to think that we can accomplish more with our own minds than by conforming to the forces of progress. The belief in skills or creativity in software-related matters is evidence of immaturity, of imperfect adjustment. The more we learn about software and programming, the further the area of individual choice, and hence of responsibility, is narrowed. Whether programmers or users, we are parts of a "whole," of a great historical process that is beyond our control. All we can reasonably be expected to do is adapt to it, by accepting the devices provided by the elites.

This is what we are told, but the existence of the forces of software evolution, like the existence of the *social* forces of evolution, was never proved. The generations of software devices are said to be a reflection of this evolution, but software evolution is explained in terms of generations of devices. So, just like the argument for social evolution, the argument for software evolution is circular, and hence fallacious.

Totalitarian systems like Nazism and Communism were said to be an expression of the great forces of social progress; so they represented an inevitable social evolution, and the elites were merely helping to bring about a social transformation that, historically, was *bound* to happen. Similarly, the software devices are said to be an expression of technological forces; so they represent an inevitable evolution of software principles, and the elites are merely helping to bring about a technological transformation that was *bound* to happen. Like the political elites, the software elites see themselves as an enlightened vanguard: they alone can understand this evolution, and it is their duty to make us conform to it.

In reality, the mechanistic software theories are pseudoscientific, as are the social theories behind totalitarianism. The software elites, therefore, are the same kind of charlatans as the totalitarian elites. Like them, the software elites managed to assume control of society by entrancing us with their utopian visions – visions that are in fact absurd, and are accepted only because of our mechanistic tradition. Moreover, by accepting their deterministic theories we are *all* participating in the creation of a totalitarian society, just as the millions of people living under Nazism and Communism were by accepting the deterministic *social* theories promoted by *their* elites.

Totalitarian Democracy

The Totalitarian Elites

Our modernity and our mechanistic culture have engendered a new and dangerous phenomenon: educational and business institutions that have more power than our *political* institutions. This phenomenon is dangerous because, while engaged in teaching, or research, or marketing, these institutions are promoting totalitarianism. Thus, although politically our society is still democratic, in effect it is becoming totalitarian.[1]

[1] The expression "totalitarian democracy" (the present section's title) is explained in the next subsection, "Talmon's Model of Totalitarianism."

The totalitarianism promoted through education and through business is due to the mechanistic ideology. Mechanism has been so successful in the exact sciences, and in fields like engineering, that we are ready to accept it in any endeavour. As a result, these institutions have attained, undeservedly, an elitist position. For, we judge their importance, not by the *validity* of their ideas, but by the formality and precision with which they pursue these ideas; in other words, by the mechanistic nature, rather than the usefulness, of their activities.

Universities and corporations like the mechanistic ideology, therefore, because it affords them a privileged position in society regardless of whether their activities are useful or not. Our problems are becoming more and more complex – that is, less and less mechanistic; and yet, these institutions restrict themselves to mechanistic concepts. The reputation of mechanism was established long ago, when our problems were simpler, and mechanistic concepts were generally useful. Today, the mechanistic failures exceed the successes, but we continue to trust the mechanists just as we did in the past.

The reason we continue to embrace mechanism despite its failures is the propaganda conducted by universities and corporations. Since a widespread acceptance of mechanism is the only way to maintain their privileged position, these institutions are aggressively promoting it, while discrediting all other forms of thought. Thus, whereas in the past they were *practising* mechanism, today they are *enforcing* it. Clearly, the propaganda is necessary because mechanism is becoming less and less useful. Were mechanism indeed as beneficial as it is said to be, the mechanists would gain our respect simply by providing answers to our problems. As we saw in the previous chapters, however, mechanistic methods and theories are, in most fields, mere delusions.

❖

Mechanism becomes totalitarian when its status changes from *method* to *myth*. Once it becomes a system of belief, its principles are accepted unquestioningly, and those who were already practising it become revered elites. At that point, mechanism is officially taught and promoted, and every member of society is expected to adopt it.

Thus, if previously the harm was limited to the exploitation of society by the mechanists, now society itself is becoming mechanistic. If previously only the mechanists were wasting their time by pursuing useless activities, now everyone does it. In every field and occupation, people are increasingly judged, not by the value of their work, but by how faithfully they adhere to the mechanistic ideology. Through systematic indoctrination, every person is turned into a bureaucrat – a worker whose task is, simply, to obey the mechanistic principles.

The academic elites carry out the mechanistic indoctrination through the process known as education. The academics, we saw, restrict themselves to mechanistic ideas, because this permits them to stress the *methods*, rather than the *results*, of their research programs. Worthless activities can then be made to look important. And they extend this attitude to education: what they teach is only mechanistic concepts, only what can be described with neat theories and methods. Whether these theories and methods are effective or not is unimportant. Disciplines whose phenomena are almost entirely non-mechanistic – psychology, sociology, linguistics, economics, programming – are taught like the exact sciences. And the notions of expertise, talent, and professional responsibility in these fields are being degraded accordingly: what practitioners are expected to know is, not the utmost that human minds can attain, but only how to apply certain methods.

The business elites, for their part, carry out the mechanistic indoctrination through the process known as marketing. Like the universities, corporations restrict themselves to mechanistic ideas, because this permits them to engage in activities that are predictable and profitable. For example, they prefer products that can be made largely by machines, and workers whose skills are very low or very narrow, in order to control the process of production "scientifically." Most goods and services, therefore, are restricted to whatever can be done efficiently, through exact methods; so they reflect the limitations of business rather than our real needs. This is why we must be *persuaded* to depend on these goods and services. The persuasion – what we call advertising – consists mainly in exploiting human weaknesses through deceptive messages. The deception is required, obviously, in order to make the limited options possible through mechanistic concepts appear more important than they actually are. The ultimate goal of this indoctrination, then, is to force us all to replace our genuine values with artificial, mechanistic ones.

So mechanism can be just as utopian, and just as totalitarian, as a *political* ideology. As in the case of a political system, if accepted as unquestionable truth – as myth – the entire society will modify itself to fit its dogmas. In fact, it is even easier to be mesmerized by mechanistic ideas than it is by political ones, because we are less likely to suspect their advocates of dishonesty. Mechanism, after all, is not promoted by political parties, but by such respected institutions as universities and corporations. And its dogmas are justified by invoking such notions as science, efficiency, and progress.

A society where various elites are free to promote their ideologies is intrinsically totalitarian, if these ideologies seek to control the values held by large numbers of people. Whether an ideology concerns politics, or religion, or business, or technology, the result is always a spreading bureaucracy: more and more people cease to live a normal life, and follow instead the ideology's precepts. But totalitarian ideologies are pseudoscientific, so their utopian promises cannot be fulfilled. The harm they cause, therefore, extends beyond the inevitable disillusionment. For, if too many people accept the promises and cease performing useful activities, the entire society may collapse. Thus, totalitarianism destroys individual lives by tempting people to pursue worthless ideas, and destroys societies by corrupting their human resources.

By describing the promotion of an ideology with terms like "education" and "marketing," instead of "indoctrination" and "propaganda," we can continue to call our system democratic even while making it increasingly totalitarian. Again, it is not so much the *promotion* of ideologies that is harmful, as the fact that they are pseudoscientific, and hence bound to fail. Were these ideologies useful, they could be promoted through logic – as are, for instance, our truly scientific theories – and there would be no need for lies and delusions. (We studied in previous chapters the methods employed by charlatans to defend useless concepts, theories, and products; and we saw that most promotions today rely, in fact, on these methods. So the conclusion must be that most concepts, theories, and products promoted today are not as useful as they are said to be.)

A society that tolerates the advancement of *any* ideas through deception is, in effect, out of control. For, as these ideas are worthless, such a society is actually expending its energies on projects that constitute its own destruction.

❖

The reason we fail to notice the totalitarian nature of our present-day culture is that our brand of totalitarianism is being enforced, not by one, but by many different elites. Thus, when we are indoctrinated by the academic elite to embrace mechanism, or by the fashion elite to wear certain clothes, or by the tobacco elite to smoke cigarettes, or by the soft-drink elite to consume beverages, or by the fitness elite to manipulate certain contraptions, or by the automotive elite to prefer certain types of vehicles, or by the entertainment elite to enjoy certain types of TV shows, or by the cosmetics elite to use various concoctions, or by the financial elite to trust certain investments, or by the technology elite to depend on devices, we may be dealing with different organizations, but their ideologies share a common goal: to control our minds. They prevent us from developing useful knowledge, and encourage us to accept instead senseless ideas.

So, just like a *political* totalitarian elite, our elites have the right to shape the lives of millions of people. Also like a political elite, they do this by distorting knowledge, in order to make reality fit their ideology. But if we call this process informing, or educating, or marketing, or public relations, we can delude ourselves that our culture is different from a *politically* totalitarian one. In reality, totalitarianism based on a non-political ideology is just as harmful.

In addition to their diversity, it is the *weakness* of these elites that prevents us from noticing their totalitarian attitude. In the past, none of these organizations had the power to impose its particular ideology on more than a fraction of the population. And as a result, no one person was influenced by more than a few of them at a given time. Thus, although each elite believes that the whole world ought to accept its ideas, and is attempting to turn us all into the kind of bureaucrats needed to implement these ideas, none has had, so far, the power to carry out its plan. What we have had so far, therefore, is only mild, fragmentary totalitarianism.

Software, however, has changed this. Through software, it has finally become possible for a non-political elite, upholding a non-political ideology, to dominate the entire society. It is the *nature* of software that makes this possible. We mistakenly perceive software as a new kind of product, and hence the software elite as a risk no greater than the traditional elites. In reality, software is a new phenomenon, a new way for human beings to represent the world and to communicate with it. And consequently, the totalitarianism of a software ideology is far more virulent than the traditional ones. If software is like language (as we saw in chapter 4), an elite that controls software controls in effect the means to represent the world; so it controls the way we acquire knowledge, communicate, and conduct our affairs. Ultimately, the elite can control the way we think. For, by distorting our knowledge of software, it can distort our knowledge of everything else. Thus, while the traditional elites could affect only separate aspects of our life, a software elite can affect our entire existence. Through software, therefore, an elite can achieve complete totalitarianism.

As we saw earlier, what is needed to implement totalitarianism is a myth, an elite, and an expanding bureaucracy (see pp. 30–31). With this principle, we can explain why the other elites failed to achieve complete totalitarianism: because, through the traditional mechanistic myth, their bureaucracies could not expand beyond a certain point. And we can also explain why the software elite *can* achieve complete totalitarianism: because, through the mechanistic *software* myth, a bureaucracy can expand easily; it can expand, in principle, to include every member of society.

If the software variant of totalitarianism is as harmful as the political one, we must treat it the same way: we must be as worried about the theories promoted by the software elite, or the spread of a software bureaucracy, or the degradation of minds caused by software devices, as we would if these phenomena were part of a *political* movement. In previous chapters we studied the pseudoscientific nature of the mechanistic software theories, and the dishonesty of their promoters. Thus, if software mechanism is worthless, as we saw, we already have good reasons for fighting this ideology and the charlatanism it engenders. But full-fledged software totalitarianism will cause much greater harm. The incompetence and corruption found today in the world of programming is but a small problem compared with the widespread destruction of knowledge that we will suffer when our immersion in software mechanism is complete.

To combat software totalitarianism we must first understand it. However, this being a new phenomenon, we only have the studies of *political* totalitarianism as guide. Still, if the main characteristics of totalitarianism are the same in both variants, we should be able to identify in the political studies various aspects that parallel the trends we are witnessing in our software-related affairs. And this, in turn, will help us to appreciate the threat posed by the mechanistic software ideology; that is, to appreciate why the argument against software mechanism is more than just an argument against mistaken programming concepts.

What most political studies do is merely analyze various totalitarian societies and ideologies. To understand those aspects shared by the political and the software variants, however, what we need is an analysis of the fundamental, philosophical aspects of totalitarianism. This is why I have selected two particular studies. One is Talmon's model: a society that is founded on democratic values but is pursuing, in fact, a totalitarian dream – a dream stemming from mechanistic beliefs. The other is Orwell's model: a society where the elite is reducing language to its mechanistic aspects in order to degrade people's minds. We already know that our culture is mechanistic. Thus, since this culture permits us to pursue a totalitarian software ideology while living in a democracy, and since the role of software in society is similar to that of language, these two studies are especially relevant.

Talmon's Model of Totalitarianism

1

In his classic work, *The Origins of Totalitarian Democracy*, J. L. Talmon argues that modern totalitarianism derives from *democratic* ideas; specifically, from the ideas prevailing in the eighteenth century, and which led eventually to the French Revolution.[1] The French thinkers of that period were seeking an answer

[1] J. L. Talmon, *The Origins of Totalitarian Democracy* (New York: Praeger, 1960). Talmon continued this study in two later books, which discuss the evolution of the totalitarian ideology in the nineteenth and twentieth centuries.

to the age-old problem of human freedom: what kind of society can guarantee liberty, equality, and happiness for all its citizens? They were convinced that the oppression and misery pervading the world were due simply to ignorance, to man's failure to understand his own nature. If people agreed to explore this issue with an open mind, rather than allow one authority or another to influence them, they would be able to establish a perfect social system. That system would be democratic, and would constitute the best society that human beings can create.

While all thinkers shared this ideal, they differed in the method they recommended for achieving it. There were two schools: The first – which Talmon calls the *liberal* type of democracy – held that, left alone, people will one day discover, simply through trial and error, what is the best way to run a society. The second – which Talmon calls the *totalitarian* type of democracy – held that ordinary people are irrational and undisciplined, so they will never accomplish this on their own. The only practical solution is to endow a wise elite (the leaders, the rulers, the state) with absolute power, and allow it to control all human affairs. With this power, the elite could follow an exact plan, based on objective theories, and force every citizen to conform to it. The guarantee for a perfect society, it was believed, lies in this combination of scientific principles and complete conformism. The elite, of course, would never abuse its power. Since by definition a wise elite identifies itself with the people, it would only use its power benevolently.

The first opportunity a society had to put these ideas to the test was during the French Revolution. And, surprisingly perhaps, it was the *totalitarian* alternative that was chosen. Thus, because the revolution's leaders preferred a democracy based on scientific principles to one based on spontaneous decisions, the movement that started with the promise of universal liberty ended in a violent, totalitarian system. The advocates of scientific social planning learned nothing from this failure, however, and continued to promote their type of democracy ever since. In the twentieth century, the best-known implementation of totalitarian democracy was Communism, while Nazism was an extreme, especially brutal manifestation of the same idea. In the end, liberal democracy – which is what we understand today as "democracy" – proved to be a better alternative, and was adopted by one society after another. The struggle between the totalitarian and the liberal types of democracy, says Talmon, has shaped the history of civilization since the eighteenth century.

❖

The purpose of this discussion is to show that today's academic and business elites hold ideologies that are very similar to the political ideology of *totalitarian* democracy. And this is no accident: like the traditional elites, our elites claim that their ideologies are scientific (because based on mechanistic principles) and democratic (because beneficial to the majority of people); at the same time, they ask us to renounce all individual freedom (because only through conformism can a scientific and democratic ideology succeed). Stemming as they do from the same mechanistic delusions, these ideologies suffer from the same self-contradictions. Thus, if we want to understand today's totalitarian tendencies, particularly in our software-related affairs, a brief analysis of the original totalitarian theories will be helpful.

The eighteenth-century thinkers believed in the existence of an ideal, natural order. Impressed by the scientific advances of those days, they assumed that similar advances were imminent in social and political matters. Now, the discoveries in physics, chemistry, and astronomy had revealed a simple and logical pattern in the laws of nature. Thus, those thinkers concluded, a similar pattern must exist in the natural *social* laws. Human beings and human societies evolved as part of nature, and, given the beauty and logic of the natural laws already discovered, it is inconceivable that the natural *social* laws would prescribe oppression or unhappiness. The current societies suffer from these evils, therefore, only because we have not yet discovered the natural social laws. Once we discover them, our social relations will be as logical and successful as are our exact sciences.

Those thinkers also believed that all human beings are basically alike. The great differences we note in personality, intelligence, or wealth are due to accident; or they are artificially fostered by certain institutions, which have vested interest in maintaining these differences. Nature could not possibly have intended that such great disparity emerge between creatures which are practically identical when born.

In conclusion, since there undoubtedly exist some logical, natural laws for running a society, and since all human beings are naturally alike, the only explanation for the current misery is that our societies do not reflect the natural order. Specifically, we permit variation and inequality to arise among individual citizens. Hence, once we correct this mistake – once we reduce human existence to those aspects common to all people – the only social principles required will be those based on natural laws. By definition, the resulting system will constitute a perfect society: "If there is such a being as Man in himself, and if we all, when we throw off our accidental characteristics, partake of the same substance, then a universal system of morality, based on the fewest and simplest principles, becomes not only a distinct possibility, but a certainty. Such a system would be comparable in its precision to geometry."[2]

[2] Ibid., pp. 29–30.

These ideas were expressed most forcefully by Jean-Jacques Rousseau, who also explained how to implement them: by divesting people of their personal qualities and endowing them instead with the common, natural ones – what he called the general will. Achieving this transformation is the task of a sovereign (in practice, the ruling elite). Although immanent in each individual, the natural qualities are masked by his current, selfish character, and must be brought out through special education. The individual, in fact, may be so corrupt that he would not appreciate the importance of the transformation, in which case the sovereign must *enforce* it. The revolution will succeed only when all citizens adopt without reservation the general will.

Thus, while the other thinkers saw the idea of a natural social order as little more than a theory, Rousseau presented it as a plan for immediate action. Through the absurd notion of a general will – said to be natural to people but at the same time requiring a powerful elite to enforce it – Rousseau gave rise to modern totalitarianism: a political system that is totalitarian even though grounded on democratic principles.[3]

So it is the idea of natural, innate qualities that is used to justify totalitarianism. The elite claims that, since its ideology reflects some natural, and hence superior, human qualities, forcing an individual to conform to it is not an act of coercion but a sort of teaching. In an ideal world, that individual would display those qualities on his own. It is the fact that he lives in a corrupt society that distorts his character and prevents him from attaining his higher, natural self. In effect, society has denatured him. All that the elite does, then, is restore him to what nature had intended him to be. Thus, we do not object when a teacher forces his pupils to learn rules of grammar or arithmetic, and punishes them if they forget those rules. We do not object because the rules reflect valid, natural laws. Similarly, forcing people to conform to a natural ideology is a form of education: instead of grammar or arithmetic, what we must assimilate now is some *social* rules – rules which reflect the natural human condition, and which will therefore help us to create a perfect society.

❖

We can recognize in these ideas the circularity characteristic of mechanistic delusions. Totalitarian elites see themselves as social engineers, as experts who know how to design societies. Their ideas are a breakthrough in political thought, they say, and what they need now in order to create a perfect society is the authority to implement these ideas. What they need, in other words, is the power to control the lives of millions of people. And they invoke the

[3] Ibid., pp. 40–43.

mechanistic philosophy as justification: first they invent some mechanistic theories that match their ideas, then they use these unproven theories in support of each other. Since mechanism is universally equated with science, few notice the circularity of this line of logic.

The first theory the elites invent to defend their ideas is that societies are deterministic systems. So, they say, we should be able to design a society by following rigorous methods, just as we do in engineering projects. The elites need such a theory because they intend to replace the current social order with a new one; and only if societies can indeed be created from plans, like buildings, can such a project succeed. In reality, there is no evidence that societies can be designed as we design buildings. The millions of individuals who make up a society are sufficiently different from one another to cause complex, unpredictable social phenomena. It is precisely this diversity that makes the notion of a perfect society a fantasy.

Since it is the differences between individuals that would prevent the elites from implementing their ideas, they are compelled to invent a second theory. They say that human beings are naturally identical and virtuous, and the differences between them are simply deviations from this ideal, due to the corrupt society they live in. And this theory too is mechanistic: it claims, in effect, that human beings are born as a sort of automatons, all driven by the same program. Also like the first theory, there is no evidence that this is true. The elites *wish* this to be true; for, only if human beings are indeed automatons can the plan for erasing the differences between them succeed. The plan calls, in effect, for deleting the diverse, wrong programs running now in millions of individual minds, and installing in all of them an identical, correct program.

So the totalitarian philosophy is not the serious political thesis it is claimed to be, but a mechanistic delusion. What the elites really want is the power to control society; and they rationalize this megalomania by making their arguments look like scientific theories. They invent a theory about individuals in order to support a theory about societies. Both are fantasies, but together they seem to express self-evident truths. What the elites do, in reality, is invoke one mechanistic hypothesis as support for another. Ultimately, they use the mechanistic philosophy to defend the mechanistic philosophy. Being circular, their arguments are fallacious; they do *not* prove that totalitarianism can help us to create a perfect society. The circularity can be detected, in fact, in the very definition of the general will: "There is such a thing as an objective general will, whether willed or not willed by anybody. To become a reality it must be willed by the people. If the people does not will it, it must be made to will it, for the general will is latent in the people's will."[4]

[4] Ibid., p. 43.

Talmon calls this delusion the paradox of freedom: the elites promise us freedom, but at the same time they tell us that the only way to have freedom is by giving up individuality, and by conforming to the great whole that is society. Conformism, though, is the opposite of freedom. So, to resolve the contradiction, the elites *redefine* the notion of freedom as *conforming to an ideal*: "On the one hand, the individual is said to obey nothing but his own will; on the other, he is urged to conform to some objective criterion. The contradiction is resolved by the claim that this external criterion is his better, higher, or real self.... Hence, even if constrained to obey the external standard, man cannot complain of being coerced, for in fact he is merely being made to obey his own true self. He is thus still free; indeed freer than before."[5]

The most striking feature of totalitarianism, then, is this insistence on shaping the character of millions of individuals to fit a common mould, while claiming that they continue to be free: "From the difficulty of reconciling freedom with the idea of an absolute purpose spring all the particular problems and antinomies of totalitarian democracy. This difficulty could only be resolved by thinking not in terms of men as they are, but as they were meant to be, and would be, given the proper conditions. In so far as they are at variance with the absolute ideal they can be ignored, coerced or intimidated into conforming, without any real violation of the democratic principle being involved."[6]

❖

Absurd as they are, these question-begging arguments have been adduced to justify totalitarianism for more than two hundred years. From science-fiction authors to progressive sociologists, from paranoid dictators to learned philosophers, every apologist has defended his particular brand of totalitarianism through the same mechanistic delusions. Thus, since mechanism dominates our present-day culture no less than it did previous ones, we shouldn't be surprised that our own elites, in universities and in business, invoke it to justify *today's* brands of totalitarianism. In the end, all elites say the same thing: the only way to improve matters is through complete conformism; specifically, by implementing a mechanistic ideology and forcing everyone to adhere to it.

What the elites want, in reality, is power – the power that comes from controlling knowledge and minds. And they attain this power by promising to solve our non-mechanistic problems with simple, mechanistic methods. All *we* have to do, they tell us, is obey the ideology; that is, restrict ourselves to

[5] Ibid., p. 40.

[6] Ibid., pp. 2–3.

mechanistic concepts. Non-mechanistic problems, however, cannot have mechanistic solutions, so the promise is a fraud. But because it is so appealing, we believe it. And this is how, at any given time, one or more elites are exploiting us.

Thanks to their similarity, then, totalitarian ideologies are easy to recognize: An elite promotes certain ideas about people and societies – ideas that are precise and attractive, but very different from the way people normally live. For these ideas to succeed, therefore, everyone must change so as to conform to them. And if some of us resist the change, this can only mean that we are too ignorant to appreciate the promised benefits. After all, being mechanistic, the ideas themselves cannot possibly be wrong. So we must be *forced* to change. This is not coercion, though, but education: we are forced, in effect, to think and live correctly. Whether the ideas concern politics, or work, or personal life, with proper teaching anyone can learn to appreciate them. (See also the related discussion in the introductory chapter, pp. 17–18.)

2

Let us see now how Talmon's totalitarian model is reflected in today's ideologies. Starting with the academic elites, the idea promoted is that phenomena involving minds and societies can be represented mechanistically, just like physical phenomena. In other words, we should be able to explain all human phenomena from a knowledge of the basic human propensities, just as we explain the operation of a machine from a knowledge of its basic components. It *is* possible to discover exact theories of mind and society. One day, we will be as successful in fields like psychology, sociology, and linguistics as we are in physics and astronomy.

These theories, however, do not work. And they do not work because human beings and human societies are *not* the deterministic systems the mechanists assume them to be. In chapter 3 we saw that these theories are, in fact, pseudoscientific: when falsified by evidence, the mechanists resort to various stratagems in order to cover up the falsifications. So, if three hundred years of mechanistic philosophy have failed to produce a single working theory in the human sciences, and if it is so easy to show that the promoters of these theories are not scientists but charlatans, common sense alone ought to prompt us to question the academics' elitist position. The fact that we do not question it demonstrates how successful is the mechanistic propaganda conducted by the universities.

Academic mechanism, thus, is a totalitarian ideology – because it asks us to change so as to conform to its tenets. It does not earn its status through real achievements, but through coercion: we are *forced* to accept it, regardless of whether the theories work or not. We are intimidated by its successes in the exact sciences, and we allow charlatans to fool us into accepting it in every other field. Both education and research are now little more than mechanistic indoctrination: every aspect of reality is described in mechanistic terms, and we must restrict ourselves to mechanistic practices. As a result of this indoctrination, we treat mechanism as unquestionable truth, as the only valid form of thought. And we respect anyone who upholds a mechanistic idea, even if the idea is worthless.

So the change demanded by the mechanistic ideology consists in replacing our traditional perception of knowledge, science, and research with a degraded one: the pursuit of mechanistic ideas. Instead of admiring accomplishments, we admire conformism. What we expect to see in academic work is not expertise and originality, not the utmost that human beings can attain, but merely the faithful application of mechanistic methods. Thus, since anyone with a bureaucratic mind can follow methods, individuals incapable of doing anything useful are perceived as scientists.

And this is not all. The theories promoted by the mechanists are about human beings – that is, about *us*. So, when we accept them, we do more than just agree to treat the academic charlatans as elites. What we really do is accept their claim that we are deterministic systems. By respecting the mechanists and their work, we are saying in effect that we think their project is important, and likely to succeed. But this project is an attempt to prove that human beings are in reality automatons. So our acceptance means that, like the academics themselves, we believe this *is* what we are. Our acceptance shows, therefore, how advanced is our mechanistic indoctrination – our dehumanization. For, we would not respect researchers who try to prove that we are automatons unless we already thought and acted, to some degree, like automatons. The reason we accept their theories, then, is that we no longer see ourselves as free and responsible agents.

In the end, because we trust the mechanists and increasingly restrict ourselves to mechanistic performance, these theories are becoming more and more plausible: they describe human beings and human societies more and more accurately. This is not because the mechanists are right, though, but because *we* are becoming, little by little, the deterministic systems they say we are. Thus, while failing in human affairs as scientific concept, mechanism is successful as totalitarian ideology: we are indeed changing to conform to its tenets.

Let us turn next to the corporate elites. The idea promoted now is that every problem can be solved by purchasing something. In personal or professional pursuits, in our kitchens or in our offices, in matters of health or intellect or finance, the solution to a problem can always be found in a product sold by a company. While the traditional view is that we must study if we want to gain knowledge, practise if we want to develop expertise, change our lifestyle if we want to be fit, do something useful if we want to get rich, and alter our world view if we want to be happy, modern companies can help us avoid these challenges: we can achieve the same results, immediately and effortlessly, simply by purchasing their latest products.[7]

These products, however, do not work – at least, not in the way we are promised. And they do not work because difficult challenges cannot be met simply by purchasing something. Ready-made products are limited, by their very nature, to mechanistic concepts: they embody specific combinations of features and capabilities, on the assumption that every problem can be reduced to such combinations. Our most important problems, though, are non-mechanistic, because they reflect the complex phenomena that make up our existence. They can only be solved, therefore, through our own knowledge, experience, and effort. Products alone cannot help us, because no set of products can embody enough combinations of details to satisfy our combinations of needs. We are impressed by their ability to solve isolated, mechanistic problems, and we are fooled by the claim that they can also solve the important, complex ones. So, although products usually *function* as promised, this doesn't mean that they can also improve our life as promised.

If a product is actually not as useful as we think it is, the only way for its maker to make us buy it is by deceiving us. The process whereby a useless thing is made to appear useful is known as advertising. And, since more and more products need to be sold in this manner, advertising has become the most important part of trade. To put it differently, if advertising were restricted to factual information about a product's features (similarly, for instance, to the arguments accepted in a court of law), perhaps only 10 percent of what is being bought today would continue to be bought: those products that are indeed as useful as we think they are. It is not too much to say, then, that our economy is almost entirely dependent on the permission that companies have to tell lies and to exploit people's ignorance.

This contrasts sharply with the situation, say, one hundred years ago, when most products were useful and very few had to be sold through deception. In the past, the promotion of a product needed only plain statements, and perhaps

[7] To extend the range of this ideology, many services (bank accounts, insurance plans, investment schemes) are now called products.

some flourishes and exaggerations. Today, on the other hand, promotion means a systematic generation of delusions. Thus, advertising techniques that are now universal were employed in the past only by charlatans. Some examples: presenting particular instances (testimonials, success stories, case studies) as evidence of the product's usefulness, which is logically equivalent to lying (see p. 218); describing the product with deliberately misleading sentences – sentences that appear to state important facts while saying in reality nothing meaningful or accountable (see "The Practice of Deceit" in chapter 5); arbitrarily displaying attractive, smiling faces, which compels us to associate the product with beauty, youth, health, and happiness; deceptive prices, like $19.99; adding background music and special effects on radio and television – in order to distract and confuse us, and to induce a favourable mood.

The reason for the incessant lies is the declining usefulness of mechanism. In the past, when our problems were simpler, ready-made products were quite effective, so there was no need for deception. But our world is becoming more and more complex, and complex problems cannot be solved mechanistically – that is, by separating them into simpler ones. Advertising, thus, serves as mechanistic indoctrination: the corporate elites must persuade us that their products, which are based on mechanistic concepts, can solve our complex problems.

Like academic mechanism, then, business mechanism is a totalitarian ideology – because it asks us to change so as to conform to its tenets. When we succumb to advertising, we do more than just agree to be exploited by charlatans: we agree to forgo our non-mechanistic capabilities, and to restrict ourselves to mechanistic performance. While the world consists of complex phenomena, we see only its mechanistic aspects. Ultimately, the change demanded of us is to simplify our lives to the point where all our needs can be satisfied by purchasing ready-made products, and to limit our knowledge so as to remain dependent, in everything we do, on these products.

If we resist the change, the elites tell us that the mechanistic concepts only *appear* to be restrictive: we fail to appreciate their value because of our current, inefficient habits. Mechanism means science, we are told, so it is silly to think that our minds can be better than products based on scientific concepts. What we interpret as creativity and originality – what these products are eliminating – is in reality an old-fashioned, undisciplined way of doing things. Thus, just as education often forces children to accept notions they don't understand, for their own good, *we* must be forced to depend on ready-made products, for our own good. In the end, the restriction to mechanistic concepts is no more coercive than any type of education. What we are taught now is how to live efficiently; in particular, how to replace the dependence on personal knowledge and skills with a dependence on modern products.

Also like academic mechanism, business mechanism is successful as totalitarian ideology; that is, we *are* becoming the automatons the elites say we are. For, if we are forced to spend more and more time with useless mechanistic solutions, we are bound to spend less and less time developing our non-mechanistic capabilities. As we get to depend on ready-made products in every activity, the only knowledge we acquire is the trivial, mechanistic type needed to use these products. So we are being reduced, little by little, to the level of machines. But the result of this transformation is that the claims made for ready-made products are becoming increasingly accurate: since we no longer care about complex phenomena, it no longer matters that our complex problems remain unsolved; since we are dealing only with the mechanistic problems, the products increasingly appear to be as useful as their promoters say they are.

Orwell's Model of Totalitarianism

1

George Orwell's conception of totalitarianism is best known from his last work, *Nineteen Eighty-Four*, which was published in 1949. But to appreciate his remarkable insight into the nature of totalitarianism, and his ongoing preoccupation with it, we must study his writings over the preceding ten years. Although in *Nineteen Eighty-Four* he depicts an established totalitarian state, his aim was not to expose the evils of Nazism and Communism (the totalitarian ideologies of the 1940s), but to draw attention to the totalitarian tendencies of the *democratic* cultures.

Because he died shortly after the book's publication, Orwell did not have the opportunity to clarify its links to his actual views and concerns. A letter he wrote at the time, and which was widely published, is probably the only record of these links: "I do not believe that the kind of society I describe necessarily *will* arrive, but I believe (allowing of course for the fact that the book is a satire) that something resembling it *could* arrive.... The scene of the book is laid in Britain in order to emphasize that the English-speaking races are not innately better than anyone else and that totalitarianism, *if not fought against*, could triumph anywhere."[1]

Orwell's model, then, involves not just a certain type of totalitarianism, but

[1] George Orwell, "Letter to Francis A. Henson," in *The Collected Essays, Journalism and Letters of George Orwell*, vol. 4, eds. Sonia Orwell and Ian Angus (London: Penguin Books, 1970), p. 564.

also the *progression* of a society toward totalitarianism. He noticed that many aspects of the degradation reached in the totalitarian countries could also be found, to some degree, in the democratic ones. And this degradation was growing and spreading. In *Nineteen Eighty-Four*, Orwell drew an exaggerated, unrealistic picture of totalitarianism, in order to stress its dehumanizing effects. But, he warns us, while *that* totalitarianism is indeed a fantasy, its milder counterpart in our own society is real. It would be instructive, therefore, to review some of the totalitarian aspects of our culture, and to see how they have evolved since Orwell's time. This will help us to recognize the totalitarian aspects of our software practices, which, of course, he could not have anticipated.

❖

One thing Orwell noticed was the ease with which people could be persuaded to accept totalitarian ideas. Totalitarianism, we saw, is presented as a *scientific* doctrine, because it is derived from mechanism. Most people fail to recognize its fallacies, and succumb to its utopian promises. Thus, like all pseudosciences, totalitarianism is appealing because it seems to offer easy solutions to complex problems: people accept it for the same reason they accept astrology, superstitions, and magic systems. But Orwell was especially annoyed to see that the most ardent supporters of totalitarianism are found among *educated* people: "I believe … that totalitarian ideas have taken root in the minds of intellectuals everywhere."[2] "What is sinister … is that the conscious enemies of liberty are those to whom liberty ought to mean most.… The direct, conscious attack on intellectual decency comes from the intellectuals themselves."[3]

Recall the mechanistic pseudosciences we examined in chapter 3. Instead of trying to understand the true nature of minds and societies, the academics *assume* they are mechanistic phenomena. Theories based on this assumption never work, but the academics refuse to admit that they are wrong, that human phenomena are in fact non-mechanistic. Thus, the academics are not serious scientists. They have redefined their responsibility, from the difficult challenge of discovering useful theories, to the easier challenge of practising mechanism. And, since mechanistic ideas in human affairs are intrinsically totalitarian, the tendency among intellectuals to accept totalitarian ideas implicitly – what Orwell condemned – is a consequence of their tendency to accept *mechanistic* ideas implicitly. Orwell noticed this corruption even in the 1940s, and was right to warn us about its growth. For, this is indeed what has happened: while

[2] Ibid.

[3] George Orwell, "The Prevention of Literature," in *Collected Essays*, vol. 4, p. 93.

mechanistic theories were already a temptation in the human sciences, they have become, since then, the only type of theories officially accepted.

In the end, "a society becomes totalitarian when its structure becomes flagrantly artificial: that is, when its ruling class has lost its function but succeeds in clinging to power by force or fraud."[4] If we take the academic elite to be one of our ruling classes, this observation describes perfectly its degradation since Orwell's time. The mechanists have turned disciplines like linguistics, economics, and programming into pseudosciences. What they perceive as research is in reality a never-ending series of attempts to cover up the failure of mechanistic theories. Thus, they are deceiving society in order to maintain their elitist position. As Orwell said, they are clinging to power through fraud, and in so doing they are fostering totalitarianism.

❖

Another thing Orwell noticed and warned about was the trend toward a centralized economy, or collectivism. While enthusiastically advocated by experts as a progressive and effective system, a state-directed economy is, in reality, the exact opposite: it corrupts both the economy and politics, and undermines liberal values by promoting conformism. Thus, Orwell was one of the few to recognize the link between a government-controlled economy and totalitarianism. Writing in 1941, he makes this observation: "When one mentions totalitarianism one thinks immediately of Germany, Russia, Italy, but I think one must face the risk that this phenomenon is going to be world-wide. It is obvious that the period of free capitalism is coming to an end and that one country after another is adopting a centralized economy that one can call Socialism or state capitalism according as one prefers."[5] A socialist himself, Orwell had by then realized that socialism is largely a theoretical concept, that in practice it leads to totalitarianism.

In practice, therefore, the *economic* philosophy of central planning cannot be distinguished from the *political* philosophy of totalitarianism. And it is hardly necessary to point out that the intervention of governments in their country's economy has been increasing steadily since Orwell's time, as he said it would. In the last twenty years, particularly, fantastic monetary and fiscal policies – politically motivated – have given rise to the kind of central control that feeds on itself. We have reached the point where many countries can no longer function as liberal, free economies, and depend for survival on a perpetual increase in central control and a continuation of the same

[4] Ibid., p. 89.

[5] George Orwell, "Literature and Totalitarianism," in *Collected Essays*, vol. 2, p. 162.

fantastic policies. Under these conditions, the drift toward totalitarianism is not surprising.

With our model of simple and complex structures it is not difficult to understand the delusions of central economic planning. A country's economy is a complex phenomenon. It is the result of an infinity of interactions between millions of individuals, who act in various capacities: consumers, producers, workers, managers, inventors, entrepreneurs, financiers, and so forth. Thus, by encouraging uninhibited interactions, a free economy is the most likely to reflect, in the long run, the true needs and capabilities of the people. Governments like the idea of central planning because they believe it to be an improvement over a disorganized, free economy: why wait for the results of some random interactions, when we have experts who can control this phenomenon scientifically, and thereby guarantee a stable, ideal economy?

To control the economy, though, the experts must understand it. And, as we know, a complex phenomenon cannot be understood as we understand the working of a machine; that is, precisely enough to predict all its manifestations. The experts, therefore, are compelled to invent theories based on a simplified, mechanistic version of the economy. They ignore the infinity of low-level interactions that make it up, and study separately its high-level aspects: inflation, unemployment, growth, government debt, stock market, gross domestic product, and so forth. In other words, they attempt to depict a complex structure as a combination of several simple ones. At this point, it seems logical to represent those separated aspects with exact values (averages, percentages, formulas, charts), and, moreover, to attempt to control the economy by manipulating these values. They forget that what they are studying is no longer the real economy, but a simpler, imaginary version. They may even manage to improve one aspect or another. But because they ignored the interactions between them, this is accomplished at the expense of other aspects, which deteriorate.

So the mechanistic economic theories are pseudoscientific. In the end, because they are concerned with minds and societies, they suffer from the same fallacies as the theories we examined in chapter 3. All these theories fail for the same reason: their assumption that human beings and human societies are deterministic systems.

Our model also explains why the idea of central economic planning is totalitarian. Its most appealing element is the promise of financial security for every citizen: the state will take care of our basic needs, leaving us free to pursue our careers and lifestyles. This promise, however, is an illusion. To implement a centrally-controlled economy, the state must assume that the needs of millions of individuals can be analyzed and controlled. It must assume, in other words, that human beings are a sort of automatons, driven by known programs. So, because it is based on invalid premises, because our needs are in reality complex and diverse, this economy is bound to fail. The only way to make it work is by *enforcing* it; namely, by asking us to replace our actual needs with the kind of needs that make central planning possible. Through education and through propaganda, we are told what knowledge is correct, what facts are important, what career is appropriate, what things must be purchased, what conduct is desirable, what to expect in the future, and so on. In the end, our needs will be simple, uniform, and predictable – the needs of automatons. To put this differently, since mechanistic economic theories do not reflect human nature, to make them work we must modify the people to match the theories: we must turn them into deterministic systems.

We can have government-controlled financial security, then, only if we agree to obey certain standards. In exchange for security, we replace individuality with conformism. Thus, there is only one step from accepting central economic planning to accepting totalitarianism. And, again, Orwell saw this trend clearly: "With [centralized economy] the economic liberty of the individual, and to a great extent his liberty to do what he likes, to choose his own work, to move to and fro across the surface of the earth, comes to an end. Now, till recently the implications of this were not foreseen. It was never fully realized that the disappearance of economic liberty would have any effect on intellectual liberty. Socialism was usually thought of as a sort of moralized liberalism. The state would take charge of your economic life and set you free from the fear of poverty, unemployment and so forth, but it would have no need to interfere with your private intellectual life.... Now, on the existing evidence, one must admit that these ideas have been falsified. Totalitarianism has abolished freedom of thought to an extent unheard of in any previous age."[6]

2

The best-known aspect of Orwell's totalitarian model is the use of language to control minds. (Orwell is generally recognized as the first thinker to study seriously this phenomenon.) It is from his discussion in *Nineteen Eighty-Four* that most people are familiar with Orwell's ideas (see "Orwell's Newspeak" in chapter 5). Just as it exaggerates the other aspects of totalitarianism, though,

[6] Ibid. Note how Orwell is referring to socialism and to totalitarianism interchangeably. The most outspoken critic of central economic planning was probably philosopher and economist F. A. Hayek. For fifty years, in numerous studies, Hayek exposed the fallacies and the totalitarian tendencies of this idea. His best-known book on this subject is *The Road to Serfdom* (Chicago: University of Chicago Press, 1994, 50th anniversary ed.). It is worth noting that Orwell actually read this book and praised it in a brief review.

that book exaggerates the language abuses, in order to demonstrate the *potential* of language manipulation. Orwell's intent was not so much to attack the totalitarian ideology itself, as to warn us that any society can become totalitarian. Thus, when we study his earlier writings, we realize that the hypothetical language abuses depicted in *Nineteen Eighty-Four* are a reflection of *real* abuses – those he noticed in the society of his time. So they are not a wild fantasy, but a logical extrapolation of *existing* conditions.

In our analysis we concluded that Orwell's chief contribution has been to make us aware of the link between language, mechanism, and totalitarianism (see pp. 403–405). The three are inseparable. Thus, in a totalitarian society people must act like automatons, and language is an important part of this transformation: by reducing language to its mechanistic aspects, the elite can restrict knowledge and thought to the level of machines. Conversely, a society where various elites are permitted to manipulate language in this fashion will be restricted to mechanistic values, and will become in the end totalitarian. How a society uses language, therefore, is a good indication of its progression toward totalitarianism: the greater the manipulation of language, the more totalitarian the society.

Orwell studied the language employed in speeches, pamphlets, articles, and debates, and saw that it was designed largely to deceive, rather than inform. He also noticed that the deception was achieved by restricting discourse to high levels of abstraction. Instead of simple and precise statements, the propagandists use euphemisms, vague terms, slogans, and standard phrases: "The whole tendency of modern prose is away from concreteness."[7] "As soon as certain topics are raised, the concrete melts into the abstract and no one seems able to think of turns of speech that are not hackneyed: prose consists less and less of *words* chosen for the sake of their meaning, and more of *phrases* tacked together like the sections of a prefabricated hen-house."[8]

We recognize this style as mechanistic language. Recall our discussion in chapter 5. Only by starting with *low-level* linguistic elements can a message convey information. When communicating through high-level elements – through prefabricated linguistic parts – the deceivers force us in effect to commit the two mechanistic fallacies, abstraction and reification: they restrict us to a fraction of the alternatives present in the new knowledge, and they prevent us from linking their message to our previous knowledge.

The aim of mechanistic language, then, is to control minds. To discover the meaning of a message, we must combine the meaning of its words and phrases with the knowledge structures already present in the mind. And when this

[7] George Orwell, "Politics and the English Language," in *Collected Essays*, vol. 4, p. 163.
[8] Ibid., p. 159.

process starts at high levels of abstraction, very few combinations are possible. Moreover, if those words and phrases are purposely selected so as to mislead us, we will create only *wrong* combinations, those that do *not* reflect reality.

Charlatans prefer high levels of abstraction, therefore, because of their usefulness as means of deception. An acronym, for example, is in effect a word that stands for a whole phrase – a phrase which in its turn stands for many combinations of facts. But by employing the acronym instead of the whole phrase, a charlatan can make us associate it with just *a few* combinations: those we already perceive as "good." So we end up interpreting the acronym itself, and everything involving it, as "good." Having lost the lower levels – the individual words, their meanings and associations – we can no longer judge how important or unimportant are the facts subsumed by the acronym. Thus, while the high level of abstraction of the acronym seems to function merely as abbreviation, its real purpose is to shape and restrict thought. (See the discussion in chapter 5, pp. 371–372, 393–394, 401–402.)

Like acronyms, any high-level linguistic form – standard phrases, slogans, and the rest – can be used to avoid details and to obscure facts. In the aforementioned essay, Orwell analyzes several instances of political writing, and notes that this style is widespread: "This mixture of vagueness and sheer incompetence is the most marked characteristic of modern English prose, and especially of any kind of political writing."[9] But, while found earlier mainly in political writing, this style is employed today in nearly every field. In business computing, for instance, an article may be nothing more than some bombastic sentences praising the latest fads, reinforced with fashionable acronyms, and interspersed with slogans like "IT strategic planning," "empowering the enterprise," "competitive advantage," "mission-critical applications," and "business agility."

Improper use of high levels of abstraction is a sign of bad English, of course. But those who employ this style do it deliberately. For, their intent is not to debate logically a particular issue, but on the contrary, to force their readers to accept a distorted view of that issue. So this kind of writing betrays not so much a linguistic deficiency as an effort to control minds, which is the essence of totalitarianism. Or, putting this in reverse, only writers with a totalitarian attitude need to employ such a style. Also, the style's prevalence – the fact that we accept it rather than condemn it – indicates that the entire society is becoming totalitarian. It is this link between language and totalitarianism that preoccupied Orwell: "There does seem to be a direct connexion between acceptance of totalitarian doctrines and the writing of bad English ...."[10]

[9] Ibid.

[10] George Orwell, "Editorial to *Polemic*," in *Collected Essays*, vol. 4, p. 190.

❖

"To be corrupted by totalitarianism one does not have to live in a totalitarian country."[11] What Orwell meant is that the totalitarian mentality – elitism, conformism, bureaucratization, mind control – is found everywhere, and can corrupt any society. He was describing mostly its effect on writers and commentators, but this mentality has been spreading, and it affects now every aspect of society.

In the end, non-political totalitarianism can be as harmful as the political kind. If every elite is permitted to promote its ideology, and to deceive and exploit society, their total effect can be significant even if the individual elites are not. This is true because all these ideologies are similar to the totalitarian one: they claim that ideas based on mechanistic principles can solve our complex, non-mechanistic problems. The elites must uphold such ideologies because they can *only* offer us mechanistic solutions. We alone, with our minds, can conceive the non-mechanistic ones; and for this we need no elites. To stay in power, therefore, the elites must incessantly persuade us that their mechanistic concepts are more important than our minds. And this is why, ultimately, all elites deceive and exploit society in the same way.

We can also understand now why all elites end up manipulating language. We use language to represent the world in our minds, and to communicate with it. This is possible because language permits us to create complex knowledge structures. Since the world consists of complex phenomena, we *must* develop complex structures if we want to mirror the world accurately in the mind. By restricting language to its mechanistic aspects, the elites hope to make us see only the *mechanistic* aspects of the world – only the simple, isolated phenomena. And this, in turn, would make us accept their mechanistic ideas.

Each elite misleads us in a few, specific situations; but if all of them do this, it means that we are being misled all the time. Each elite wants to control just one aspect of our life; but between them, they control our entire existence. While each elite is promoting a different idea, they all do it by restricting us to high levels of abstraction, so they all prevent us from developing complex knowledge structures. Their goal, again, is to make their mechanistic ideas appear more important than they actually are. But, even though individually the deceptive messages may be weak, their cumulative effect is pernicious. If we are restricted to mechanistic values in all our affairs – in personal and in professional matters, in education and in business – our non-mechanistic capabilities remain undeveloped. Ultimately, we will indeed see only the mechanistic aspects of the world, just as the elites intended. At that point, those

[11] Orwell, "Prevention of Literature," p. 90.

useless ideas will finally seem important to us, because we will only be able to judge them with limited, mechanistic knowledge.

Clearly, then, if we live in a society where various elites have the right to control our knowledge and our values, the fact that our *political* system is democratic is irrelevant. If these elites are shaping our minds so as to accept mechanistic ideas that serve their interests, and if between them they have more power than our political institutions, our system is in effect totalitarian. To appreciate this, imagine that we had, not many academic and business elites inducing us to accept mechanistic ideas, but only one, political elite doing it. We would then easily recognize the system as totalitarian. In practice, therefore, there is no real difference between the two alternatives.

Software Totalitarianism

1

Talmon's model, we saw, can explain why academic and business mechanism become totalitarian ideologies. As in the case of political ideologies, the elites ask us to change so as to conform to an exact theory. This combination of science and total conformism, they say, is what will bring about a perfect society.

Let us use Talmon's model to explain why *software* mechanism becomes totalitarian. Software mechanism is, ultimately, the marriage of academic mechanism and business mechanism: the mechanistic software theories are invented in universities, and the software companies invoke these theories to justify the idea of software *products*.

The software theories claim that software applications are nothing but modules within modules, so the most effective way to develop them is by emulating the process of manufacturing. Devices like cars and appliances are designed as hierarchical structures of smaller and smaller subassemblies, ending with parts that are simple enough to be made directly. With this method, the task of manufacturing is reduced to the easier task of *assembling*: no matter how complex the finished product, every stage in its manufacture is now as simple as combining a number of parts into a larger part. Similarly, if we design our software applications as hierarchical structures of modules, programming will be reduced to the easier task of assembling pieces of software: starting with some small parts, we will build larger and larger modules, until we reach the complete application. Working in this fashion, even the most complex applications can be developed with skills no greater than those required to combine pieces of software.

If software can be built as we build cars and appliances – if, in other words, software is merely a new kind of product – the conclusion is that what we need is not expert programmers but a software industry: companies that make software products just as manufacturing companies make the traditional products. By running, as it were, efficient software factories, these companies should be able to supply most applications that society needs. And to help us build on our own those applications that are too specialized to be made as mass-market products, the software companies can give us *development tools.* These sophisticated software devices simplify the development of applications by providing high-level starting elements; namely, relatively large software subassemblies, instead of the small parts used in traditional programming. With these devices, even the least experienced among us should be able to create unique, customized applications.

This mechanistic software dream, however, cannot be fulfilled. As we saw in previous chapters, the theories are wrong when assuming that software applications can be treated as simple hierarchical structures. The facts, processes, and events that make up our affairs give rise to complex phenomena, and hence interacting structures. So, to represent them accurately, our software applications too must consist of interacting structures. If we follow the theories and separate the software structures, our applications will not match reality; for, as simple structures, they cannot display all the alternatives displayed by the complex phenomena. And if, in addition, we start with high-level elements, there will be even fewer alternatives. When forcing us to separate structures and to start from higher levels, the software elites force us in effect to commit the two mechanistic fallacies, reification and abstraction. The reduction in alternatives, then, is not surprising.

Whether we buy ready-made applications or make our own with development tools, applications based on mechanistic concepts can represent only the simple, mechanistic aspects of our affairs. Thus, the claim that these expedients have replaced the need for traditional programming, and for programming expertise, is a fraud. Only by resorting to our non-mechanistic capabilities – that is, through personal skills and experience – can we create applications versatile enough to represent accurately our affairs.

If the ready-made applications and the development tools are not, in fact, as useful as we think they are, the only way to make us depend on them is through deception. And indeed, software products are advertised just like the traditional consumer products: through testimonials, success stories, misleading language, portrayal of happy faces, and so forth. Thus, while addressing mostly businesses, and while discussing such issues as productivity and efficiency, software advertising is merely exploiting human weaknesses and ignorance, just like traditional advertising.

The goal of traditional advertising, we saw, is to persuade us that products based on mechanistic concepts will also solve our *complex* problems. Similarly, the goal of software advertising is to persuade us that applications based on mechanistic software concepts can represent complex phenomena – our business, social, and personal affairs. Also like traditional advertising, if software advertising were restricted to verifiable claims, only a small fraction of the applications and tools being used today would continue to be used: those that are indeed as beneficial as we think they are (specifically, those addressing problems that can be usefully approximated through mechanistic methods). For software, however, this fraction is much smaller than it is for the traditional products, probably less than 1 percent.

Like business and academic mechanism, then, software mechanism asks us to change: we must limit ourselves, in all software-related activities, to what can be accomplished with mechanistic concepts alone. In reality, we *can* develop non-mechanistic capabilities – knowledge, skills, experience – so we *can* create non-mechanistic software. But, the elites tell us, these capabilities are unreliable, and it is best to forgo them. Software mechanism, thus, is totalitarian – because it asks us to conform to its tenets. We must replace our natural, non-mechanistic capabilities with mechanistic ones. And we must replace our intuitive definition of software expertise – the utmost that human minds can attain – with a degraded one: the capability to understand mechanistic software concepts.

Now, if software were indeed just a new kind of product, software totalitarianism would mean only that the elites have found one more way to impoverish our existence. The harm, in other words, would be no worse than the harm caused by the traditional forms of academic and business totalitarianism. Software, however, is not just another product. Because of its versatility, software must be treated as a new phenomenon – a phenomenon comparable in potency to the phenomenon of language. Like language, software permits us to represent the world through symbols, and to communicate with it. It is their ability to generate *complex* structures, and hence to represent the world as it actually is, that distinguishes language and software systems from ordinary products. And it is precisely this ability that is lost when they are reduced to mechanistic systems. They behave then just like ordinary products, and they cease to mirror the world accurately.

Software is different from the expedients promoted by the traditional elites, therefore, because of its potential as a means of domination. When restricting us to mechanistic software, the elites restrict us in effect to thinking like automatons. Before, a certain type of product could be used by an elite to restrict only one aspect of our life; and no one was affected by more than a few types of product at a time. But as we get to depend on computers in more and more activities, the software elites have the opportunity to restrict practically every aspect of our life.

In conclusion, a society can become totalitarian simply by pursuing mechanistic ideas. And we saw that Talmon's model of totalitarianism can explain this phenomenon. There is a clear progression: from mechanistic theories that affect just academic bureaucrats, to products that affect many individuals but in a limited way, to software concepts that affect all members of society, in all their activities. In the past, only political institutions could enforce an ideology on such a large scale, and this is why Talmon's model describes *political* mechanism. But if software is now comparable in its scope to politics – if, that is, software concepts affect society as drastically as do the traditional political concepts – this model should also depict *our* situation.

Politically, a society becomes totalitarian when millions of people are forced to conform to mechanistic social concepts. And if the same people are forced to conform instead to mechanistic *software* concepts, the result is bound to be the same. For, in both cases, the effect is to restrict these people to mechanistic performance in every aspect of their life. Thus, all we have to do in order to use Talmon's model for today's society is substitute software for politics. The model explains then why our widespread adoption of mechanistic software ideas is causing a drift toward totalitarianism.

2

Orwell's model, we saw, explains our progression toward totalitarianism by pointing to the steady degradation in social values: the growing politicization of the economy, the growing corruption of the elites, and, especially, the growing use of language to control minds. The elites are promoting mechanistic notions; in addition, they are restricting language to high levels of abstraction, and this prevents us from recognizing how limited the mechanistic notions actually are.

The greatest value of Orwell's model, however, lies in helping us to understand the phenomenon of *software* totalitarianism – a phenomenon that did not even exist in his time. Thus, while warning us about the growth of traditional totalitarianism in the 1940s, Orwell created a model that can be used to explain the growth of software totalitarianism today.

Recall the similarity of language and software. Both function as systems of representation and communication, so both allow us to create complex structures that mirror the world. But, above all, it is our capacity to process these structures together with various knowledge structures present in our minds that permits language and software to represent the world. With language, we saw, only by starting at low levels of abstraction can a message represent the world accurately. And the same is true of software: only by starting with low-level elements can a software application represent our affairs accurately. With both systems, when starting from high levels we lose the low-level interactions between structures, and hence many combinations of elements. The values we see at the top level are then only a fraction of all possible values. Thus, our knowledge of the world can be impoverished by restricting *software* to high levels of abstraction just as it can be by restricting language.

Orwell criticized the use of high-level linguistic elements – expressing an idea by combining ready-made phrases instead of starting with words. But this style, the essence of language impoverishment, is precisely the style recommended by our software experts for *programming*. All programming theories claim that applications can be designed as modules within modules, just like the appliances built in a factory; so programming can be based on the idea of software subassemblies – ready-made, high-level pieces of software. We have reached, therefore, an absurd situation: what we recognize as harmful in language – the mechanistic mode of communication – we strive to attain in software. Take, for instance, the following remark: "Political writing in our time consists almost entirely of prefabricated phrases bolted together like the pieces of a child's Meccano set."[1] Orwell is using here the metaphor of a child's building blocks to mock the high-level, vacuous linguistic style employed by propagandists. He could not have imagined that a few decades later, in programming, similar metaphors would be *seriously* used by experts to *praise* the high-level style.

Mechanism, we saw, destroys minds by reducing knowledge and thought to the level of machines. And mechanistic language enhances this process, because language structures interact with knowledge structures. Restricted to mechanistic thinking, we cannot develop *complex* knowledge. We become, in effect, automatons. We also saw that it is in the interest of the elites to maintain a mechanistic culture, because this guarantees ignorance and dependence: the mechanistic concepts promoted by the elites prevent us from using our minds; we cannot solve our complex problems, and we believe that the only answer is to adopt even more of these concepts; but this only increases our dependence on the elites and on mechanistic concepts, further degrading our minds, in a vicious circle.

And if software fulfils the same social role as language, a dependence on mechanistic software is bound to have the same effect. The software elites

[1] George Orwell, "The Prevention of Literature," in *The Collected Essays, Journalism and Letters of George Orwell*, vol. 4, eds. Sonia Orwell and Ian Angus (London: Penguin Books, 1970), p. 89.

promote mechanistic software concepts in order to keep us ignorant and dependent on their devices. But it is not only in software-related matters that we remain ignorant. Because we depend on computers in every aspect of our life, software structures, like language structures, interact with the knowledge structures present in the mind. So the software elites are controlling our minds through software just as the traditional elites are through the older concepts and through language. The traditional elites, we saw, have more power between them than our political institutions. And, since their ideologies are totalitarian, our society is becoming totalitarian despite its democratic foundation. But the software elites are even more powerful than the traditional ones, so our progression toward totalitarianism is now even faster. They are more powerful because they are permitted to control, not an ordinary concept, but software.

The manipulation of language by the traditional elites forms, in the end, only a small part of our entire use of language. The manipulation of software by the software elites, on the other hand, is almost total. Only in the imaginary society of *Nineteen Eighty-Four* is the enforcement of language mechanism comparable to the enforcement of software mechanism in our own society. Thus, while no elite in a real society can ever have enough power to manipulate *language* to such an extent, our software elites already have this power in manipulating *software*. And, if even the relatively mild *language* mechanism currently imposed on us can degrade our minds, and can foster a totalitarian culture, it is safe to predict that complete *software* mechanism will cause widespread ignorance, and will bring about full-fledged totalitarianism.

3

Recall Orwell's observation that even a democratic society can be corrupted by totalitarianism (p. 851). Under the guise of administration, or education, or marketing, every elite is distorting knowledge in order to promote its ideas. The process of communication becomes a process of indoctrination: people are seen as little more than automatons that must be programmed to accept whatever ideology serves the interests of the elites.

And now we must add to this the corruption caused by *software* totalitarianism: universities are teaching pseudoscientific software notions, instead of fostering professionalism and responsibility; programmers rely on worthless theories, development tools, and ready-made pieces of software, instead of improving their skills; software systems are routinely promoted through testimonials and success stories – means of deception traditionally employed to promote useless consumer products; important software decisions are being made following the advice of charlatans acting as consultants, or lecturers, or gurus; workers in all fields are spending more and more of their time with software-related problems, instead of practising their profession; respected computer associations are promoting software notions that serve the interests of the software elite, not society; major government projects are abandoned after spending vast amounts of public money, while the incompetents responsible for these failures continue to be seen as software experts.

So it is true that we can be corrupted by totalitarianism even if we don't live in a totalitarian country. But Orwell made this observation *before* we had a software elite and a software bureaucracy. To bring the observation up to date, we must say that the corruption caused by software totalitarianism can *exceed* that found in a totalitarian country. It is the *nature* of software, its similarity to language, that makes this possible: we are becoming dependent on software in practically every activity; and if at the same time we are being restricted to *mechanistic* software concepts, it is only natural that we are increasingly thinking like machines. The incompetence, the irresponsibility, the apathy, the delusions we suffer from – this is exactly what we should expect to find in a society where people are prevented from using their minds.

❖

It should be obvious, then, why I chose Talmon's and Orwell's models of totalitarianism. While dealing with political matters, their generality makes them equally suitable for the study of *software* matters. All we have to do is substitute software for politics, and these models will depict *software* totalitarianism. In the past, only a political elite had sufficient power to control society. No one could have imagined that one day we would invent something as potent as software, and that we would permit an elite to control it. Thus, if the software elite has the same power as a political elite, it is not surprising that our software ideology can be depicted with political models.

Another fact explained by the two models is why modern societies, founded upon liberal and democratic principles, end up nevertheless drifting toward totalitarianism. The reason is that the modernity which engenders democracy tempts us at the same time to accept blindly all mechanistic notions. Because of its successes in the exact sciences, we also see mechanism as the answer to our political, social, and economic problems. An elite can gain our support, therefore, by promising us simple, mechanistic solutions to these problems. So mechanism leads to utopianism: the belief that the methods we use in science and in engineering can help us to create a perfect society. But the only way to make a mechanistic ideology work is by enforcing conformism. Utopianism, thus, leads to totalitarianism: since a mechanistic ideology does not reflect human nature, *we* must be modified to match the ideology. Still, the elite says, this is not coercion but education. The ideology is based on scientific principles; so, if we accept science we must also accept totalitarianism. In a modern, efficient society, preferring individuality to conformism is a sign of maladjustment.

It makes little difference, thus, what type of mechanistic ideology one starts with – political, religious, business, educational, or software; if implemented on a large scale, it is guaranteed to become totalitarian. And this is why the same model can depict any type of totalitarianism. To demonstrate this kinship for software, I include below a passage from Talmon's book – the text where he describes the difference between the liberal and the totalitarian types of democracy (we discussed this earlier, see pp. 834–835). Read the passage, however, by substituting the word "software" for the twelve occurrences of "politics" and "political" (which I emphasized). And, with this change, the text describes perfectly our totalitarian *software* ideology.

> The essential difference between the two schools of democratic thought as they have evolved is not, as is often alleged, in the affirmation of the value of liberty by one, and its denial by the other. It is in their different attitudes to *politics*. The liberal approach assumes *politics* to be a matter of trial and error, and regards *political* systems as pragmatic contrivances of human ingenuity and spontaneity. It also recognizes a variety of levels of personal and collective endeavour, which are altogether outside the sphere of *politics*.
>
> The totalitarian democratic school, on the other hand, is based upon the assumption of a sole and exclusive truth in *politics*. It may be called *political* Messianism in the sense that it postulates a preordained, harmonious and perfect scheme of things, to which men are irresistibly driven, and at which they are bound to arrive. It recognizes ultimately only one plane of existence, the *political*. It widens the scope of *politics* to embrace the whole of human existence. It treats all human thought and action as having social significance, and therefore as falling within the orbit of *political* action. Its *political* ideas are not a set of pragmatic precepts or a body of devices applicable to a special branch of human endeavour. They are an integral part of an all-embracing and coherent philosophy. *Politics* is defined as the art of applying this philosophy to the organization of society, and the final purpose of *politics* is only achieved when this philosophy reigns supreme over all fields of life.[2]

[2] J. L. Talmon, *The Origins of Totalitarian Democracy* (New York: Praeger, 1960), pp. 1–2 (italics added). A similar effect is achieved if substituting "software" for politics-related or language-related words in Orwell's writings (the quotations on pp. 849 and 856, for instance).

Let us analyze these statements. The "sole and exclusive truth" in software is the ideology of software mechanism; in particular, the belief that software is a kind of product, so the most effective way to create and use software is through the devices supplied by software companies. Just like the political theories, the software theories are defended by invoking their mechanistic foundation. Mechanism is invalid both in politics and in software, and its failure in these domains is obvious. But if we accept blindly the mechanistic ideology, we must also accept mechanistic political and software systems.

The state "to which men are irresistibly driven, and at which they are bound to arrive" is the state where software mechanism is universally accepted, and no other form of programming or software use is envisaged. And software messianism is the belief that such a state is imminent: at any given moment, the latest concept, theory, or system is perceived as the revolution that would finally deliver us from all software evils.

Recall the notion of a general will – the hidden, natural qualities said to inhere in all of us (p. 837): by enforcing conformism, the political elite will bring out these superior qualities, and thereby create a perfect society. Similarly, we may call the mechanistic software concepts a general *software* will. We all possess from birth such qualities as the appreciation of structured programming, relational databases, high-level environments, and ready-made applications. But these natural, superior qualities are masked by the inefficient habits we acquire as individuals. So, by enforcing software conformism, the software elite helps us to attain our natural, higher self. The secret for becoming perfect programmers and users is an unwavering acceptance of the mechanistic software ideology.

Similarly to its political counterpart, the *liberal* software view recognizes "a variety of levels of personal and collective endeavour" for which software devices are ineffective, or unnecessary; and the *totalitarian* software view recognizes "only one plane of existence," our software-dependent activities.

Under *political* totalitarianism, all social and personal affairs are modified to reflect the prevailing ideology, in order to permit the elite to directly control the life of every citizen. But to make this possible, the scope of politics is extended to encompass every aspect of human life. Politics, therefore, becomes more important than it ought to be. It grows into an exaggerated, morbid preoccupation, and every other activity in society is made subordinate to it. Under *software* totalitarianism, it is software – as expressed through the mechanistic ideology – that becomes more important than it ought to be, and grows into a morbid preoccupation. The scope of software is extended "to embrace the whole of human existence."

Thus, by invoking progress, or efficiency, or standards, the software elite attempts to replace every type of human knowledge with a software device. The only thing we need to know from now on, we are told, is how to operate these devices. Through these devices, then, the elite directly controls our capabilities, our values, our beliefs. The notion of personal skills and experience is becoming obsolete, as we are all expected to accomplish about the same thing: whatever can be done by combining the features found in the latest software devices. Instead of using our time to gain knowledge and experience, we waste it with software-related preoccupations; instead of using it to solve real professional and personal problems, we waste it solving specious, software-related problems.

The software elite treats software, thus, not as "a set of pragmatic precepts or a body of devices applicable to a special branch of human endeavour," but as "an integral part of an all-embracing and coherent philosophy" – the mechanistic philosophy. More precisely, software is seen as "the art of applying this philosophy to the organization of society." The elite treats software, in other words, not as what it really is – the means to represent the world with computers – but as a way to control society. And it does this by enforcing the mechanistic software ideology; specifically, by making us dependent on software devices. Since this kind of software cannot solve our real, complex problems, we end up spending more and more time with senseless pursuits. Simply by inducing us to squander our time on useless activities, then, the elite ensures our permanent ignorance; and this in turn ensures a growing dependence on software devices, in a process that feeds on itself. But "the final purpose" of software will only be achieved "when this philosophy reigns supreme over all fields of life"; that is, when every aspect of human existence is controlled through software devices.