TECHNOLOGY AND COMPLEXITY

Subrata Dasgupta

1. *Introduction: systemic and epistemic complexity*

In a seminal paper first published in 1962, Herbert Simon articulated the nature of complexity as it is manifested in both natural and artificial systems.[1] A system, according to Simon, is said to be complex if it is composed of a large number of parts or components that interact in nontrivial ways. Even if one knows the properties of the components, one may not be able to infer in any obvious way the properties or behavior of the system as a whole. In this sense, a complex system is more than the sum of its parts.

Let us refer to Simon's notion by the term *systemic complexity*. Entities that are systemically complex frequently manifest the property of being hierarchically organized.[2] More generally, since just after the Second World War, Ludwig von Bertalanffy, Norbert Wiener and others have suggested that entire classes of complex natural and artificial systems share certain striking properties as, e.g., homeostasis.[3] And in the past two decades, a variety of theorists have concerned themselves with

---

[1] H.A. Simon, "The Architecture of Complexity", *Proc. Amer. Phil. Soc.*, 106, Dec. 1962, pp 467-82. Reprinted in H.A. Simon, *The Sciences of the Artificial*, MIT Press, Cambridge, MA, 1996, (3rd Edition), pp 183-216.

[2] L.L. Whyte, A.G. Wilson & D. Wilson (eds.), *Hierarchical Structures*, American Elsevier, New York, 1969. H.H. Pattee (ed.), *Hierarchy Theory*, George Braziller, New York, 1973.

[3] L. von Bertalanffy, *General System Theory*, George Braziller, New York, 1968 (3rd Edition). E. Lazlo, *The Relevance of General Systems Theory*, George Braziller, New York, 1972. N. Wiener, *Cybernetics*, MIT Press, Cambridge, MA, 1961, (2nd Edition).

complex dynamical systems that are 'chaotic' in the special sense that the behavior of such systems may change drastically with very small changes in the initial conditions.[4]

Artifacts -- by which I mean specifically, useful things that are "produced or consciously conceived in response to some practical need, want or desire[5] -- are clearly more or less complex in the systemic sense. But artifacts also possess another interesting property which is connected to complexity: like organisms, they manifest *evolution*.

The term 'technological evolution' carries with it at least five implications. First, for some thinkers, the evolution of artifacts is the source of technological *diversity*.[6] Thus, for example, George Basalla was drawn to the evolutionary idea in technology by the analogy of the diversity of artifacts with that of organisms.[7] For Basalla, technological diversity is explicable by positing an evolutionary process.

Second, artifacts are said to evolve in the sense that technological change, even radical change such as the invention of entirely new forms, is postulated to be *gradualistic*. A well known early proponent of this notion was the anthropologist A. Lane Fox Pitt-Rivers who showed the evolutionary relationship amongst a variety of Australian aboriginal weapons such as shields, clubs and boomerangs.[8] Pitt-Rivers was drawn to his insight by the Darwinian analogy. Recently, appealing to the biological notion of phylogeny, I have suggested that *technological evolution* follows a *phylogeny law*: That is, every act of invention or design has a

---

[4] P. Cvitanovic (ed.), *Universality in Chaos*, Adam Hilger Ltd., Bristol, 1984.

[5] S. Dasgupta, *Technology and Creativity*, Oxford University Press, New York, 1996, p. 9.

[6] G. Basalla, *The Evolution of Technology*, Cambridge University Press, Cambridge, 1988. J. Mokyr, "Evolution and Technological Change : A New Metaphor for Economic History", in R. Fox (ed.), *Technological Change : Methods and Themes in the History of Technology*, Harwood Academic Publishers, U.K., 1996. J. Mokyr, *Lever of Riches*, Oxford University Press, New York, 1990, pp 273-300. H. Petroski, *The Evolution of Useful Things*, Alfred A. Knopf, New York, 1992. Dasgupta, *Technology and Creativity, op cit,* pp 122-149.

[7] Basalla, *op cit,* p. 14.

[8] P. Steadman, *The Evolution of Designs*, Cambridge University Press, Cambridge, 1979, pp 86-94.

phylogenetic history[9] -- where by the 'phylogeny of an artifact' I refer to the "linked network of mature artifacts or forms that lead up to the invention of a given artifact".[10]

Third, evolution in the realm of artifacts also refers to the nature of the *cognitive process* whereby, beginning with a set of objectives, aims or desired characteristics, a particular artifact comes into being either symbolically (that is, as a design) or in actual, operational form. Likening this process to that of biological ontogeny, I have referred to this as the *ontogenetic evolution* of an invention or design.[11] Some writers, for example the engineer-historian Walter Vincenti, have invoked a Darwinian explanation for the ontogeny of artifacts.[12] Others, such as myself and the computer scientist B. Chandrasekaran, have suggested a non-Darwinian evolutionary schema in which the cognitive process of invention entails one or more cycles of hypothesis creation, testing and modification.[13] I have called this the *hypothesis law of design*.[14]

Fourth, technological evolution suggests a sense of *progress* -- the implication that the transformations of an artifactual form through successive stages in the course of an evolutionary pathway (that may extend over centuries) result in the improvement of the artifact and, more significantly, in the improvement of the social, cultural and economic well-being of humanity. The Darwinian overtone to the linking of technological evolution with progress is obvious. Amongst recent writers, the economic historian Joel Mokyr is an advocate of this linkage.[15] A more sceptical position is that of George Basalla who has very reasonably

---

[9] Dasgupta, *op cit*, p. 146.

[10] Dasgupta, *op cit*, p. 123.

[11] S. Dasgupta, *Design Theory and Computer Science*, Cambridge University Press, Cambridge, 1991, pp 77-80. Also, Dasgupta, *Technology and Creativity, op cit*, p. 123.

[12] W.G. Vincenti, *What Engineers Know and How They Know It*, Johns Hopkins University Press, Baltimore, MD, 1992.

[13] Dasgupta, *Technology and Creativity, op cit*, p. 66. B. Chandrasekaran, "Design Problem Solving : A Task Analysis", *AI Magazine*, 11, 1990, pp 59-71.

[14] Dasgupta, *ibid*. Also, S. Dasgupta, "Testing the Hypothesis Law : The Case of the Britannia Bridge", *Research in Engineering Design*, 6, 1, 1994, pp 38-57.

[15] Mokyr, *Lever of Riches, op cit*, pp 15, 301-304.

suggested that technological progress as a correlate of artifactual evolution is meaningful only under very narrow contexts and strictly technical objectives, and providing one disassociates improvements in artifacts from social, cultural and economic progress.[16] In other words, technological evolution within narrowly circumscribed domains may indeed lead to artifactual improvements with respect to specific technical goals, but that may have no bearing on the well-being of humanity. This is consistent with the economist-philosopher Amartya Sen's broader critique that the Darwinian idea of progress insofar as it applies to the fitness characteristics of a species does not entail improvement in the quality of human lives.[17]

Fifth and finally, technological evolution carries with it the idea of *growth in complexity*; that is, artifacts evolve from the simple to the complex, from the less to the more complex. Here again, there are parallels drawn between the natural and the artificial since biological organisms are considered to have evolved in complexity.[18]

And so, after this lengthy preamble, we are led to the central topic of this paper. My thesis is that systemic complexity does not tell the whole story of the evolution of technological things. Indeed, on occasion, it can be positively misleading. I suggest that there is another deeper, more compelling kind of complexity in the world of made things, and that is the *richness of the knowledge that is embedded in an artifact*. I shall call this *epistemic complexity*. It consists of the knowledge that both contributes to, and is generated by, the creation of an artifact. This paper is, thus, an examination of the nature of the epistemic complexity of artifacts and its relationship with systemic complexity; it also explores the implications of epistemic complexity for the history of technology and its connection with technological creativity.

---

[16] Basalla, *The Evolution of Technology, op cit,* pp 210-18.

[17] A.K. Sen, "On the Darwinian View Of Progress", *London Review of Books*, Nov. 5, 1992, Vol. 14, No. 21.

[18] J.T. Bonner, *The Evolution of Complexity by Means of Natural Selection*, Princeton University Press, Princeton, NJ, 1988. See, however, D.W. McShea, "Complexity in Evolution : A Skeptical Assessment", this issue, for an alternative view of the relationship between complexity and evolution in the natural world.

2. *Technological knowledge and epistemic complexity*

In an earlier study of the nature of technological creativity, I have pointed out that invention or design is a *knowledge rich* cognitive process; that is, the creative technologist is armed with a rich body of interconnected knowledge which he or she brings to bear in any particular cognitive act of invention.[19] Some of this knowledge is not specific to technology at all but is shared by all cognitive beings -- as, for example, common rules of inference or general mental tools for planning and problem solving. More specific *technological knowledge* is itself quite heterogenous. It includes mathematics, the basic sciences, and engineering theory. But these types of knowledge have entered the technologist's 'knowledge base' primarily in the past two centuries, since the Industrial Revolution in fact.[20] In the long history of technology, there is, however, one kind of knowledge which, following the scientist-philosopher Michael Polanyi, I call *operational principles*.[21] This term refers to any proposition, rule, procedure, concept or heuristic about artifactual properties or characteristics that facilitate the creation, manipulation and modification of artifacts.[22]

Here then, we have the basis for identifying more precisely the concept of epistemic complexity. The act of conceiving, creating and bringing into practical form *any* artifact entails the deployment, on the part of the inventor/designer, of his or her 'knowledge base'. Knowledge is, thus, an input to the cognitive process of invention or design. The outcome of that process is an artifactual form. *The latter is also knowledge*: specifically, a design embodies and encapsulates one or more operational principles. And, in the case of true invention, when the artifactual form is *original* in some significant sense, the operational principles it encodes constitute genuinely *new* knowledge. Thus, what characterizes the highest form of technological creativity -- that is, what distinguishes invention or what the engineer-historian Walter Vincenti

---

[19] Dasgupta, *Technology and Creativity, op cit,* esp. Chapters 9 and 10.

[20] A.E. Musson and E. Robinson, *Science and Technology in the Industrial Revolution,* University of Manchester Press, Manchester, 1969.

[21] M. Polanyi, *Personal Knowledge*, University of Chicago Press, Chicago, 1962.

[22] Dasgupta, *op cit*, pp 157-158.

called 'radical design'[23] from 'normal design' (also a Vincenti term) is characterized by two epistemic features: (a) The fact that genuinely new knowledge is produced, predominantly in the form of operational principles; and (b) The fact that old knowledge is put to use in unexpected or surprising ways. And what seems to most characterize technological creativity and the originality of its products is the *richness (i.e., the amount, variety and newness) of the knowledge embedded in the artifact.* It is this embedded knowledge that I call the epistemic complexity of an artifact.

3. *Complexity in normal design*

At first blush, it might seem that there is a direct connection between systemic and epistemic complexities. It might be assumed that an artifact having many components that interact in complicated ways and produce potentially unexpected and obscure behavior is also one which embeds a rich body of knowledge. It is important to keep in mind, however, that the epistemic complexity of an artifact (as I have characterized it) arises not simply because of the amount of knowledge but by the combination of old knowledge in *unexpected* and *surprising* ways and the *richness* of the new knowledge which such synthesis gives rise to.

Using examples, I shall show, however, that there may not be a connection between the two kinds of complexities. Rather, the history of technology reveals important instances of the following possibilities: (a) Artifacts that are systemically complex but epistemically simple (relatively speaking); (b) Artifacts that are systemically complex and, *consequently*, epistemically complex also; and (c) Artifacts that are systemically simple but epistemically complex. In this and the next sections, I will discuss the first of these scenarios; the second and third are discussed respectively in later sections.

Consider a 'mature' technology. Suppose that its operational principles and even theoretical basis are extremely well understood. The artifacts belonging to this technology may well be systemically complex. However, given that the technology is 'mature' we may expect that its

---

[23] Vincenti, *What Engineers Know* . . . , *op cit*, pp 8-9.

artifacts have been designed and made many times in the past, and that each new version differs from its predecessors only in some specific set of parameters. This scenario is a special case of Vincenti's depiction of *normal design* wherein, as he put it

> The engineer . . . knows at the outset how the device in question works, what are its customary features and that, if properly designed along such lines, it has good likelihood of accomplishing the desired task.[24]

In normal design, the overall composition of the artifact is known *a priori*. The designers of an aircraft engine prior to the advent of turbojet, for example,

> . . . took it for granted that the engine should be piston driven by a gasolene-fueled, four-stroke, internal combustion cycle. The arrangement of cylinders for a high-powered engine would also be taken as given . . . so also would other, less obvious features . . . The design problem . . . was one of improvement in the direction of decreased weight and fuel consumption or increased power or both.[25]

In such situations, very little *significant* new knowledge may be produced in the act of creating the artifact. Old knowledge is used in almost the same way as in the past. There is little anticipation of surprise. (This does not mean, of course, that normal design *always* precludes surprise; it could be the case, e.g., that a slight change in one of the parameters will lead to a 'chaotic' effect totally unforseen. In technology, this would be analogous to the occurrence of 'anomaly' in the practice of normal science *a la* Thomas Kuhn[26] and may lead, possibly, to the situation of radical design or invention.) The systemic complexity of artifacts produced by normal design may be considerable, as anyone unfamiliar with such an artifact will realize when attempting to analyze and understand its behavior. But the epistemic complexity, in contrast, may be

---

[24] Vincenti, *op cit*, p. 7.

[25] Vincenti, *op cit*, p. 8.

[26] T.S. Kuhn, *The Structure of Scientific Revolutions*, The University of Chicago Press, Chicago, 1970, (Revised Edition).

quite unimpressive.

In fact, it is this characteristic that makes it at all possible to deploy the computer as an agent in the design process. Researchers in the application of artificial intelligence (AI) to design have developed techniques that rely heavily on the idea of normal design. For instance, David Brown and B. Chandrasekaran have shown how the air cylinder -- a piston and rod arrangement which, by moving backward and forward against a spring within a tube creates a to-and-fro movement of some other connected device -- can be automatically designed by an 'expert system' (a computer program that has access to the kind of expert knowledge a mechanical designer may possess) using the idea of normal design. Air cylinders have a well-defined hierarchical form. Starting with this 'generic' form, their program makes the necessary changes, fills in the details and establishes the specific parameter values in order to meet the requirements for a specific air cylinder.[27]

4. The role of 'style' in the management of complexity

Epistemic complexity is also avoided when the designer takes recourse to well-established *styles* and adapting a chosen style to the specific needs of the technological problem at hand. A style is any complex of characteristics or features that sets apart one group of artifacts from another in the same class.[28] It encapsulates a particular *integrated* body of past knowledge for a particular class of artifacts. By selecting a particular style for the solution of a design problem, the designer avoids the need for a full-blown, ontogenetic cognitive process for the creation of the desired artifact. Much of the knowledge that will contribute to the artifact is already there, in the style itself. It is not to be expected, therefore, that

---

[27] D.C. Brown and B. Chandrasekaran, "Knowledge and Control for a Mechanical Design Expert System", *Computer*, 19, 7, July 1986, pp 92-100.

[28] The role of style in design was first explored in a general way in H.A. Simon, "Style in Design", in C.M. Eastman (ed.) *Spatial Synthesis in Computer-Aided Building Design*, John Wiley & Sons, New York, 1975. I have further explored the topic in some depth in the context of computational artifacts (viz., computers, software, algorithms and computer languages) in S. Dasgupta, *Design Theory and Computer Science*, Cambridge University Press, Cambridge, 1991, pp 310-22.

the artifact as a whole will be significantly original (though parts of it may well be so). Epistemic complexity is largely bypassed; and yet the artifact may still be systemically complex by its very nature.

Architecture obviously provides profuse examples of such style-based design -- indeed, books on architecture and its history are often organized along stylistic lines.[29] Gothic, perhaps the most recognizable of architectural styles with its pointed arches, soaring rib vaults and flying buttresses, was the dominant form of cathedral architecture in the twelfth and thirteenth centuries and continued to prevail well into the Renaissance era. The first Gothic cathedral was built in St. Denis near Paris between 1140 and 1144.[30] Paraphrasing Nickolaus Pevsner, it constituted a revolutionary architectural invention, both aesthetically and in its engineering. The basic form of the Gothic style was in place by the 1160s when Notre Dame was built in Paris. Thereafter, the cathedrals that sprang up all over France and then in England over the next hundred years were elaborations of the style: the vaults soared higher, the ribbed forms of the vaults became more elaborate and ornate, the flying buttresses more profuse.[31] Many of the later cathedrals that architectural historians now regard as exemplars of the Gothic style -- including those in Chartres (built c. 1194), Rheims (c. 1211), Amiens (c. 1220) and Beauvais (c. 1247) -- no doubt embodied new knowledge as their builders endeavored to lend aesthetic distinction or practical advantage to the structures. (For example, the cathedral at Beauvais had passageways at five levels to facilitate access in case of fire and for maintenance.[32]) But they were all, fundamentally, recognizably Gothic in style. The later cathedral builders drew upon this knowledge and added to it. Thus, the celebrated *Sketchbook* of the architect-engineer Villard de Honnecourt (fl. 1225-1250) shows not only drawings of the cathedrals at Rheims and Laon, but also describes a geometric procedure used to produce the plan of the tower of

[29] N. Pevsner, *An Outline of European Architecture*, Penguin Books, Harmondsworth, Middlesex, 1963.

[30] Pevsner, *op cit*, p. 89.

[31] A. Pacey, *The Maze of Ingenuity : Ideas and Idealism in the Development of Technology*, MIT Press, Cambridge, MA, 1992, (2nd Edition), pp 19-22.

[32] J. Gimpel, *The Medieval Machine*, Penguin Books, New York, 1977, pp 122-23.

Laon Cathedral.[33] Villard also provided instructions to include passage-ways "to allow circulation in case of fire" and gutters "to carry off the water".[34] Other surviving documents from builders' guilds and lodges of the time reveal details of design procedures for other architectural components characterizing the Gothic style, as for example, pinnacles.[35]

In sum, an artifactual style constitutes codification of knowledge acquired during the most creative, inventive stage in the history of that artifact. It is a means for reducing (but not eliminating) epistemic com-plexity even when that 'species' of artifact evolves towards greater sys-temic complexity. In this sense, the adoption and elaboration of style in technology is a version of normal design.

5. On the causal connection between systemic and epistemic complexities: the case of Multics

A direct causal connection between systemic and epistemic complexities -- that is, the situation in which a high level of epistemic complexity is a *consequence* of a high level of systemic complexity -- can arise in some acts of invention. An example is the development, in the 1960s, of the computer operating system known as *Multics*.

Generally speaking, an operating system -- software that automatical-ly manages the resources, supports applications software and controls the proper functioning of the computer as it goes about its multifarious tasks -- is arguably one of the most systemically complex species in the soft-ware universe. Indeed, it is arguably among the most systemically com-plex entities in the technological universe. Thus, when an operating system is conceived and designed to *be* significantly original, its systemic complexity directly causes epistemic complexity.

Multics is a case in point. It was designed and built at the Massachusetts Institute of Technology (MIT) in collaboration with Bell

[33] T. Bowie (ed.), *The Sketchbook of Villard de Honnecourt*, Indiana University Press, Bloomington, Indiana, 1951.

[34] Bowie, *op cit*, p. 100.

[35] W. Addis, *Structural Engineering : The Nature of Theory and Design*, Ellis Horwood, Chichester, West Sussex, 1990, pp 124-126.

Laboratories and General Electric in the mid to late 1960s as a time-sharing operating system for the General Electric GE645 mainframe computer.[36] In its mature state, Multics consisted of some 1500 modules for a total of approximately one million lines of machine instructions.[37] Its structure was a direct outcome of its overall objective: to create a general computer utility analogous to electrical power and telephone utilities, which would run continuously and reliably and provide a comprehensive range of services to a population of users interacting with it through remote terminal access. Multics was, thus, conceived not just as a powerful computational artifact but also as a *technological system.*[38] The designers refined this all-encompassing objective into a collection of more specific capabilities which the system would have to possess. These included: time-sharing facilities, an elaborate information storage system that would protect a user's programs and data from unauthorized access, reliability measures, the provision of a sophisticated programming environment for users to work in, support for a number of computer languages, inter-user communication facilities (a forerunner of the e-mail), maintenance and monitoring facilities, features to enhance the management of the system's users, and flexibility so that the system could absorb new technologies and changes in user expectations.

As this list suggests, systemic complexity was built into the very requirements demanded of the system. Furthermore, though Multics was not the first time-sharing system to be built, it was the first experiment in creating a comprehensive computer utility -- a "community computer facility".[39] Multics had to be invented, not merely designed.

It was because it had to be invented that the systemic complexity inherent in its goals gave rise to the epistemic complexity of Multics as

---

[36] E.I. Organick, *The Multics System : An Examination of its Structure,* MIT Press, Cambridge, MA, 1972. Later, the GE645 and Multics came to be products of Honeywell and were marketed as Honeywell products.

[37] F.J. Corbato, J.H. Saltzer and C.T. Clingen, "Multics -- The First Seven Years", in P. Freeman, *Software System Principles,* SRA, Chicago, 1975, pp 556-577; esp. p. 560.

[38] For a discussion of technological systems, see T.P. Hughes, "The Evolution of Large Technological Systems", in W.E. Bijker, T.P. Hughes & T.J. Pinch (eds), *The Social Construction of Technological Systems,* MIT Press, Cambridge, MA, 1987, pp 51-82.

[39] Corbato, Saltzer and Clingen, *op cit,* p. 557.

an artifact. Thus, it had an elaborate phylogeny; it drew upon: (a) An operating system called CTSS ('Compatible Time-Sharing System') also built at MIT between 1960 and 1963, which was, in fact, the first operational time-sharing system.[40] (b) Two alternative schemes called, respectively, 'paging' and 'segmentation' for providing the user the illusion of unlimited or 'virtual' memory; both these had been invented elsewhere in the early 1960s. (c) A technique called 'multiprogramming', invented almost contemporaneously, whereby several user programs simultaneously share main memory and the computer's central processor is passed around amongst them so as to keep the processor always busy. (d) Schemes devised in the early to mid 1960s for protecting a user's program and data from unauthorized access by other users.

The designers of Multics did not just draw upon these earlier inventions; they combined, expanded and generalized them, and by this synergy, created a significantly original product. Furthermore, the development of Multics constituted a significant experiment in the use of abstract (or 'high-level') computer languages to write a large operating system[41] and in the application of iterative design -- "A straightforward approach is to create a perhaps crude and incomplete system, begin to use it and to observe the behavior. Then on the basis of the observed difficulties, one simplifies, redesigns and refines the system".[42] (One cannot imagine a clearer manifestation of the idea of ontogenetic evolution in technology.)

Thus, the Multics effort generated, and was an encoding of, significant new knowledge. And though it was not a hugely successful commercial system, the knowledge it spawned became a significant part of the knowledge base of operating system designers. The situation was not unlike that of the wrought-iron, tubular Britannia Bridge built in Wales by Robert Stephenson and William Fairbairn in the 1840s: The

---

[40] M.V. Wilkes, *Time Sharing Computer Systems*, Macdonald & Jane's, London / American Elsevier, New York, 1975 (3rd Edition), p. 2.

[41] F.J. Corbato, "PL/I as a Tool for System Programming", *Datamation*, May 1969, pp 68-76.

[42] F.J. Corbato & C.T. Clingen, "A Managerial View of the Multics System Development", in P. Wegner (ed.) *Research Directions in Software Technology*, MIT Press, Cambridge, MA, 1979, pp 139-158; esp. p. 140.

latter led to only a few more bridges of its type; but its very design and construction produced valuable knowledge about the behavior and properties of wrought iron structures.[43] The Multics system provides an exemplary case study of technological creativity in which systemic complexity is inherent in the aims and objectives of the artifact, which in turn necessitate a combination of a rich phylogeny on the one hand and the generation of new knowledge on the other. It illustrates how systemic complexity gives rise to epistemic complexity.

6. On the complexity of the 'first technology'

We have seen that styles and the practice of normal design allow the inventor/designer to draw upon previously proven prior knowledge to manage and diffuse the epistemic complexity of a new artifact that may nevertheless be systemically complex. We have also seen an example wherein the invention of a radically original artifact with built-in systemic complexity necessitated a high level of epistemic complexity to be embodied in that artifact. But there are also artifacts found in the history of technology that reveal a coupling of systemic *simplicity* with epistemic complexity. Such artifacts are quite simple according to Simon's criterion but are nonetheless embodiments of considerable original knowledge.

It may seem surprising to use stone tools to ornament a discussion of technological complexity. And yet, it is precisely because they constitute the 'first technology'[44] that these artifacts are so compelling and dramatic in what they suggest about the nature of invention.

The archeological record reveals that stone tool ('lithic') technology developed by the genus *Homo* between about two and a half million years before the present (ybp) to about 35,000 ybp (when 'modern' *Homo sapiens* emerged) had itself evolved in form. The simplest and probably the very first lithic tools were crude choppers made from water-eroded

---

[43] N. Rosenberg and W.G. Vincenti, *The Britannia Bridge : The Generation and Diffusion of Technological Knowledge*, MIT Press, Cambridge, MA, 1978, p. 67; S. Timoshenko, *History of Strength of Materials*, Dover, New York, 1983, p. 160; Dasgupta, *Technology and Creativity, op cit*, p. 64.

[44] N. Toth, "The First Technology", *Scientific American*, April 1987, pp 112-121.

pebbles, from which a few flakes were removed to provide a sharp, jagged edge. The next significant tool was the handaxe. This was in use from the later Lower Paleolithic period (about 1.4 million ybp) when *Homo erectus* is thought to have emerged. Handaxes were characterized by a relatively sharp point and a cutting edge along one or more sides. Most significantly, it was something of a general purpose tool.

Choppers and handaxes are the earliest surviving manifestations of the creative mind.[45] And, indeed, from the perspective of technological creativity, they are of great interest. From *our* point of view, seen as artifacts, they are systemically, almost comically, simple; indeed, from *our* point of view, one can hardly envision simpler objects. And yet, these first inventions obviously necessitated the production of new knowledge where none existed before. There was no remembrance of things past. More precisely, for the inventors of the first stone tools, the only source for new ideas was nature itself -- the pebbles and stones found in nature. The emergence of these artifacts constituted entirely new knowledge. It is, thus, not unreasonable to surmise that the first choppers and handaxes were epistemically complex though systemically simple.

Prehistorians and anthropologists such as Louis Leakey and, more recently, Nicholas Toth have shed some light on this matter. Contrary to what some might think, stone tools are not created by simply "bashing two rocks together".[46] Rather, the first technologists appear to have accumulated quite a sophisticated corpus of operational principles for making stone tools.

Thus, Louis Leakey has described the so-called 'hammerstone' technique for fashioning choppers from water-worn pebbles.[47] From his own experiments, he concluded that in order to remove flakes to create a chopper, early *Homo* had to learn the correct angle at which the blow must be struck so as to detach a flake at the desired point in the desired direction. More precisely, the first *Homo* had to learn that in order to detach a flake from a pebble 'core' in a particular direction, the stone

---

[45] See S. Mithen, *The Prehistory of the Mind*, Thames & Hudson, London, 1996, for an extensive discussion of the 'mind of the first stone tool maker'.

[46] R. Leakey, *The Origin of Humankind*, Basic Books, New York, 1994, p. 38.

[47] L.S.B. Leakey. *Adam's Ancestors : The Evolution of Man and His Culture*, Harper & Row, New York (4th Edition), 1960, pp 32 ff.

would have to be struck at an angle of about 120 degrees to the direction in which the flake was to be removed. Moreover, the point at which the blow was to be struck must be near the edge of the stone. This is an operational principle -- the quintessential form of technological knowledge.

Moreover, not all kinds of stone are amenable to flaking. The stone must be brittle; this is yet another piece of knowledge -- about materials -- which early *Homo* must have come to acquire. It is only in recent times that the twentieth century science of fracture mechanics has provided the basis for a theoretical understanding of stone tool making techniques.[48]

The archeologist Nicholas Toth has also studied the making of stone tools of the kind that were excavated at sites in the Koobi Fora district of northern Kenya. These sites were dated to be between 1.9 and 1.4 million years old, corresponding to the later Lower Paleolithic and early Middle Paleolithic eras. Using the same kinds of rocks known to prevail in Paleolithic times, Toth's experiments yielded further insight into the nature of the hammersmith technique.[49] He discovered that for a stone to flake in a controlled manner, it has to have an acute edge near which the hammer can strike, it must be struck a glancing blow about a centimeter from that edge, and the blow should be directed through an area of "high mass", such as a bulge.[50]

Toth also had other people, having no prior experience, attempt to produce stone tools. Initially, their efforts yielded results that bore little resemblance to the tools produced by early *Homo*. Within a few hours, however, they were able to master the basic technique of removing flakes and were soon able to produce a range of tools.

For Leakey, Toth and Toth's lay experimenters, the process was ontogenetic. In the case of Paleolithic man, we may surmise that the process was probably distributed over time and involved many tool

---

[48] B. Cotterell & J. Kamminga, *Mechanics of Pre-Industrial Technology*, Cambridge University Press, Cambridge, 1990, Chapter 6.

[49] Comparing the pattern of fracture obtained by him with those seen in the archeological remains led Toth to conclude that the tools found in the Koobi Fora sites were made predominantly by means of the hammersmith technique.

[50] Toth, *op cit.*

makers and, thus, was in the nature of a phylogenetic process. But what is more to the point is that for Leakey *et al*, it was not only the tools that were progressively modified in the course of the experiments but the technique of tool making itself. The hammersmith technique is a procedure -- knowledge in the form of an operational principle -- that emerged from these experiments as much as did the tools themselves. Each successive cycle of the ontogenetic act yielded new knowledge -- a more refined operational principle than its predecessor. In other words, despite the systemic simplicity of these artifacts, based on the experiments conducted by Leakey and Toth, we are led to believe that the first stone tools were epistemically nontrivial.

7. *Decreasing systemic complexity: the case of the 'reduced instruction set computer'*

That choppers and handaxes are systemically simple is not surprising; after all, they constituted the first artifacts known to have been made. There was no technological past to draw upon, only nature. But does the evolution of technology inevitably entail the emergence of progressively greater systemic complexity? This is a large question that cannot be adequately answered in a discussion of this length. However, on the Popperian premise that a single counterinstance may suffice to falsify a conjecture we can explore, briefly, one very recent case that seems to suggest that the general answer to this question is in the negative.

I refer here to the development of the 'reduced instruction set compute' (RISC) between 1980 and 1985. (We are now in the realm of 'contemporary history'.) From the functional viewpoint, a computer presents a certain 'facade' to those who will use it. This facade is referred to by a number of different names in the technical literature but, for our purposes, I shall refer to it simply as a computer's *architecture*.[51]

---

[51] S. Dasgupta, "Computer Architecture", in R.A. Meyers (ed.), *Encyclopedia of Physical Science and Technology, 1991 Yearbook*, Academic Press, San Diego, CA, 1991, pp 153-162. Strictly speaking, in this work, I call the facade 'outer architecture' to distinguish this characteristic of a computer from other hidden chracteristics referred to as 'inner architecture'. The qualification is not important in this discussion since our only concern

Briefly stated, a computer's architecture specifies precisely those features of the computer that must be known for a programmer to compose an executable program for that machine, or that will allow a compiler (that is, a program that automatically translates another program from one computer language into an equivalent one in another) to generate an executable program for the computer. A prime example of such an architectural feature is the number of distinct instructions in a computer's instruction set; another is the variety of instruction formats -- that is, the variety of different ways in which instructions can be encoded in a particular computer's memory; a third is the variety of types of data that can be processed by the computer.

In general, the architecture of computers manifests one of the basic characteristics of systemic complexity: its various features are mutually dependent; they interact with one another.[52] More interestingly, by the end of the 1970s, the pattern of evolution of computer architectures showed a distinct tendency towards *increased* systemic complexity. One had only to examine particular genealogical lines of the various classes of computers -- the 'mainframes', the 'minicomputers' and the 'microprocessors' -- made by specific manufacturers to see that this was the case. The sizes of the instruction set, instruction formats and other architectural features had all increased markedly in any manufacturer-defined 'species' of computers.

In the early 1980s, a handful of computer scientists and engineers at the IBM Thomas J. Watson Research Center, the University of California, Berkeley, and Stanford University initiated a movement to reverse this trend. This was not the outcome of a nostalgic yearning for an earlier, simpler world of computers. The movement was based on empirically testable arguments that drew upon a number of factors, notably: the historical causes of the progressive increase in systemic complexity, the change over time in programming techniques, observations and measurements of the poor utilization of architectural features by compilers, and the influence of remarkable developments at the time in integrated circuit fabrication technology -- developments that were captured by the term

---

here is the facade.

[52] S. Dasgupta, *Computer Architecture : A Modern Synthesis, Vol. 1: Foundations*, John Wiley & Sons, New York, 1989, pp 108-109.

'very large scale integration' (VLSI) and that led to the possibility of manufacturing almost an entire computer on a single silicon chip.[53]

Based on such arguments, these designers proposed the idea of the 'reduced instruction set computer' (RISC) -- the idea of designing computers with massively simplified architectures wherein all features are greatly reduced in variety, numbers, and mutual interdependence and interaction.

The RISC idea quickly transformed into a style for commercial computers. It was an invention that marked a reversal of the general trend towards progressively greater systemic complexity of computer architectures. The first experimental RISCs were thus *systemically* simple compared to their conventional counterparts. However, both the invention of the RISC concept (a technological idea) and the translation of the concept by way of design and implementation into actual computers were far from *epistemically* simple. Much historical knowledge was brought to bear by the original inventors in arriving at the RISC concept. And in the transition from concept to reality, significantly new knowledge was generated in the realms of both computer design and compiler technology.[54] The first RISCs were, thus, systemically simple (compared to their predecessors) but such simplicity was gained at the 'cost' of considerable epistemic complexity.

8. *Creativity and epistemic complexity*

We have seen that systemic and epistemic complexities are not necessarily coupled. However, our examples also reveal that epistemic complexity is entirely related to the originality of artifacts and, hence, to the *creativity* of the artificer. An artifact may be systemically complex but, if unoriginal it will be epistemically simple. The civil engineer who designs an

---

[53] D.A. Patterson & D. Dietzel, "The Case for the Reduced Instruction Set Computer", *Computer Architecture News*, 8, 6, 1980, pp 25-33; D.A. Patterson, "Reduced Instruction Set Computers", *Communications of the ACM*, 28, 1, 1985, pp 8-21; J.L. Hennessey, "VLSI Processor Architecture", *IEEE Transactions on Computers*, C-33, 12, 1984, pp 1221-1246. For a summary of these arguments, see Dasgupta, *op cit*, pp 260-267.

[54] M.G.H. Katevenis, *Reduced Instruction Set Computer Architecture for VLSI*, MIT Press, Cambridge, MA, 1985.

elaborate flyover system connecting several busy freeways is undoubtedly creating an artifact of considerable systemic complexity, both structurally and functionally; but if that system is an exercise in normal design, it will not be original; it will be (relatively) simple epistemically speaking.

On the other hand, when in 1936 the engineer-architect Pier Luigi Nervi designed and built aircraft hangars for the Italian Air Force -- functionally, surely the most pleibian of buildings -- he rejected several "traditional solutions".[55] Instead, his design yielded "a single resisting organism" which transmitted the loads to the supports and columns at the sides and thus provided a large, completely uninterrupted volume of space for the aircrafts. Most strikingly, the huge, dome-like vault was composed of a curved, intersecting network of ribs; here was old knowledge invented eight hundred years earlier by the Gothic cathedral builders adapted to a radically different building type. The sublimity of medieval houses of worship was transposed to the most utilitarian of buildings -- with arresting aesthetic effect. Nervi's aircraft hangers were not just aircraft hangars; they were hangars that were visually pleasing. Here, then, was a structure that was epistemically complex because it deployed old knowledge in a wholly surprising context. Epistemic complexity is, then, a measure of the maker's creativity.

9. Measures and descriptors of systemic complexity

But can complexity be *measured* at all? In the realm of systemic complexity, there is certainly no single set of universally accepted measures; rather, each different domain, wherein such complexity is relevant, has adopted its own specific metrics. Thus, John Tyler Bonner, in his detailed discussion of the evolution of (systemic) complexity of organisms and the biological universe, drew upon such factors as the body size of organisms, the diversity of cell types within an organism, and the diversity of the types of organisms within a community.[56] But Bonner also admitted

[55] P. Nervi, *Aesthetics and Technology in Building*, Harvard University Press, Cambridge, MA, 1966.

[56] J.T. Bonner, *The Evolution of Complexity by Natural Selection*, Princeton University Press, Princeton, NJ, 1988.

function and behavior into the realm of complexity, in the sense that the more elaborate the functional/behavioral repertoire of an organism the more complex the organism; in the latter case, however, one cannot adequately quantify these factors.[57]

In the realm of artifacts, similar quantitative and qualitative criteria sometimes prevail. Thus, the common measure of systemic complexity of integrated circuit chips is the number of transistors on the chip. The systemic complexity of a software system has been described in terms of such measures as the size of the system (given by the number of lines of instructions), the number of modules comprising the system, the average size of modules, and the amount of connectivity of, or dependence amongst, modules. Indeed, it was precisely these measures that M.M. Lehman and L.A. Belady drew upon in their pioneering studies of the nature of software evolution in the 1970s.[58]

As in Bonner's treatment of biological behavior, the repertoire of functions designed into an artifact is frequently employed to describe systemic complexity: an artifact with a larger, more elaborate repertoire of functions or behaviors is generally regarded as being more complex than another more specialized artifact. But here again, functional or behavioral diversity is meaningful only in a qualitative sense.

Yet another kind of measure in the realm of artifacts is that used to characterize the complexity of algorithms.[59] One of the key properties of algorithms, especially computer algorithms, is the (average or maximum) number of operations of a certain type it performs in order to solve the class of problems it has been designed for. This number is usually stated as a mathematical function of some characteristic of the problem itself. Thus, for example, an algorithm for sorting a set of N unordered numbers into a strictly ascending sequence may be said to be of "order

---

[57] See D.W. McShea, "Complexity in Evolution : A Skeptical Assessment", this issue, for more on the measurement of biological complexity.

[58] M.M. Lehman, "Programs, Cities and Students : Limits to Growth?", Inaugural Lecture, Imperial College of Science and Technology, reprinted in D. Gries (ed.), Programming Methodology, Springer-Verlag, Berlin, 1974, pp 42-69; L.A. Belady and M.M. Lehman, "Characteristics of Large Systems", in P. Wegner (ed.), Research Directions in Software Technology, MIT Press, Cambridge, MA, 1979, pp 106-138.

[59] Algorithms belong to what I have elsewhere called the class of 'abstract artifacts'. See Dasgupta, Technology and Creativity, op cit, pp 11-12, 74-78.

N ". That is, the number of computer operations required to perform the sort is proportional to the square of the number of numbers to be sorted.[60]

Consider quite a different class of artifacts: alloys. An alloy is basically a solid solution in which the more abundant component (the solvent) is a metal while the less abundant components (the solutes) may be metals or nonmetals. The systemic complexity of an alloy arises from the fact that its properties are determined not only by its chemical composition but also by its particular constituent crystal structures ('phases') and by the thermal treatment ('heat treatment') the alloy is subjected to. Metallurgists have indeed used the number of alloying elements as a measure of systemic complexity[61] but it is seldom that this number alone serves the purpose. Two other features are almost inevitably employed. One is 'microstructure' -- that is, those structural features of an alloy that can be observed under the microscope. The other is the 'phase diagram' or 'equillibrium diagram' -- that is, a diagram that shows the various phases (or crystal structure states) of an alloy system as a function of its composition and temperature.[62] Both microstructure and phase diagram reveal systemic complexity in a manner that alloy composition does not. Thus, for example, the so-called 'plain carbon steels' are composed, basically, of only two elements, iron and carbon and, hence, using the number of alloying elements alone as a measure of complexity (in this case, there is only carbon), such steels might seem systemically rather simple. When we examine the iron-carbon equillibrium diagram, however, the picture is quite different: plain carbon steel is now seen not so much as the alloyed mixture of two elements but rather as a system of phases wherein the alloy is transformed from one phase into another at different temperatures and varying percentages of carbon. Microstructural

[60] A.V. Aho, J.E. Hopcroft & J.D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974, pp 2-5, 12-14.

[61] F.L. Ver Snyder, "Status and Future of Nickel-Based Alloys", in R.F. Hehemann & G.M. Ault (eds), *High Temperature Materials*, John Wiley & Sons, New York, 1959, pp 3-15.

[62] See, for example, R.W. Cahn (ed.), *Physical Metallurgy*, North-Holland, Amsterdam, 1970, esp. H.J. Rack & R.W. Newman, "Microstructures", chapter 12; and G.V. Raynor, "Phase Diagrams and their Determinations", chapter 7.

studies yield yet further insight into systemic complexity by revealing visually the morphology of the alloy and the variation in the geometry of its constituent phases under different heat treatment conditions.[63] Attributes of the microstructure *can* be measured, for example, the sizes of the individual grains of the alloy[64] but, again, no single measure characterizes the microstructure as a whole.

Thus, singular measures of systemic complexity appear to be less than adequate even in restricted technological (or for that matter natural) domains. Systemic complexity is a *holistic* property that is determined by several dimensions characterizing the relevant domain. It is more meaningful, then, to search for *descriptors* of complexity rather than some unequivocal metric and, indeed, that is exactly what engineering and natural scientists implicitly do. Thus, a statement that a given sorting algorithm is of "order of N "is a description of its level of complexity; it is more complex than another sorting algorithm of (say) "order of N x log N". The iron-carbon diagram is *one* description of the systemic complexity of carbon steels; the compendium of distinct microstructures is yet another. The geometric procedures devised by the medieval builders are descriptors of the complexity of some aspects of Gothic cathedrals (such as the plan of the Laon Cathedral tower shown by Villard de Honnecourt[65] or the rib vaults and the arches of windows.[66]).

Such procedures are instances of what Herbert Simon called 'process descriptions'.[67] They serve to provide descriptions that are intended to be simpler than the entities described. In the realm of population biology, first-order nonlinear difference equations of the general form $X(t + 1) = F(X(t))$ describe the magnitude of a population of organisms in generation $t + 1$ as a function F of the population magnitude $X(t)$ in the previous generation. Specific instances of this rather innocent looking equation, in fact, are descriptors of the rather complex behaviors of popu-

---

[63] R.W.K. Honeycombe, *Steel : Microstructure and Properties*, Edward Arnold, London, 1981, pp. 28 *ff.*

[64] Rack & Newman, *op cit*, pp 746 *ff.*

[65] Bowie, *op cit.*

[66] Addis, *op cit*, pp 138-139.

[67] Simon, *The Sciences of the Artificial, op cit*, p. 210.

lations of organisms.[68]

In the realm of linguistics, a grammar, in the form of a set of 're-writing rules', is a process descriptor of all the syntactically correct sentences in a given language. A grammar is also a descriptor of the syntactic complexity of a language. Based on this notion, the architectural design theorists William Mitchell and George Stiny have described grammars to define and produce a variety of architectural forms such as the forms of classical Greek columns or the floor plans of villas in the Palladian style.[69] In the Palladian case, a set of some seventy rewriting rules define the 'Palladian floor plan grammar'; Mitchell has shown the rich variety of the floor plans that can be generated by this grammar, including several of the actual plans shown in Palladio's treatise, *Four Books of Architecture*.[70] This grammar can, thus, serve as a descriptor of the systemic complexity of Palladian floor plans.

Unfortunately, the systemic complexity of certain artifacts càn only be captured adequately by a description of the *entire* artifact. For instance, although such measures as the number of instructions and the variety of instruction formats are used to signify the complexity of a computer's architecture, the interactions amongst such components cannot be captured by these measures alone. Nor do they reflect, necessarily, the functional diversity of the computer. Ultimately, it is only the entire specification of the computer architecture that can serve as a descriptor.

10. *Descriptors of epistemic complexity*

Epistemic complexity appears to be even less amenable to quantification than its systemic counterpart. One can, of course, claim to measure it by simply counting the number of significant knowledge 'tokens' (i.e., distinct items of knowledge such as facts, concepts, hypotheses, etc.) that

---

[68] R.M. May, "Simple Mathematical Models with Very Complicated Dynamics", *Nature*, 261, 5560, 1976, pp 459-467.

[69] G. Stiny & W.J. Mitchell, "The Palladian Grammar", *Environment and Planning*, B5, 1, pp 5-18, 1978; W.J. Mitchell, *The Logic of Architecture*, MIT Press, Cambridge, MA, 1990, chapter 8.

[70] Mitchell, *op cit*, pp 152-179.

took part in the invention of the artifact. Such a count would serve as the crudest of measures. For example, in my discussion of the invention of the first 'superalloy' for gas turbine blades, I was able to identify some twenty three significant tokens of knowledge that appeared to have participated in the invention process.[71] A majority of these constituted 'old' knowledge that the metallurgists drew upon; the remaining were generated in the course of invention. But such a count conveys nothing of the intricacy of the interactions of these knowledge tokens, nor the manner in which they came to participate in the cognitive act, nor (in the case of old knowledge) why they were invoked at all. As noted earlier, epistemic complexity of an artifact reflects directly the creativity of the cognitive act that resulted in the artifact, and the mere 'size' of the relevant knowledge base conveys only one dimension of this creativity. The only adequate descriptor of epistemic complexity is *a description of the ontogenetic process itself* or some adequate *representation* of this process.

An example of such a representation was employed by me to describe the epistemic complexity of the Britannia Bridge, designed by Robert Stephenson and William Fairbairn in the mid-nineteenth century.[72] Stated briefly, the main part of this description consists of a chain of four main hypotheses which Stephenson and his associate formulated in the course of design. Each hypothesis was associated with a specific bridge form,and each successive hypothesis was derived from its predecessor as the result of critical thinking, analysis or laboratory tests. The main chain of hypotheses is augmented by several auxiliary hypotheses which were also generated in the course of design. The resulting network of knowledge is still an abstraction which leaves out a great deal of detail. It does, nevertheless, capture rather well the holistic nature of the epistemic complexity of the Britannia Bridge. It is possible however, to fill in some of the details as, for example, identify the network of knowledge tokens generated in Stephenson's conceptualization of the basic tubular form of the bridge -- that is, in the formation of the bridge form associated with the first main hypothesis.[73] The outcome is a description of the epistemic complexity of the *concept* of the tubular bridge;

---

[71] Dasgupta, *op cit*, pp 69-74, 152-156.

[72] Dasgupta, "Testing the Hypothesis Law of Design . . . ", *op cit*.

[73] Dasgupta, *Technology and Creativity, op cit*, pp 110-116.

this description consists of an interacting web of previously established goals, facts about various bridge forms, general heuristic rules pertaining to engineering design, general problem solving strategies, as well as new goals, facts and hypotheses produced in the course of the design process.

11. *Conclusions: towards a cognitive history of technology*

In this paper, I have argued that artifacts are characterized by two kinds of complexity. Of these, the one I have called 'systemic' is the complexity of forms of artifacts and of the behaviors such forms give rise to. Systemic complexity is not unique to technological products; natural systems manifest it also. The other type of complexity, which I have termed 'epistemic', is however uniquely characteristic of made things (including abstract things), for it pertains to the combined richness of the knowledge the maker brings to bear and is generated in the course of creating artifacts. Here too, epistemic complexity is not unique to technological products: 'non-useful' things manifest it also. Paintings, sculptures, novels, poems and plays, symphonies, fugues and ragas are all infused with epistemic complexity, especially in the intricate ways their creators summon the past and integrate it into their works.

Understanding systemic complexity tells us *what* the nature of an artifact is. Understanding epistemic complexity tells us *how* that artifact assumed the form it did. Systemic complexity, in the case of artifacts, is ahistorical. Epistemic complexity is profoundly historical; it is a record of how the present is informed by the past.

Most significantly, the epistemic complexity of an artifact, useful or otherwise, provides a trace of its maker's creativity. In this sense, it is a far richer characteristic of artifacts than its systemic counterpart, for it contributes to a depth of understanding of the artifact which the analysis of systemic complexity cannot. To take an example from art -- a domain of 'nonuseful' artifacts! -- a viewer's appreciation of Picasso's *Guernica* is vastly enriched when she makes some contact with the web of past ideas that Picasso brought to bear in painting the picture.[74] So also in

[74] R. Arnheim, *Picasso's Guernica : The Genesis of a Painting*, University of California Press, Berkeley, CA, 1962.

the case of technology. To understand the epistemic complexity of the
Britannia Bridge, or the first high-performance superalloy, or the Multics
operating system, or the first stone tools (to cite only those examples
discussed in this paper) is to understand both the evolutionary pathway
which informed such an invention and, to some extent, the cognitive
process entailed in the act of invention. Epistemic complexity is, thus, a
gateway into both the phylogeny and ontogeny of an invention.

It follows that the so-called 'sciences of complexity', such as general
systems theory, cybernetics and chaos theory are not the appropriate
disciplines for the investigation of epistemic complexity. Such sciences
are ahistorical; they are, in fact, sciences of *systemic* complexity. In
contrast, epistemic complexity lies at the intersection of the history of
technology and the psychology of invention. Its investigation demands a
marriage of the disciplines of history of technology and cognitive science.

Such cooperation between distinct disciplinary cultures is by no
means unknown in the realms of literature and art, wherein historians and
biographers have freely appealed to cognitive and perceptual psychology
as a source of insight.[75] Even the historian of science has ventured into
such cognitive studies.[76] Conversely, some cognitive scientists and psychi-
atrists have entered territories conventionally monopolized by the his-
torian and the biographer.[77]

While historians of technology have expressed interest for some time

[75] See, e.g., E.H. Gombrich, *Art and Illusion : The Psychology of Pictorial Represen-
tation*, Princeton University Press, Princeton, NJ, 1969; L. Edel, *Stuff of Dreams :
Experiments in Literary Psychology,* Avon Books, New York, 1982.

[76] A.I. Miller, *Imagery in Scientific Thought*, MIT Press, Cambridge, MA, 1987.

[77] V. John-Steiner, *Notebooks of the Mind*, University of New Mexico Press, Albuquer-
que, NM, 1985; V. John-Steiner, "From Life to Diary to Art in the Work of Anais
Nin", in D.B. Wallace & H.E. Gruber (eds.), *Creative People at Work*, Oxford Univer-
sity Press, New York, 1981, pp 209-226; L.R. Jeffrey, "Writing and Rewriting Poetry
: William Wordsworth", in Wallace & Gruber, *op cit*, pp 69-90; A. Rothenberg, *Crea-
tivity and Madness,* The Johns Hopkins University Press, Baltimore, MD, 1990; A. Storr,
The Dynamics of Creation, Martin Secker & Warburg, London, 1972; K.R. Jamieson,
Touched With Fire : Manic-Depressive Illness and the Artistic Temperament, The Free
Press, New York, 1993.

in the nature of technological creativity[78] it is only very recently that cognitive ideas are being applied to its understanding and to the concomitant understanding of epistemic complexity.[79] It is my belief that such questions as whether the history of technology is also a history of the evolution of technological complexity cannot be addressed adequately without knowledge and understanding of the place of epistemic complexity in the evolution of artifacts. To answer, for example, the question of whether the development of the electronic computer from its origins in the late 1930s to its most recent forms has resulted in the evolution of progressively greater complexity of the computer-as-artifact certainly demands analysis of its systemic complexity; but more significantly, it demands the investigation and understanding of the evolution of its epistemic complexity over time. Such an understanding necessitates the emergence of what I shall call a *cognitive history* of technology. In the absence of such a history, our understanding of technology as knowledge (or as self-knowledge, in the words of Lynn White, Jr[80]) will remain severely incomplete.

University of Southwestern Louisiana

---

[78] J.M. Staudenmaier, *Technology's Storytellers*, MIT Press, Cambridge, MA, Chapter 2, 1985.

[79] R.J. Weber & D.N. Perkins (eds), *Inventive Minds*, Oxford University Press, New York, 1992; S. Dasgupta, *Creativity in Invention and Design*, Cambridge University Press, New York, 1994; Vincenti, *What Engineers Know . . .*, *op cit*; Dasgupta, *Technology and Creativity, op cit*.

[80] Staudenmaier, *op cit*, p. 35.