IEEE Transactions on Systems, Man & Cybernetics, 18(4), 532-541, 1988.

A Conceptual Framework for Person-Computer Interaction in Distributed Systems

Brian R. Gaines
Knowledge Science Institute
University of Calgary
Alberta, Canada T2N 1N4

Abstract
This paper presents a conceptual framework for complex systems of computers and people. Distinctions between technology and people, and between computers and non-programmed technology, are analyzed. This analysis is used to show how various forms of analogy and abstraction may be used to derive design principles for person-computer interaction. The analysis is extended to include relations between system structure and behavior, and used to develop a hierarchical model of the protocols in person-computer systems.

1 Introduction
The increasing capabilities of computing systems, and the increasing demands for powerful command and control systems, are leading to the development of very complex computer-based systems involving large-scale software development, a wide variety of human-computer interfaces, and the integration of complex behavior in people and computers. It is difficult for human factors models and guidelines to keep pace with the rapid evolution of such systems. Design recommendations in the literature are largely empirical, based on practical experience rather than deep models, and, although they have grown from some ten rules in the 1970s (Hansen 1971) to over one thousand in the 1980s (Smith & Mosier 1986), they apply largely to past system designs, and generalizations to new systems depend on the intuitions of the system designer.

Empirical guidelines may be seen as a 'bottom-up' approach to system design, taking specific experience and generalizing it to other cases. An alternative, 'top-down' approach is to analyze the significant distinctions being made by a system designer and develop a conceptual framework for the design of complex systems. The two approaches are complementary since the conceptual framework can be used to guide the generalization of empirical results, and the empirical results can be used to guide the selection of significant distinctions.

This paper develops a foundational conceptual framework for person-computer interaction in complex systems by analyzing them in terms of some very basic distinctions from the systems theory literature, and showing that these correspond to the distinctions made in established design guidelines and previously proposed conceptual frameworks. It shows that these distinctions are not only descriptive of significant system features, but also provide a prescriptive basis for design decisions relating to these features.

2 Systemic Foundations for Complex Person-Computer Systems
In analyzing person-computer interaction it is important to establish what are the critical characteristics of a person and what are those of a computer. How are they differentiated from one another and from other forms of system? Habermas (1981) has emphasized the essential differences between the dynamics of society and that of physical systems. The "laws" that

govern human behavior are largely conventions embedded in our society and propagated through our culture (Schutz & Luckman 1973). They do not have the necessity of the laws underlying physical systems and their analysis differs in many ways from those of physical systems (Ulrich 1983). Human beings create their future through acts of choice that are constrained by their forward-looking intentions. Physical systems unroll their future through acts of necessity that are determined by their previous states. From this point of view, in person-computer interaction we are dealing with phenomena of the life-world which are essentially different from those of the physical world and cannot be encompassed by causal models.

This perspective is radically different from a behaviorist psychological model of people which sees them as conditioned through reinforcement into patterns of behavior that mimic the causal dependencies of physical systems (Skinner 1974, Mackenzie 1977). Habermas' point of view is more akin to a constructivist model of people which sees them as anticipating the future through the development of personal construct systems that are, however, always reconstructable (Kelly 1955). It is reasonable to suppose that people exhibit both these phenomena, and that any model of personal-computer interaction should be able to encompass those aspects of human behavior that are best modeled as causally based and those which are best modeled as anticipatorily based.

The computer component of the system seems straightforward because it has a physical, causal basis. However, computers are programmed by people and hence their behavior also involves human choice. Once they are operating the causal model is appropriate, but in considering system design and person-computer relationships it is important to take the human component into account. The users' mental models may be influenced by their impressions of the designers' intentions. The system designer may be seen as behaving through the computer to anticipate the future behavior of users.

To encompass the possibilities discussed, a conceptual framework is required that can account for these essential differences and similarities, differentiating people from technology and computing from non-programmed physical systems. Popper's (1968) *3 worlds theory* can be used to provide foundations for the distinctions underlying physical processes, human activities, and information technologies. He bases his theory on Bolzano's notion of "truths in themselves" in contradistinction to "those thought processes by which a man may grasp truths", proposing that "thoughts in the sense of contents or statements in themselves and thoughts in the sense of thought processes belong to two entirely different worlds" (Popper in Schilpp 1974), and making the three-fold distinction: "If we call the world of things—of physical objects—the first world and the world of subjective experience the second world we may call the world of statements in themselves the third world (world 3)." These concepts capture the differences between physical systems and people, and the role of abstraction in moving the distinctions into a third realm regardless of source.

Figure 1 shows the existential hypotheses underlying Popper's 3 worlds as a conceptual framework based on very general distinctions made about distinctions. It introduces a 'World 0' of distinctions in their own right before Popper's criteria are applied. This world is one where distinctions are made without reference to any criteria except that they are useful. It captures the notions of value and of the underlying anarchy that Feyerabend (1975) sees in the scientific process. Gaines (1980) has argued that any distinction generates a system and hence that World 0 as defined here is that of general systems.

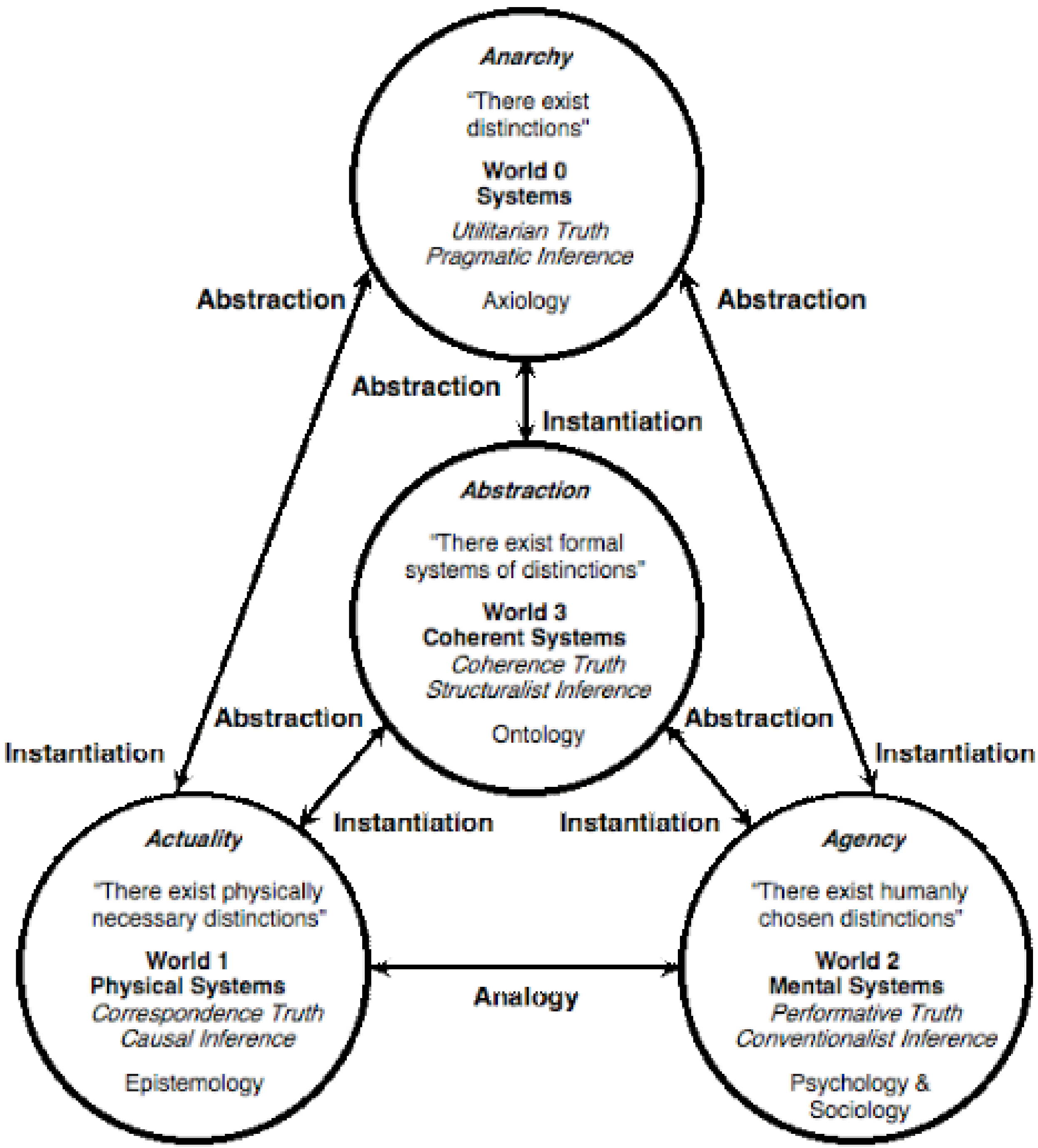

Figure 1 Anarchy, abstraction, actuality, and agency as basic distinctions among distinctions

- World 0 arises from the assertion that there exist distinctions
  - this defines a world of *systems*
  - a key concept is that of *anarchy*, that distinctions may be made freely
  - the conceptual framework involved is that of *axiology*, that distinctions have value
  - truth in this world is *utilitarian*, that distinctions made are useful
  - inference is *pragmatic*, based on chains of reasoning that appear to work

- World 1 arises from the assertion that there exist *physically necessary* distinctions
  - this defines a world of *physical systems*
  - a key concept is that of *actuality*, that the distinctions made are actual
  - the conceptual framework involved is that of *epistemology*, that the distinctions have to be drawn from the actual world
  - truth in this world is by *correspondence*, that the distinctions made correspond to the properties of physical systems
  - inference is *causal*, based on chains of physical cause and effect
- World 2 arises from the assertion that there exist *humanly chosen* distinctions
  - this defines a world of *mental systems*
  - a key concept is that of *agency*, that distinctions have a maker
  - the conceptual frameworks involved are those of *psychology and sociology*, that distinctions exist in the mental worlds of individuals and groups
  - truth in this world is *performative,* that the distinctions chosen will be continue to be chosen
  - inference is *conventionalist*, based on accepted chains of reasoning that are chosen within a culture
- World 3 arises from the assertion that there exist *formal systems* of distinctions
  - this defines a world of *coherent systems*
  - a key concept is that of *abstraction*, that systems of distinctions have their own existence
  - the conceptual framework involved is that of *ontology*, that some distinctions are made independent of any criteria other than intrinsic structure
  - truth in this world is by *coherence*, that the systems of distinctions have an acceptable intrinsic structure
  - inference is *structuralist*, based on chains of reasoning that conform to the structure

Worlds 1, 2 and 3 may be seen as components of World 0 and also as particular instances of it. Similarly Worlds 2 and 3 may be seen as instances of World 1. Conversely World 0 may be seen as an abstraction from Worlds 1, 2 and 3, and World 3 may be seen as an abstraction from Worlds 2 and 3. The conceptual framework of Figure 1 shows how the worlds relate through this *abstraction* and *instantiation*. It also shows the *analogy* between worlds 1 and 2 when we attribute agency to the causal dynamics of physical objects or necessity to the social conventions of human activity.

Figure 2 extends the conceptual framework to show equipment as an instance of a physical system, people as an instance of mental systems, and computers as an instance of both physical and mental systems, inheriting properties through both paths. The peculiar property of electronic digital computers is that they are the archetype of deterministic causal systems, modeled in their behavior as finite-state automata whose next state and output are precisely determined by their current states and inputs. However, they are also programmed devices where the transition tables of the automaton are completely under the control of people—what computers do is what we chose them to do. Hence, they are also the archetype of performative, conventional systems, modeled in their behavior as the intentional artefacts of people, and whose next state and output are in major part determined by open choices made in programming.

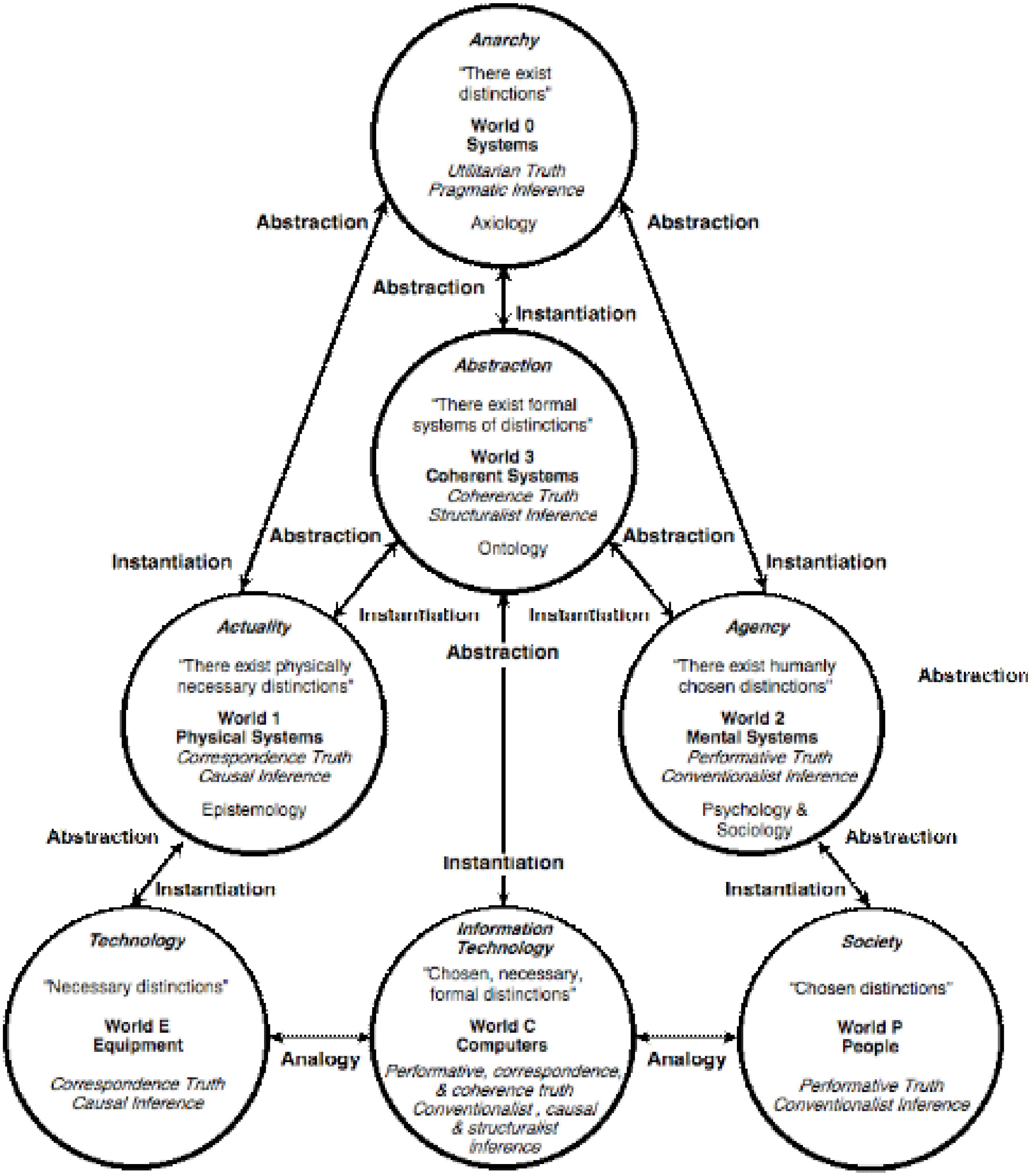

Figure 2 Conceptual structures for person-computer interaction in terms of basic distinctions

This dual nature of computing systems underlies many of the problems and controversies in person-computer interaction. The computer is a technological system and it may be advisable to present it to its users as such, avoiding animistic considerations. However, the behavior of the computer is prescribed by people and this underlying prescription may also be evident to the user. The value systems and inter-personal attitudes of the system designer and programmer may have become embedded in the system behavior, making an animistic perspective unavoidable. The performative truth and conventionalist inference aspect of computers is implicit in rules for programming effective person-computer dialog. For example (Gaines & Shaw 1984), "The system should be consistent in operation—the commands should always do the same thing throughout—the information presentation should always mean the same thing throughout" and "The system should be uniform in operation—the facilities which users have learned to use in one part of the package should be available to them in other parts if they might reasonably expect this", are both example of performative truths allowing inference from reasonable conventions.

The analogies between computers, people and equipment shown in Figure 2 are particularly interesting. If the conventions of the life-world become widely accepted in a culture then they assume the same status as the necessary distinctions of the physical world and the behavior of people within a culture may appear as completely causal. In informal terms, one may be just as hurt attempting to break a moral convention as attempting to walk through a brick wall. In person-computer interaction skilled operators may have highly practiced patterns of behavior that may be modeled through causal dynamics. They are no longer making choices at the lower levels of functioning. This is the basis for some negative "transfer of training" phenomena when functioning developed for one system is carried over to another where it may result in inappropriate choices. It should also be noted that all technology has some aspects of human choice embodied in it so that the distinction between technology in general and information technology shown in Figure 2 is not a hard boundary.

The computer is also shown in Figure 2 as an instance of a World 3 system, that is an abstract entity defined in terms of its coherent structure. Category-theoretic models of software systems exemplify the possibility for a high degree of abstraction while still capturing essential features of computing systems (Goguen and Meseguer 1983) and analogies between them, and between them and people (Gaines 1975). The abstract World 3 entity will generally lose some features of both the World 1 physical dynamics that underlies its implementation and the World 2 psychological dynamics that underlies the choice of what to implement.

3 Analogy and Abstraction in Analyzing Person-Computer Systems

The bases for the analogies between computers and people and between computers and other technical equipment show up clearly in Figure 2. The bases of the abstractions that treat both computers and people as general systems are also apparent. Figure 3 illustrates how these analogies and abstractions give rise to a variety of conceptual frameworks for the study of person-computer interaction which have been previously noted as major sources of information about, and insight into, the analysis and design of person-computer systems (Gaines & Shaw 1986).

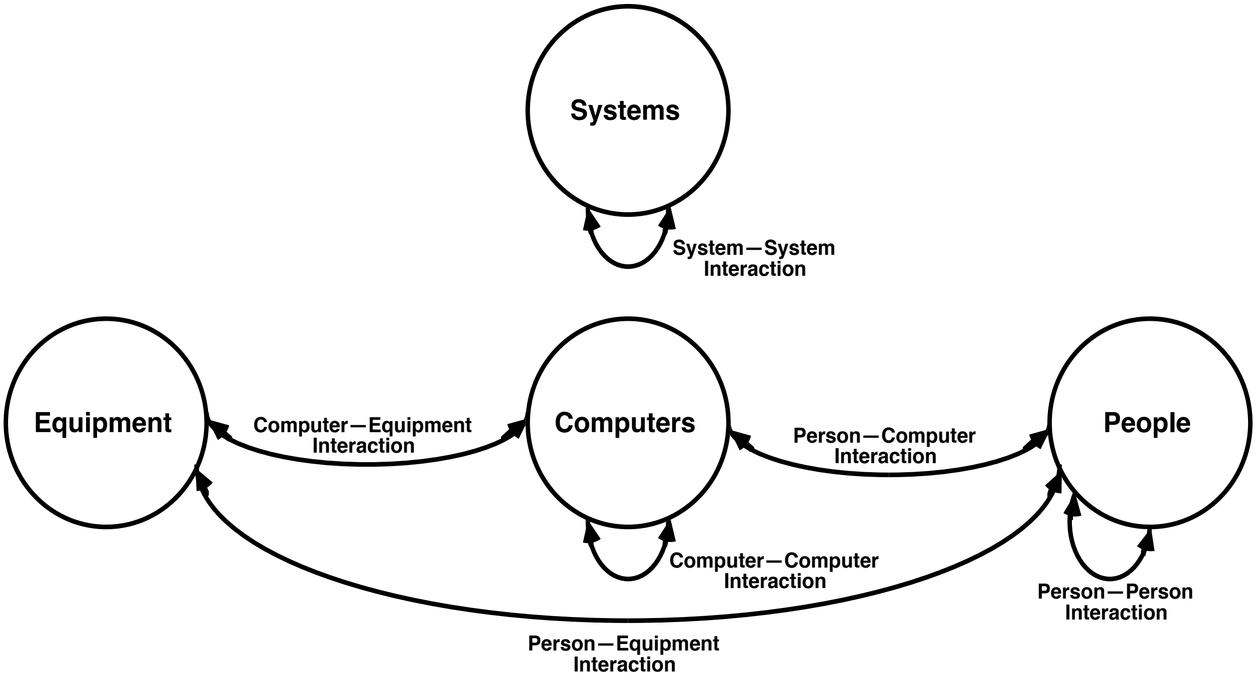

Figure 3 Abstraction and analogies in the analysis of person-computer interaction

3.1 System-System Interaction

At the abstract level of system-system interaction there are many system-theoretic principles that are instantiated in the situation where one system is a person and the other a computer. Wiener (1948) emphasized this in his development of cybernetics as the study of "communication and control in men and machines", and many principles of communication theory and control theory apply directly to person-computer interaction. For example, the dialog rule that "all actions are initiated and controlled by the user" (Cheriton 1976) derives from a stability-theoretic result in control theory, that two coupled systems with similar time constants may oscillate unstably around their intended equilibrium state: the person modeling a computer and adapting to it while the computer is modeling the person and adapting to him is a potential source of mutual instability. General systemic principles at the abstraction level interact with more specific considerations at the instantiation level. For example, the dialog rule "avoid acausality" (Gaines 1981) has a system-theoretic foundation in that causal modeling systems generate meaningless models of systems with even slight acausalities. However, to apply this principle we have to know that people are causal modelers (Gaines 1976), and we would not regard the rule as significant unless we knew that time-sharing systems generate apparently random delay distributions.

3.2 Computer-Equipment Interaction

The analogy from people to physical systems enables person-computer interaction to be seen to be analogous to interfacing a computer to another system such as a piece of equipment. The design principles applicable to computer-equipment interfaces are well known and carry over to person-computer dialog. Problems arise because the system to which the computer is to be interfaced already exists and is not another programmable computer. We may take it as it is and design an interface that copes with its peculiarities. The dialog rule "use the user's model"

(Gaines 1981) derives from this, that the dialog engineer should identify the existing interface and attempt to emulate it rather than change it. Problems also commonly arise through noise at the interface and the designer attempts both to provide a low-noise channel and to provide error-detection and correction for unavoidable noise. In person-computer interaction such noise may arise through lack of clarity in information presentation giving rise to perceptual errors in one direction, mis-keying giving rise to errors in the other direction, and so on. The dialog rules "detect user difficulties and assist him in returning to correct dialog" (Cheriton 1976) and "validate data on entry" (Gaines 1981) are principles of communication over a noisy channel.

3.3 Computer-Computer Interaction

The analogy from people to computers enables person-computer interaction to be seen to be analogous to computer-computer interaction. It is then possible to define protocols that any programmable system may be reasonably expected to be able to implement. The Open System Interconnection (OSI) ISO standard (Day & Zimmerman 1983) is particularly interesting because it hierarchically structures computer-computer protocols for networks in a way that may have relevance for person-computer protocols. The concept of an open system is itself relevant because it expresses objectives for computer networks that are equally applicable to people using those networks. The aim is to allow integrated systems to be formed from multiple components not all from one vendor and not all installed at the same time. The OSI concept is that the network is open to all systems that conform in their communications with certain well-defined protocols. In human terms the protocols may be seen as social norms for the behavior of members of a club; anyone may join provided they agree to conform to these norms. Taylor has applied such protocol concepts to the analysis of person-computer dialog as a hierarchical communications system (Taylor 1987).

3.4 Person-Equipment Interaction

The analogy from computers to physical systems enables person-computer interaction to be seen to be analogous to the classic case of man-machine interaction. Consideration of people interacting with equipment has been treated as a branch of applied psychology termed ergonomics that arose, under the same pressures as computer technology, out of World War II studies of pilots, gunners and so on. There is a wide range of results on general problems of human skills, training, its transfer between different learning situations, the effects of fatigue, and so on, that is applicable to person-computer interaction. While interactive computers were used primarily as programming and data entry systems these effects were not major considerations. However, as computer-based interfaces became increasingly the norm for a wide variety of human activities the classic results of applied psychology and ergonomics have become increasingly important. The novelty of the computer should not blind us to commonality with much earlier equipment.

3.5 Person-Person Interaction

The analogy from computers to people, enables person-computer interaction to be seen to be analogous to person-person interaction, that is normal linguistic interaction from which the terms human-computer "conversation" and "dialog" in computing terminology have been generalized. Modern linguistic theory (Bennett 1976) has become increasingly concerned with the interaction between participants in a dialog, rather than a view of linguistic output as a predefined stream to be decoded. This provides a rich source of models for person-computer interaction, particularly as artificial intelligence techniques take us closer to emulating people and their language behavior. There are also useful analogies of casual users in transactional analysis of the behavior of strangers meeting (Berne 1974). Taylor notes that many of the principles of language such as scoping and pronomial reference are general principles of communication that occur in gestural graphic dialog with computers that involves no textual interaction (Taylor 1987).

4 System Origins and Activity

The principles underlying Figure 3 provide systemic foundations for the forms of reasoning by analogy and abstraction that are commonly used in the analysis and design of person-computer systems. To develop these concepts more formally and provide a framework for the overall dynamics of person-computer interaction, we need to introduce a further distinction, that between the activity and the origins of a system, between its behavior and its structure. In abstract terms the behavior of a system provides a description of what the system does, and the structure of a system provides a description of what the system is.

One of the most important problems of system theory is the analysis of the relations between behavior and structure—in one direction, given the structure of a system, to derive its behavior—in the other direction, given the behavior of a system, to derive its structure. In the study of physical systems, mathematical techniques have been developed for moving in both directions with causal models (Gaines 1977, Klir 1985), that is interrelating the necessities of World 1 behavior and structure. However, as already noted, integrated computing systems involve the choice behavior of people and hence show phenomena of the life-world which are essentially different from those of the physical world and cannot be encompassed by causal models.

The dynamics of human behavior are best modelled as those of an *anticipatory system* (Rosen 1985), enhancing its survival by modeling the world, both passively and actively, in order to better anticipate the future. This corresponds to the choice component of World 2 phenomena, that agents are not bound by rigid necessity but can plan and chose certain aspects of their behavior. Figure 4 shows an analysis of the relations between a system, its origins and its activities, when the *actuality—agency* distinction of Figure 1 is also taken into account. The origins of a system have two components: its *causal structure* relating to how it was created; and its *anticipatory structure* relating to why it was created. The activities of a system also have two components: its *causal behavior* relating to how it carries out its activity; and its *anticipatory behavior* relating to why it carries out its activity.

The basic distinctions of *actuality—agency* and *structure—behavior* are sufficient to define the virtual machine hierarchy fundamental to computing, and to account for its technical and human components and their relations. Figure 5 fills in Figure 4 to show how the levels of virtual machine arise from the basic distinctions. The origins of any computer-based service are that a *virtual machine*, a computing system, has been programmed by agents, the *implementors*, to create an actuality, the *implementation*. This results in a second virtual machine, the programmed computing system. The activities of the computer-based service are that a *virtual machine*, a programmed computing system, is being operated by agents, the *operators*, to create an actuality, the *operations*. This results in a third virtual machine, the service itself.

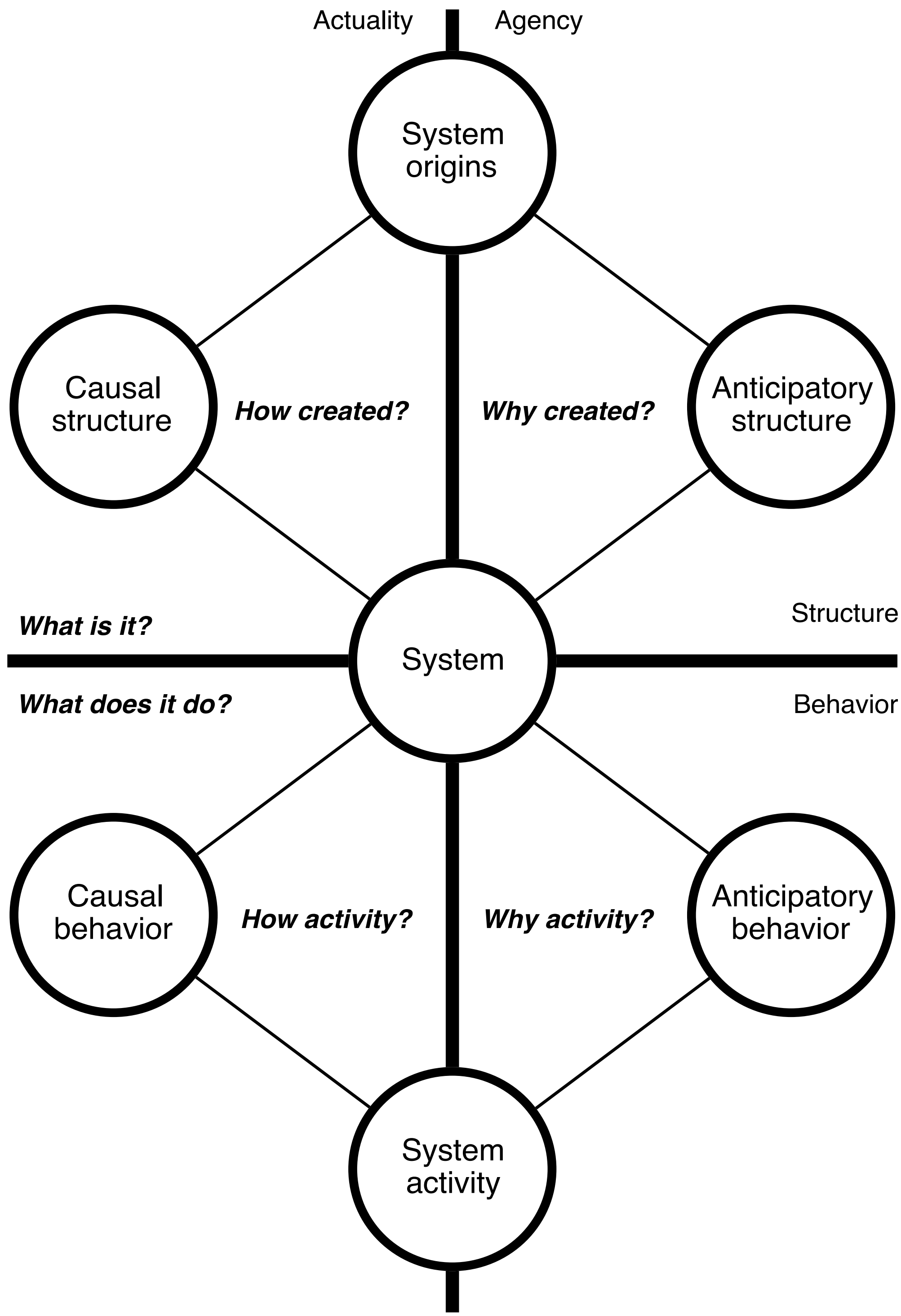

Figure 4 System origins and activity in terms of actuality and agency

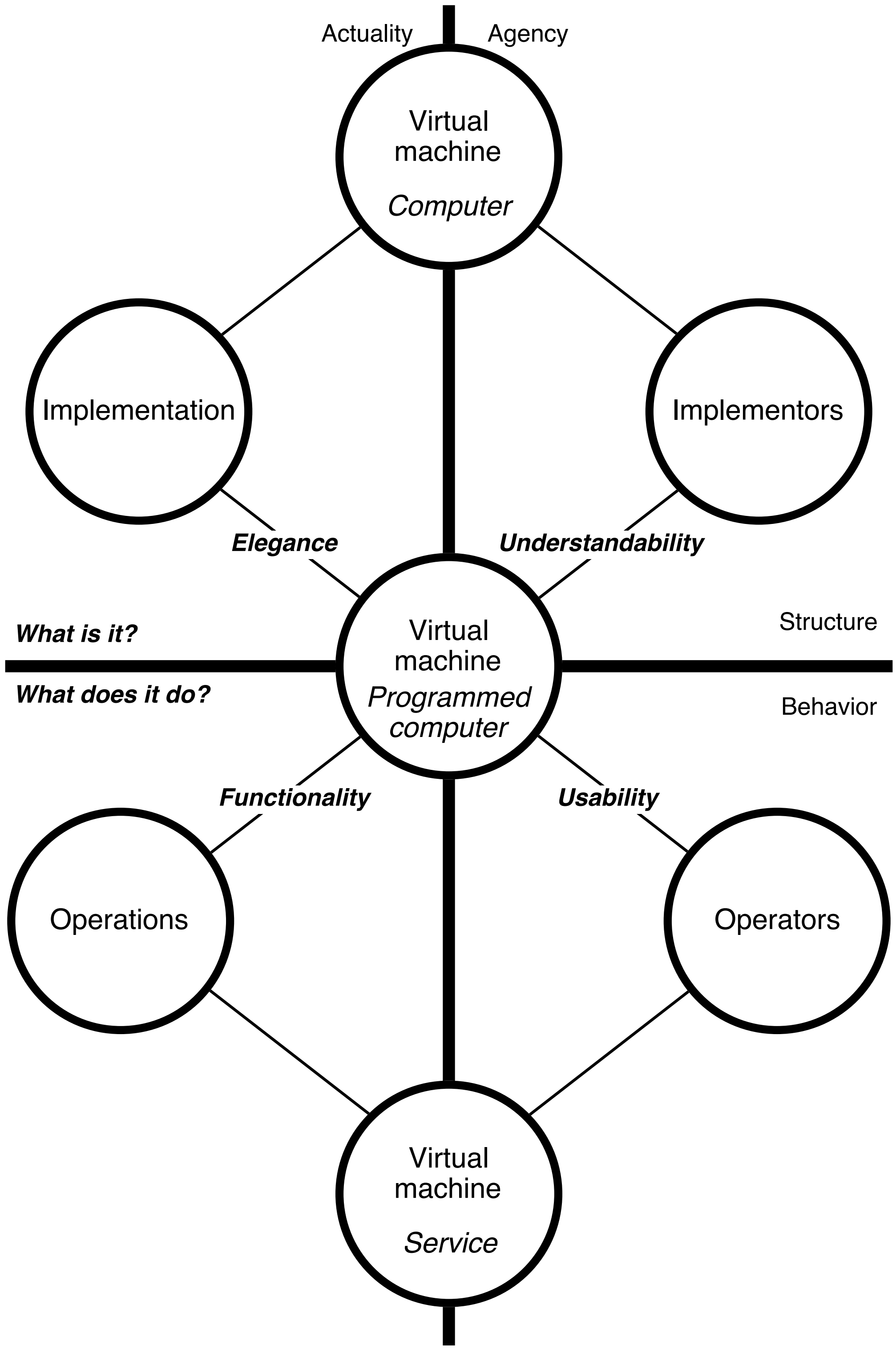

Figure 5 Virtual machines in terms of system origins and activity

The relations between the central virtual machine and its structural and behavioral components represent fundamental concepts in system implementation and operation (Edwards and Mason 1987):

- *elegance* considerations, capturing the notion of efficiency in design, concern the relation between the system virtual machine and the underlying resource with which it is implemented—this is another virtual machine capturing the characteristics of the high-level language, operating system, and so on, used in implementation;
- *understandability* considerations, capturing the notion of comprehensibility in design, concern the relation between the system virtual machine and the implementors responsible for creating it;
- *functionality* considerations, capturing the notion of capabilities provided, concern the relation between the system virtual machine and the tasks for which it is being implemented.
- *usability* considerations, capturing the notion of capabilities suitably interfaced, concern the relation between the system virtual machine and the operators who use it.

The logic underlying Figures 4 and 5 may be iterated to interpolate as much detail as required between the upper and lower levels—what is "activity" at one level becomes "structure" for the lower level. Figure 6 shows an analysis of the key features of a computer system in terms of six layers:-

- At the top level the overall computer system originates in terms of purpose and structure as part of the culture within which it embedded. This *cultural layer* captures the milieu within which the system has been generated and can itself be subject to detailed analysis.
- At the next level the activities generated through this structure and purpose lead to a system of intentionality in which anticipatory activity leads to the acquisition of knowledge, or models of the world, and the generation of goals. This *intentionality* layer captures the anticipatory nature of an intelligent system in supporting its survival through prediction and action.
- At the next level the activities generated through anticipation lead to actions and communications—a linguistic distinction between activities directed to World 1 effects and World 2 effects, respectively. This *knowledge* layer captures the internal processes supporting the modeling and control activities of an anticipatory system.
- At the next level the activities generated through action and communication have to be transmitted to some external medium. This *action* layer captures the internal processes supporting the interfacing of an anticipatory system to the world.
- At the next level the activities generated through transmission have to be expressed in such a way as to have the desired effect in the world. This *expression* layer captures the internal processes supporting the encoding of actions and communications.
- At the lowest level the activities generated through expression exist physically in the external world. This *physical* layer captures the external effects of the activities of an anticipatory system.

It is interesting to relate the iterated levels of Figure 6 back to the terminology of Figure 5. First, consider Figure 6 as a conceptual framework for a person. Systemically, the culture is the 'implementor' of the person, taking the genetic 'virtual machine' and socializing it to have intentions and knowledge that conform with the purpose of a particular society. The intentions of the person are carried out by the formation of anticipatory systems that acquire knowledge and generate goals. These in turn implement goal-directed actors or communicators, which themselves implement processes to transmit the activities of the actors to the world, ultimately as sequences of acts that are directly expressed in the physical world. Second, consider Figure 6 as a conceptual framework for a computer system, an intelligent integrated system. The diagram from bottom up represents the evolution of computer technology, from the physical device technology to carry out acts, through the programmed control of action, through the knowledge-based derivation of action in fifth-generation systems, through the goal-creating activities of intentionally-specified systems in future-generation computing, to the origins of information technology in our culture at the top level.

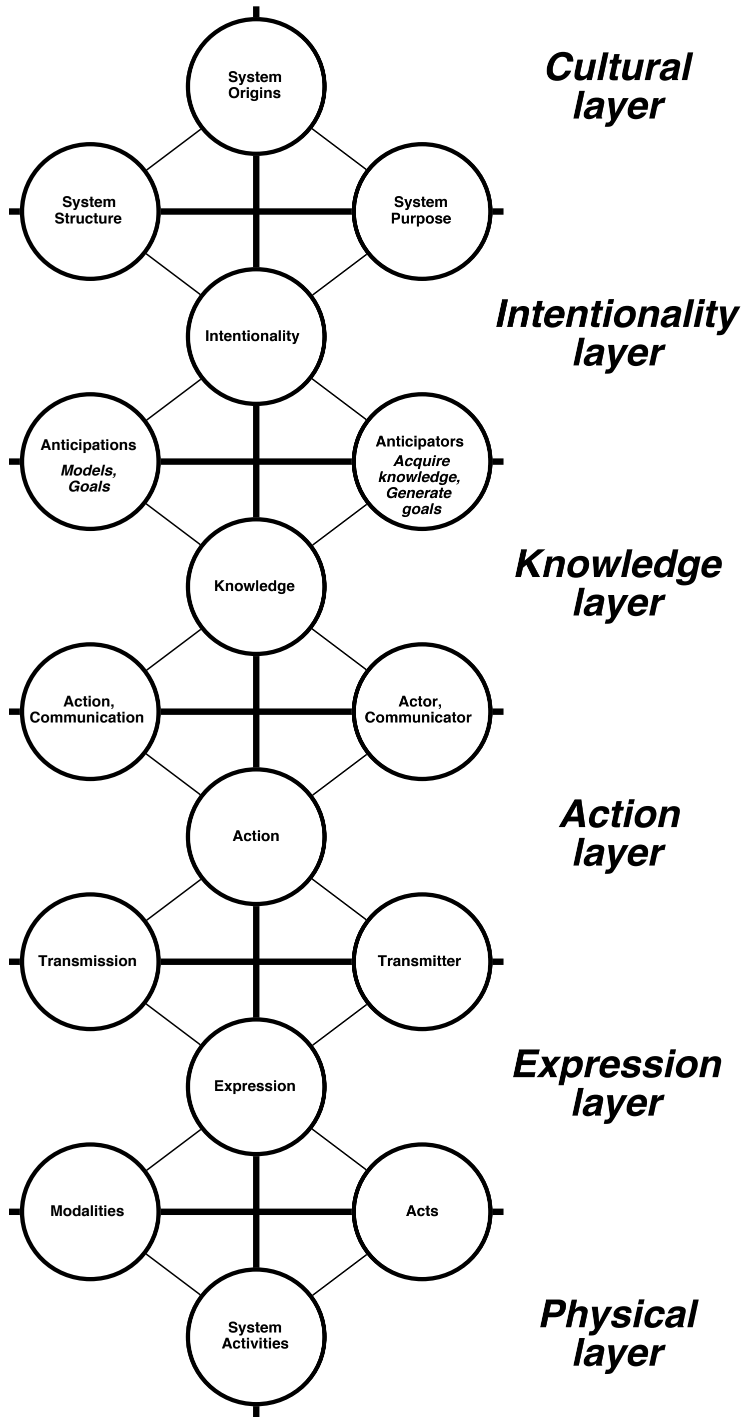

Figure 6 The hierarchy of layers in a computing system

Figure 7 extends Figure 6 to multiple systems showing the virtual circuits in, and between, two people communicating through a computer system. What is particularly interesting about this diagram is that the same distinctions, terminology and model are being applied to the people, their interactions with each other, their interface to the information technology, and its interface to other information technology. The same systems principles apply to the psychology, sociology, human-computer interaction, and computer-computer interaction because computing systems have the dual identity shown in Figure 2. They are both technological and humanistic systems and it is human component, the choice available in programming, that determines their roles and behavior in interacting with people.

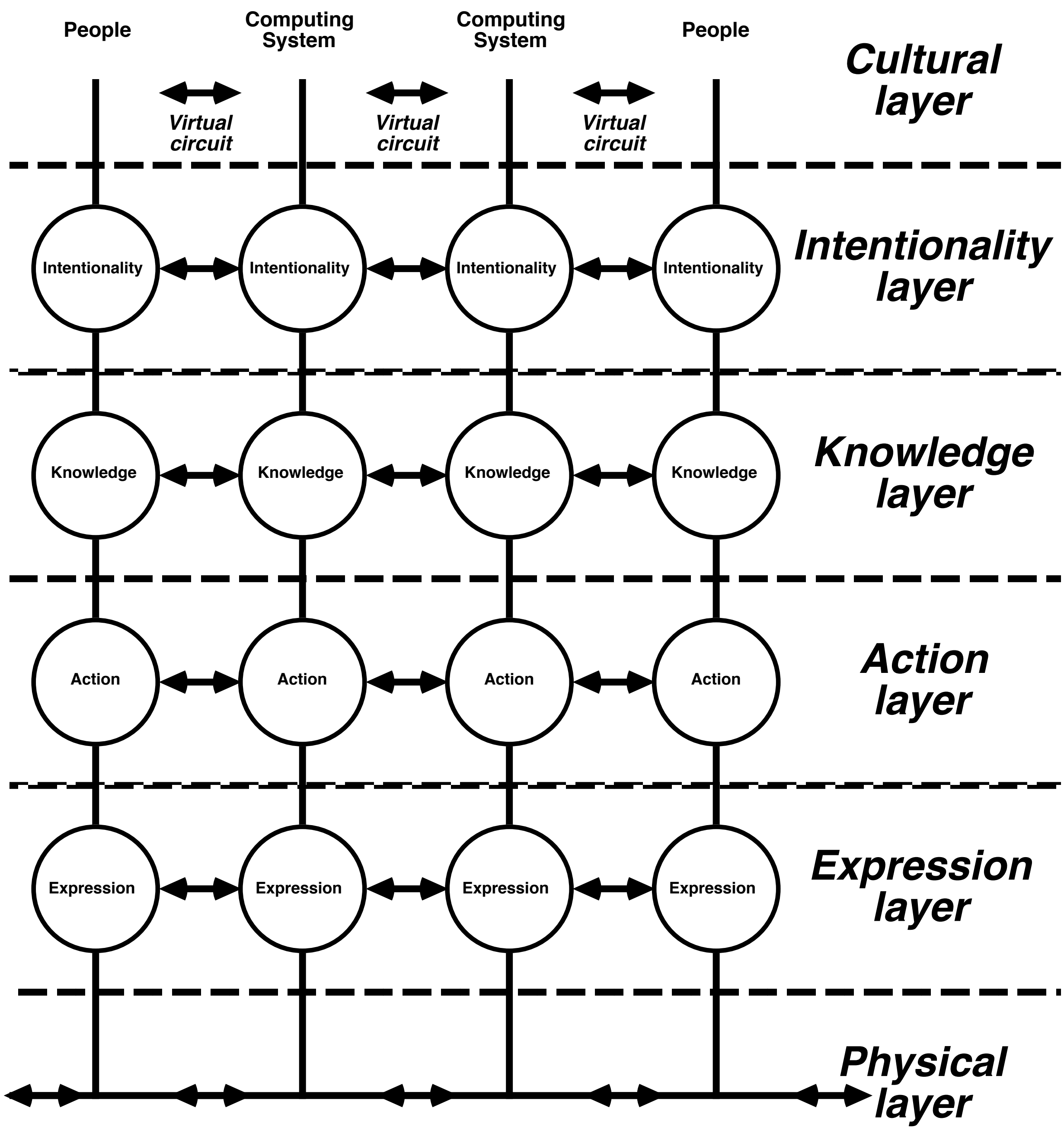

Figure 7 Virtual circuits between layers in computing systems

5 From Description to Prescription

The conceptual framework for person-computer interaction developed above is a descriptive knowledge structure showing what distinctions a system designer might be expected to make and how these derive from fundamental systemic principles. However, critical distinctions are the basis of the major prescriptive design rules that the designer has to keep in mind when making significant decisions. If the distinction is important to the design then it must be maintained or obviated and this requires two activities, each of which leads to design criteria which may be in opposition. First, the distinction may be considered to be significant to maintain and this leads to design considerations based on its maximizing its importance. Second, the distinction may be considered exaggerated and this leads to design considerations based on minimizing its importance. This possibility for opposition is what can lead to contradictory guidelines which are both significant to take into account (Maguire 1982).

For example, maintaining or the distinctions between layers in Figures 5 and 6 may lead to opposing design considerations. One group of designers may wish to maintain the importance of the intentionality layer and note that the intentions of the human participants in using the system are important. The users do not see themselves as interacting with interesting technology but as carrying out an organizational function. The sales manager does not see himself as interacting with a complex administrative and productive system, but rather as communicating directly with a customer organization by supplying it with goods. The chief executive does not see himself as concerned with that level of communication but rather as communicating directly with the board and shareholders through a flow of profits and dividends. The system must be designed around the perspectives of its users in terms of their intentions. However, another group of designers may wish to obviate this distinction and be concerned with users only in terms of their actions noting, perhaps, that in their situation the intentions of users are variable and not available to the system. It would be confusing for the system to make assumptions about, or attempt to infer, the user's intentions. It should be designed around the perspectives of users in terms of their actions.

Thus conceptual structures are not only descriptive, providing a vocabulary and semantic relationships for talking about interface design. They are also prescriptive in identifying the critical dimensions along which design decisions must be analyzed. At an abstract level a conceptual structure is arbitrary like the grid lines on a map serving a useful structure in dividing up a territory so that different parts of it may be discussed. However, the concrete conceptual structure of an expert is not arbitrary but more like the contour lines of map in following natural phenomena and significant boundaries. The distinctions made are significant in that they indicate differences where decisions are most likely to be relevant to the problems encountered.

6 Conclusions

Complex interactive systems are being built and, despite many problems, they will be made to work because they are needed to deal with some of the very complex activities necessary to our society. Conventional empirical studies of person-computer interaction based on single users operating at standard workstations do not, and cannot, give adequate foundations for the design and analysis of such systems. We have to go back to the basic distinctions between people, computing and person-computer interaction and develop conceptual structures that can encompass the problems and design considerations of complex interactive systems. The analysis developed in this paper is the first step towards a conceptual framework within which to analyze the structure and operation of the next generation of interactive systems.

Acknowledgements

Financial assistance for the studies on which it is based has been made available by the Natural Sciences and Engineering Research Council of Canada. I am grateful to the anonymous referees for detailed critical comments which have greatly improved the paper.

References

Bennett, J. (1976). **Linguistic Behaviour**. Cambridge, UK: Cambridge University Press.

Berne, E. (1974). **What Do You Do After You Say Hello?**. London: Andre Deutsch.

Cheriton, D.R. (1976). Man-machine interface design for time-sharing systems. **Proceedings of the ACM National Conference**, 362-380.

Day, J.D. & Zimmerman, H. (1983). The OSI reference model. **Proceedings IEEE**, **71**(12), 1334-1340 (December).

Edwards, J.L. & Mason, J.A. (1986). Evaluating the intelligence in dialogue systems. **International Journal of Man-Machine Studies**, **27**, to appear.

Feyerabend, P. (1975). **Against Method**. London: NLB.

Gaines, B.R. (1975). Analogy categories, virtual machines and structured programming. Goos, G. & Hartmanis, J., Eds. **GI—5.Jahrestagung**. **Lecture Notes in Computer Science**, **34**, 691-699. Berlin: Springer Verlag.

Gaines, B.R. (1976). On the complexity of causal models. **IEEE Transactions on Systems, Man & Cybernetics**, **SMC-6**(1), 56-59 (January).

Gaines, B.R. (1977). System identification, approximation and complexity. **International Journal of General Systems**, **3**, 145-174.

Gaines, B.R. (1980). General systems research: quo vadis? Gaines, B.R., Ed. **General Systems 1979**. Vol. 24, pp. 1-9. Kentucky: Society for General Systems Research.

Gaines, B.R. (1981). The technology of interaction—dialogue programming rules. **International Journal of Man-Machine Studies**, **14**(1), 133-150 (January).

Gaines, B.R. & Shaw, M.L.G. (1984). **The Art of Computer Conversation: A New Medium for Communication**. New Jersey: Prentice Hall.

Gaines, B.R. & Shaw, M.L.G. (1986). Foundations of dialog engineering: the development of human-computer interaction part II. **International Journal of Man-Machine Studies**, **24**(2), 101-123 (February).

Goguen, J.A. & Meseguer, J. (1983). Programming with parametrized abstract objects in OBJ. **Theory and Practice of Programming Technology**. Amsterdam: North-Holland.

Habermas, J. (1981). **The Theory of Communicative Action: Reason and the Rationalization of Society**. Boston: Beacon Press.

Hansen, W.J. (1971). User engineering principles for interactive systems. **Proceedings of the Fall Joint Computer Conference**, 39, 523-532. New Jersey: AFIPS Press.

Kelly, G.A. (1955). **The Psychology of Personal Constructs**. New York: Norton.

Klir, G.J. (1985). **Architecture of Systems Problem Solving**, New York: Plenum Press.

Mackenzie, B.D. (1977). **Beahviourism and the Limits of Scientific Method**. London: Routledge and Kegan Paul.

Maguire, M. (1982). An evaluation of published recommendations on the design of man-computer dialogues. **International Journal of Man-Machine Studies**, **16**(3), 237-261 (April).

Popper, K.R. (1968). Epistemology without a knowing subject. Van Rootselaar, B., Ed. **Logic, Methodology and Philosophy of Science III**. pp. 333-373. Amsterdam, Holland: North-Holland Publishing Co..

Rosen, R. (1985). **Anticipatory Systems**. Oxford: Pergamon.

Schilpp, P.A., Ed. (1974). **The Philosophy of Karl Popper**. Illinois: Open Court.

Schutz, A. & Luckmann, T. (1973). **The Structures of the Life-World**. London: Heinemann.

Skinner, B.F. (1974). **About Behaviourism**. London: Jonathan Cape.

Smith, S.L. & Mosier, J.N. (1986). **Guidelines for Designing User Interface Software**. ESD-TR-86-278. Bedford, Massachusetts: Mitre Corporation (August).

Taylor, M.M. (1987). Layered protocols for computer-human dialogue: I Principles. **International Journal of Man-Machine Studies**, **27**, to appear.

Ulrich, W. (1983). **Critical Heuristics of Social Planning**. Bern: Haupt.

Wiener, N. (1948). **Cybernetics**. Cambridge,Massachusetts: MIT Press.