A METHODOLOGICAL FRAMEWORK FOR THE DESIGN AND EVALUATION OF SOFTWARE IN SYSTEMS INVOLVING COMPLEX HUMAN-COMPUTER INTERACTION

Brian R. Gaines, Calgary

We are now in the fifth generation computing era with its emphasis on complex, knowledge-based systems involving close human-computer interaction. If we were able to use today's technology to instrument, model and understand the human-computer interfaces of yesterday software engineering for complex systems would be very much easier. However, the system designer is always one step ahead. The multi-task, multi-user, multi-modal systems of the fourth and fifth generations go beyond knowledge based on the technology of yesterday. If we continue to rely on empirical knowledge based on studies of the past we shall never be equipped to deal with the systems of the present let alone those of the future. This is the dilemma of current software ergonomics research. It can be resolved only through the development of foundational models of computers, people and human-computer interaction that can be projected to novel situations. This address reviews the state of the art in software engineering for complex systems involving computers, people and their interaction. It presents recent developments in methodological frameworks for designing and evaluating complex systems.

1 Introduction—Fifth Generation Objectives

The Japanese initiative in 1981 of scheduling a development program for a fifth generation of computers (Moto-oka 1982, Gaines 1984b) led to widespread realization that computer technology had reached a new maturity. Fifth generation computing systems would integrate advances in very large scale integration, database systems, artificial intelligence, and human computer interaction into a new range of computers that were closer to people in their communication and knowledge processing capabilities. It may be difficult to recapture the shock of this announcement: it was unforeseen, from an unexpected source, gave a status to human-computer interaction and artificial intelligence research that was yet unrecognized in the West, and proposed an integration of technologies that were still seen as distinct. Since then the fifth generation objectives have become accepted worldwide and led to many comparable research and development programs by other nations.

Human factors considerations were stated to be fundamental to the fifth generatiuon objectives. Moto-oka (1982) notes:
"intelligence will be greatly improved to match that of a human being, and, when compared with conventional systems, man-machine interface will become closer to the human system."
The fifth generation computer systems proposal may be seen as a natural response to advances in computer technology that have given us massive power in hardware and software at low cost

(Gaines 1984a). The technology which limited many aspects of human-computer interaction has now outstripped our demands and a shift may be expected from technology-push economics in computer systems to those of market-pull. Human-computer interaction is what the customer sees and is where the market requirements are being expressed. The fifth generation proposal as originally expressed is consistent with expectations that we will increasingly build systems top-down from user needs rather than bottom-up from technology availability.

However, the development of computing systems has generally been pragmatic with attempts to engineer systems whose activities go well beyond the science of their time. Our creative imaginations are usually well in advance of our scientific knowledge and skills. This applies with force to the fifth generation computer systems development program. Hardware and software are emphasized in the research program, but the ICOT research program has no human factors activities. Fuchi recognizes this problem in his reply to the interview question (Fuchi, Sato & Miller 1984):

"Are you saying that the design of the fifth-generation may be modeled by learning more about the human thinking process?"

answering:

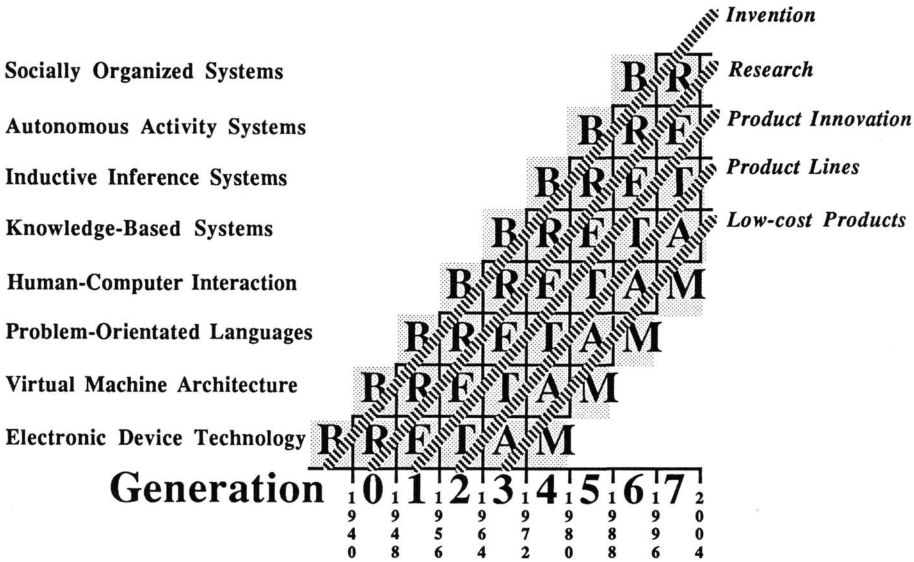
"Yes, we should have more research on human thinking processes, but we already have some basic structures. For more than 2000 years man has tried to find the basic operation of thinking and has established logic. The result is not necessarily sufficent; it's just the one that mankind found. At present we have only one solution - a system like predicate calculus. It is rather similar to the way man thinks. But we need more research. What, really, is going on in our brain? It's a rather difficult problem."

These problems are beginning to be addressed in the sixth generation development program which calls for collaboration between neurologists, psychologists, linguists and logicians (STA 1985, Gaines 1986a).

2 The Infrastructure of the Generations of Information Technology

The fifth generation proposals gave a tremendous boost to artificial intelligence research in the West and had some spin-off for human factors research. Human-computer interaction research in the West has a long history of multidisciplinary interaction with brain science, cognitive science, psycho-linguistics philosophy, an systems theory. However, this diversity of relationships has tended to give it a peripheral position in computer science. For the development of future generation computing sytems human-computer interaction must assume a more central role and become a core component of computing science curricula and research.

The role of human-computer interaction in fifth generation computing, and its implications for research and product development, can be analysed through analogy with other components of the infrastructure of computing. Like other basic technologies (Marchetti 1981), the growth of computing has an infrastructure comprised of the envelope of learning curves in successive underlying technologies (Ayres 1968). Figure 1 shows this structure through the zeroth through fifth generations projected to the sixth and seventh. Each technology has a learning curve in which a *breakthrough* leads to a phase of: *replication* in which the results are duplicated; *empiricism* in which rules for design are derived from experience; *theory* in which



B • Breakthrough: creative advance made
R • Replication period: experience gained by mimicking breakthrough
E • Empirical period: design rules formulated from experience
T • Theoretical period: underlying theories formulated and tested
A • Automation period: theories predict experience & generate rules
M • Maturity: theories become assimilated and used routinely

Figure 1 The infrastructure of information technology through eight generations of computing

generative principles are derived; *automation* in which the theory is made operational; and *maturity* when the knowledge becomes proven, widely available and widely used. In the development of computing each phase corresponds to a generation of about eight years linked in timing to the medium term business cycle (Gaines 1986b).

At the base is the learning curve for different electronic device technologies: 1940 zeroth generation using relays; 1948 first using vacuum tubes; 1956 second using transistors; 1964 third generation using using integrated circuits; 1972 fourth using large-scale integration; 1980 fifth using VLSI; 1988 sixth projected to use ultra large-scale integration with some 10 million transistors on a chip; 1996 seventh projected to use grand-scale integration with 1,000 million on a chip. The definition of generations in terms of EDT captures some important aspects of computing such as cost decreases, size decreases, power increases, and so on. However, it fails to account for the qualitative changes that have given computing its distinct character in each generation. These appear through the tiered succession of learning curves of higher level technologies based on the lower level developments.

The first generation breakthrough was the introduction of stored program and subroutine concepts around 1947 which detached computing as a separate discipline from electronics by substituting software for hardware in a *virtual machine architecture*. The second generation breakthrough was to bridge the gap between machine and task through the development of *problem-oriented languages* such as FORTRAN in 1956. The third generation breakthrough was to bridge the gap between the computer and the person with the development of interactive time-shared computers in 1964 allowing close *human-computer interaction.* The fourth generation breakthrough was in the early 1970s with developments in expert systems based on *knowledge-based systems*. The fifth generation breakthrough was in 1980 with developments in machine learning and *inductive inference systems*. One may speculate that the growth of robotics will provide the next breakthroughs in which goal-directed, mobile computational systems will act as *autonomous activity systems* to achieve their objectives, and that interaction between these systems will become increasingly important in enabling them to act as *socially organized systems* and cooperate to achieve goals.

This model of the development of computing is important in analysing the role of human-computer interaction studies and their changing priorities. In particular the emphasis of workers in a particular activity changes as the learning curve progresses so that: *invention* is focused at the **BR** interface where new breakthrough attempts are being made based on experience with the replicated breakthroughs of the technology below; *research* is focused at the **RE** interface where new recognized breakthroughs are being investigated using the empirical design rules of the technology below; *product innovation* is focused at the **ET** interface where new products are being developed based on the empirical design rules of one technology and the theoretical foundations of the technology below; *product lines* are focused at the **TA** interface where established products can rest on the solid theoretical foundations of one technology and the automation of the technology below; *low-cost products* are focused at the **AM** interface where cost reduction can be based on the the automated mass production of one technology and the mature technologies below.

For example, in the fourth generation (1972-79):

- **BR**: recognition of the knowledge acquisition possibilities of knowledge-based systems led to the breakthrough to inductive-infererence systems;
- **RE**: research focused on the natural representation of knowledge through the development of human-computer interaction, e.g. the Xerox Star direct manipulation of objects;
- **ET**: experience with the human-computer interaction using the problem-oriented language BASIC led to the innovative product of the Apple II personal computer;
- **TA**: the simplicity of the problem-oriented language RPG II led to the design of the IBM System/3 product line of small business computers;
- **AM**: the design of special-purpose chips allowed the mass-production of low-cost, high-quality calculators.

In the current fifth generation (1980-87):

- **BR**: recognition of the goal-seeking possibilities of inductive inference systems is leading to the breakthrough to automomous-activity systems in robotics;

- **RE**: research is focused on learning in knowledge-based systems;
- **ET**: the advantages of the non-procedural representation of knowledge for human-computer interaction led to the innovative designs of the Visicalc spread-sheet business product and the Lisp-machine scientific product;
- **TA**: the ease of human-computer interaction through a direct manipulation problem-oriented language led to the Apple Lisa/Macintosh product line of personal computers;
- **AM**: the design of highly-integrated language systems has allowed the mass-production of low-cost, high-quality software such as Turbo Pascal.

3 Software Ergonomics in the Information Technology Infrastructure

The horizontal **BRETAM** sequence for human-computer interaction has the *breakthrough* in 1963-64 with the development of systems such as MIT MAC (Fano 1965). In the *replication* period such systems came into widespread use well before the human factors principles underlying their design were understood. Hansen's (1971) tabulation of some user engineering principles for the design of interactive systems marks the transition to the *empirical* period. The transition to *theory* at the beginning of the 1980s was marked by studies of human-computer interaction developing theoretical foundations based on cognitive science. A reasonable expectation in the theory phase of fifth generation human-computer interaction is a set of principles that systematically generates rules for dialog engineering grounded in system theory, computer science and cognitive psychology. The principles should be applicable to the entire range of possible dialog styles and technologies, now and in the future, and operational so that they can be embedded in standard dialog shells applicable to all interactive systems (Gaines & Shaw 1984).

The vertical **BRETAM** sequence for the current fifth generation era shows the interplay between human-computer interaction and the other strata of computing. Fifth generation product lines are built upon the solid foundations of electronics, machine architecture and problem-orientated languages. The most recent developments to impact them are in human-computer interaction and these are the most critical to product differentiation. Fifth generation product innovation is based on the same technologies with the critical one being that of knowledge-based systems supporting intelligent user interfaces. Fifth generation research concentrates on the inductive aspects of knowledge acquisition, systems learning from people and from experience.

In 1988 we move into the sixth generation and the final phase of the learning curve for human-computer interaction. For any discipline in this phase research becomes harder, standards of refereeing become harsh and many researchers drop out (Crane 1972). Obtaining results in the final 10% of the curve requires a thorough understanding of the known 90%. It requires careful experimental design, precise theoretical formulations, meticulous engineering practice in system development. In short, it is a phase of professionalism.

4 A Knowledge-Based Approach to Software Ergonomics

The overall model of the infrastructure of information technology given above predicts that we are now moving into a phase of theoretical developments to underpin the empirical design of human-computer interaction. It also highlights the this fifth generation era as being one of innovation in knowledge-based systems. This suggests that advances in software ergonomics will come from the application of knowledge engineering to the development of theoretically well-founded conceptual frameworks for the relationships between people and computers. It is important that we look at the variety of relationships possible—not just the user at a computer terminal. Problems of system analysis, programming, maintenance, operational techniques, training, and upgradability, are all very significant to the viability and effectiveness of complex human-computer systems. Software ergonomics has to encompass a wide variety of technical and human factors relations, design considerations, opportunities and problems.

Modality
- physical characteristics of interface

Form
- structure of message

Connectivity
- links between different segments of message

Control
- responsibility for initiating and directing dialog

Knowledge representation/inferencing
- use of knowledge to generate dialog

Models
- knowledge of each party in dialog

Knowledge acquisition
- procedures for updating knowledge

Information sources
- access to external knowledge

Aspects of Dialog

		System Structure	Behavior
Observer	Logical	Elegance *Efficient design*	Functionality *Potential capability*
	Psychological	Understandability *Comprehensible design*	Suitability *Usable capability*

Dimensions within Aspects

Fig.2 Synopsis of dialog system evaluation scheme

Edwards and Mason (1987) have recently proposed an evaluation methodology for complex human-computer systems, and illustrated its application to a variety of decision-support systems involving multiple tasks, multiple users, and multiple modalities. Figure 2 is a synopsis of their methodology. They analyze an intelligent dialog system in terms of the eight aspects shown at the top of Figure 2, and then evaluate it in terms of the four dimensions shown at the bottom of the figure: *elegance*, capturing the notion of efficient design; *understandability* capturing the notion of comprehensible design; *functionality* capturing the notion of potential capability; and *suitability* capturing the notion of usable capability. They apply to each aspect of an intelligent dialog system as evaluative criteria, and thus each may be regarded as mapping into a basic order relation of preference, that one system on a given dimension in a given aspect is better than another, the *worse—better* distinction. For practical purposes, as is common in psychological scaling, Edwards and Mason assume that the preference order can be approximated by a well-ordered structure and hence expressed numerically. This generates an evaluative scheme based on ratings of the eight aspects on each of the four dimensions.

This approach is successful in practice but phenomenological in its foundations. The following sections show how a knowledge-based systemic analysis can regenerate the methodology in such a way that the underlying theoretical principles are fully exposed. These can then be used as the foundations for deeper analysis and design principles.

5 Principles of Systemic Analysis

Systemic analysis is based on the observation that there are common patterns in the way in which we model the world, explain phenomena, anticipate events and communicate knowledge. These patterns are part of the process of human understanding and may be termed general systems principles. Their significance is that, since they underly physical laws and social conventions, their identification in particular modeling schema enables us to analyze those schema in a universal framework. The extraction of these systemic principles is straightforward if we examine the systems of *distinctions* being used in the evaluation and protocol methodologies. By analysing the distinctions made in any discipline:

"we can begin to reconstruct, with an accuracy and coverage that appear almost uncanny, the basic forms underlying linguistic, mathematical, physical and biological science, and can begin to see how the familiar laws of our own experience follow inexorably from the original act of severance." Brown (1969)

In terms of cognitive psychology, such distinctions are the *constructs* underlying our modeling of the world Kelly (1955).

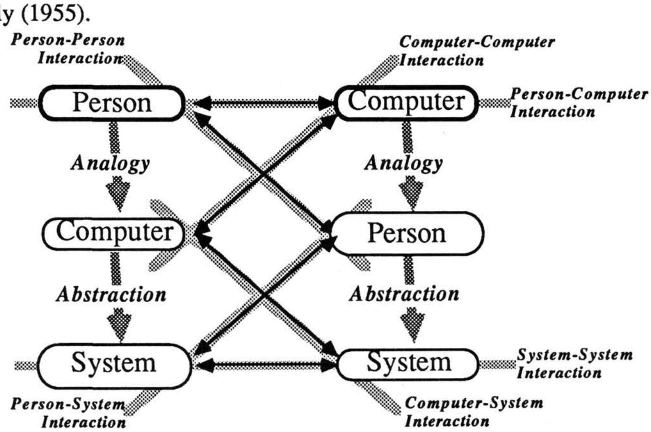


Fig.3 Processes of analogy and abstraction in the analysis of person-computer interaction

Distinctions arising from processes of abstraction and analogy are significant reasoning techniques in human modeling of experience and have important roles to play in the study of human-computer interaction. There are analogies between computers and people, and both may be regarded as systems in abstract terms. Figure 3 illustrates the way in which five additional forms of interaction arise when possible when abstraction and analogy are applied to the analysis of person-computer interaction. We draw on experience of these other five when developing models and guidelines for person-computer interaction. For example, Edwards and Mason draw on the *computer is-similar-to-a person* analogy in their use of concepts such as *knowledge* in a computer context, and on the *person is-a system* and *computer is-a system* abstractions in their use of concepts such as *control* in a dialog context.

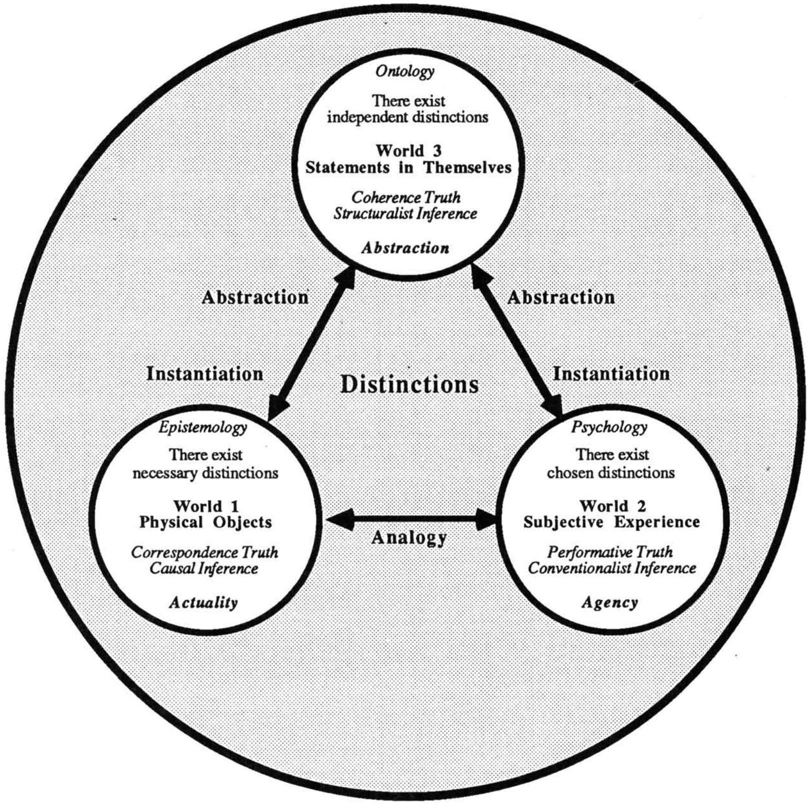


Fig.4 Actuality, agency and abstraction as basic distinctions among distinctions

The basic systemic distinctions underlying the abstractions and analogies shown in Figure 3 can be analyzed in terms of Popper's (1968) *3 worlds theory*. 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." Figure 4 shows the existential hypotheses underlying Popper's 3 worlds as very general distinctions made about distinctions:

• World 1 arises from the hypothesis that there exist *necessary* distinctions
—the conceptual framework involved is that of *epistemology*, and a key concept is that of *actuality*

—truth in this world relates to *correspondence* between distinctions made and the properties of physical objects, and hence inference is *causal*.

- World 2 arises from the hypothesis that there exist *chosen* distinctions
  —the conceptual framework involved is that of *psychology*, and a key concept is that of *agency*
  —truth in this world relates to the ongoing consistency of subjective choices of distinctions to make and is *performative* in nature, and hence inference is *conventionalist*.
- World 3 arises from the hypothesis that there exist distinctions *independent* of their source in actuality or agency
  —the conceptual framework involved is that of *ontology*, and a key concept is that of *abstraction*
  —truth in this world relates to the internal *coherence* of the systems of distinctions made, and hence inference is *structuralist*.

These basic distinctions give rise to the notions of abstraction and instantiation as relations between world 3 and worlds 1 and 2 as shown, and also to that of analogy between worlds 1 and 2 when we attribute to agency to the causal dynamics of physical objects or necessity to the social conventions of human activity.

In the analysis of system dynamics a fundamental distinction is 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 (Klir 1985), that is interrelating the necessities of world 1 behavior and structure. However, intelligent dialog systems are not purely physical systems since they involve the choice behavior of people and hence show phenomena of the life-world (Schutz & Luckman 1973) which are essentially different from those of the physical world and cannot be encompassed by causal models (Ulrich 1983).

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 5 shows an analysis of the relations between a system, its origins and its activities, when the *actuality—agency* distinction of Figure 4 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.

Figure 5 provides the systemic basis for the detailed analysis of human-computer interaction, and in the following sections it will be used to develop models of software ergonomics.

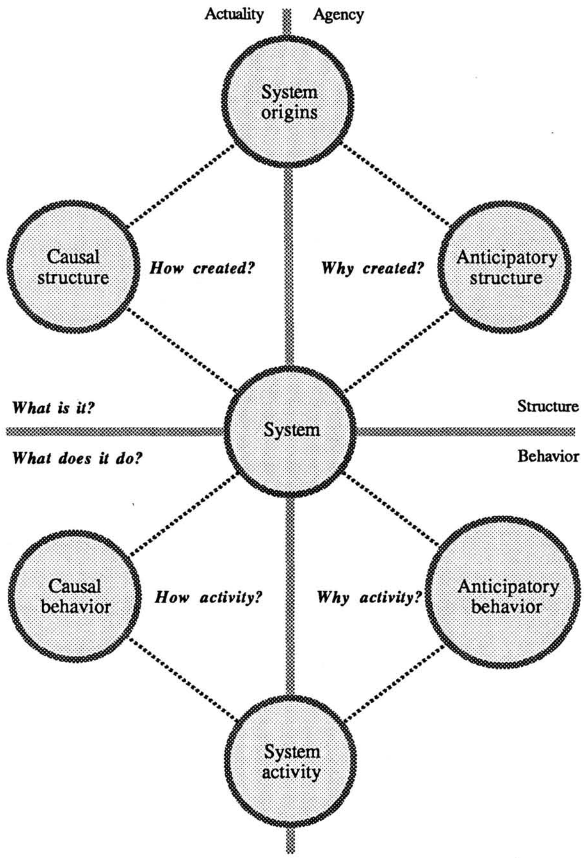


Fig.5 System structure and behavior related to actuality and agency

6 Distinctions in Evaluating Human-Computer Interaction

Each distinction made generates a *system* (Gaines 1980) and the evaluative sub-dimensions correspond to relations between these systems. The evaluative methodology is concerned not only about system use but also about system implemention, and hence about the relations between the computing system used in implementation and the virtual machine for intelligent dialog implemented upon it. Nelson (1980) and Smith (1983) have emphasized that the user sees the virtual machine not the underlying one, and it is this virtual "reality" that underlies the user inteface in machines such as the Star and Macintosh. However, the computing system itself may be regarded as a further virtual machine implemented in lower level hardware and software facilities, and the relation of implementors users and tasks to these levels of virtual machines is an important basis for a formal human factors analysis of the overall system (Gaines 1975, 1979).

Figure 6 fills in Figure 5 to show how the levels of virtual machine arise from the basic distinctions of structure and behavior, and how the evaluative dimensions arise from the basic

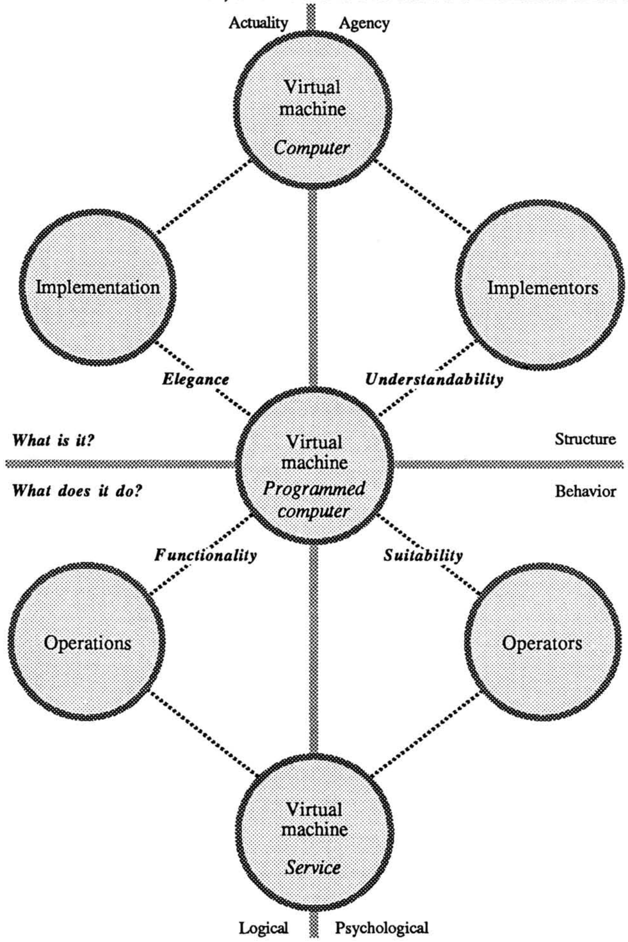


Fig.6 Dimensions of evaluation of an intelligent dialog system

distinctions of actuality and agency. The origins of the intelligent dialog system 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 intelligent dialog system as a programmed computing system. The activities of the intelligent dialog system 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 intelligent dialog system being used to provide a service. The dimensions in the lower part of Figure 2 may now be analyzed as relations between the central virtual machine and its structural and behavioral components:

- The *elegance* considerations logically and internally 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.
- The *understandability* considerations psychologically and internally concern the relation between the system virtual machine and the implementors responsible for creating it.
- The *functionality* considerations logically and externally concern the relation between the system virtual machine and the tasks for with which it is being implemented.
- The *suitability* considerations psychologically and externally concern the relation between the system virtual machine and the operators who use it.

7 Knowledge Flows in Intelligent Dialog Systems

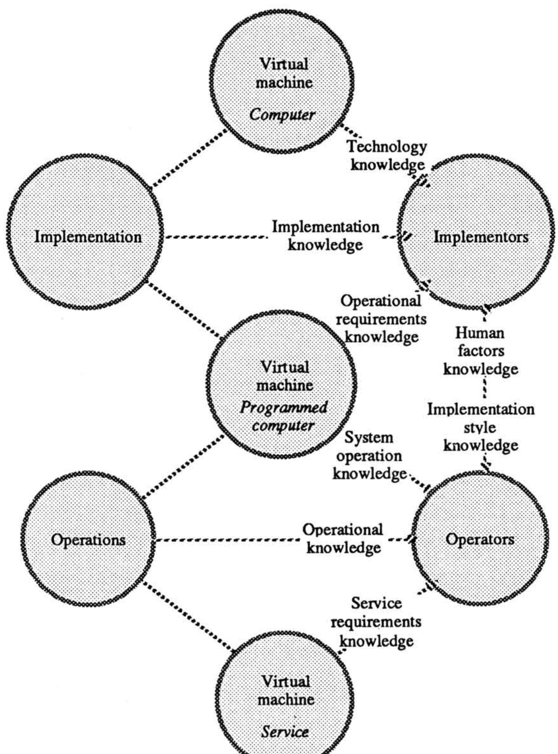


Fig.7 Knowledge flows in an intelligent dialog system

This derivation of the systems and relations underlying Edwards and Mason's evaluative distinctions enables other aspects of the system implementation and operation to be analysed. For example, Figure 7 shows the knowledge flows necessary to the implementation and operation of an intelligent dialog system:

- The implementors need:
  - —*technology knowledge* about the capabilities of the virtual machine being used as the implementation resource;
  - —*implementation knowledge* about how to create the virtual machine being used as an intelligent dialog system;
  - —*operational requirements knowledge* about the operations which the implemented system has to be able to carry out;
  - —*human factors knowledge* about the operators who will use the system.
- The operators need:
  - —*implementation style knowledge* about the way in which the system is intended to be used;
  - —*system operation knowledge* about the facilities available on the implemented system;
  - —*operational knowledge* about how to use the facilities available on the implemented system to provide a service;
  - —*service requirements knowledge* about the service which the implemented system has to be able to provide.

These eight forms of knowledge are each important to the overall system performance. If the implementor lacks one or more of the four sources of knowledge listed then the implementation is likely to have corresponding faults. It will be inefficient in its use of resources, inelegant in its implementation, disfunctional with regard to the requirements, or unsuited to the operators. Similarly, if the operator lacks one or more of the four sources of knowledge listed then the operation is likely to have corresponding faults. It will be inappropriate in its mode of operation, inefficient in its use of resources, inelegant in its operation, or inappropriate to the service required.

8 Maintainability, Upgadability and Flexibility

There is a continuity between implemenation and operation that is not apparent in Figure 6, yet implicit in its derivation from the analysis of Figure 5. Implementors and operators may both be seen as agents within the system, distinguished by being not part, or part, of its continuing activity, respectively. The *implementor—operator* distinction allows for a continuity of sub-distinctions which are in themselves very significant. For example, consider the extension of Figure 6 to the case where a number of distinctions are made between roles of implementors and operators: that some are system implementors; others system maintainers; others system enhancers; others system appliers; and others system operators. As shown in Figure 8, these new distinctions generate the possibility of many new relations.

Each of these forms of agency has a corresponding form of actuality with its own relation to the intelligent dialog system:

- The *implementors* are concerned with the *naturality* of the top level virtual machine as a resource for implementing the dialog system, and this is a major factor in the *elegance* of the implementation;
- The *maintainers* are concerned with the *understandability* of the implementation, and this is a major factor in the *maintainability* of the system, its improved implementation;

- The *enhancers* are concerned with the *enhancability* of the implementation in order to provide new features, and this is a major factor in the *upgradability* of the system,which may involve both new system implementation and new operational procedures;
- The *appliers* are concerned with the *applicability* of the system to new situations, and this is a major factor in the *flexibility* of the system in providing extended operations beyond those foreseen;
- The *operators* are concerned with the *usability* of the system, and this is a major factor in harnessing the *functionality* of the system to provide a service.

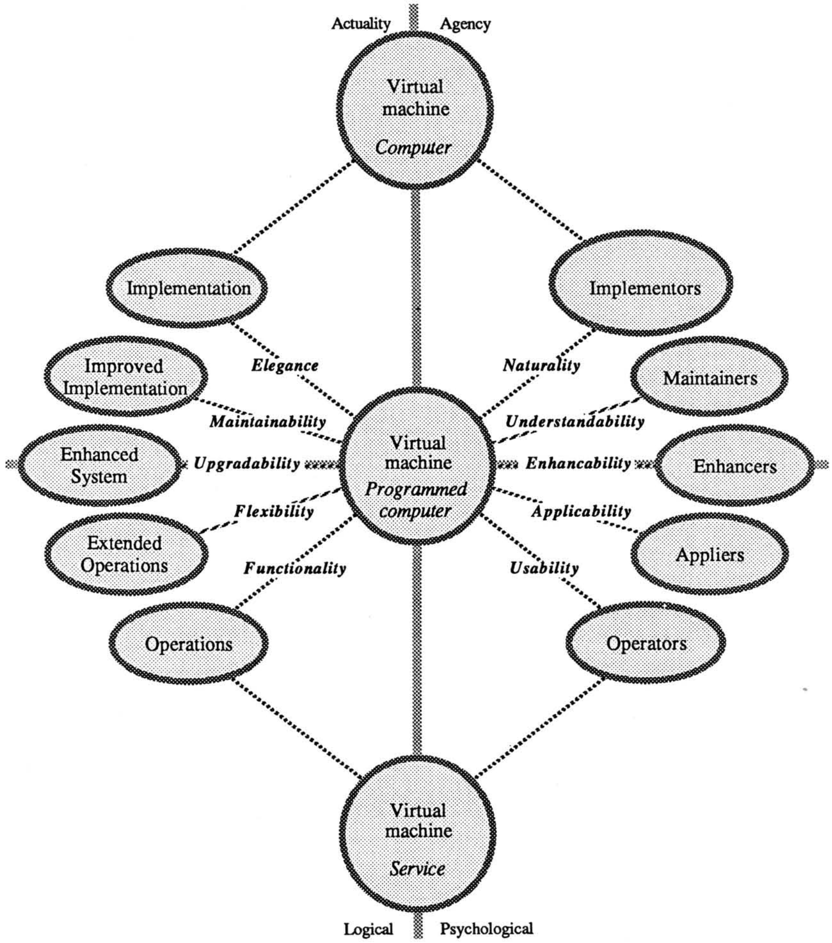


Fig.8 Extended dimensions of evaluation of an intelligent dialog system

These additional distinctions are all of great significance to the application of an intelligent dialog system. Maintainability, upgradability and flexibility are important characteristics of complex and powerful systems. The analysis above shows that the evaluative framework can be extended to encompass other aspects of evaluation, and that the relation between these can be itself analyzed in terms of basic systemic distinctions.

9 A Hierarchy of Layers in Intelligent Dialog Systems

The analysis of the previous sections has concentrated on the evaluation of intelligent dialog systems in terms of the relationships of the computational and human components. However, the central virtual machine that forms the intelligent dialog system as shown in Figure 8 is itself a complex structure with many internal sub-systems, relations and behavior. It is these that are analysed in terms of eight *aspects* shown in the upper part of Figure 2.

Figure 9 shows a top-down analysis of the structure of an intelligent dialog system based on a four-fold iteration of the systemic analysis of Figure 6:-

- At the top level the overall intelligent dialog system originates in terms of purpose and structure, and this results in activity in the form of anticipation with knowledge acquisition leading to the formation of models. This is termed the *intentionality* layer since it primarily concerned with the goals of the dialog system. Note that 'acquisition' is used here in an anticipatory systems sense to encompass both perception and action, and hence to encompass planning.
- At the next level knowledge originates from the modeling process, and this results in activity in the form of dialog with control of interaction based on the knowledge representation. This is termed the *knowledge* layer since it primarily concerned with the use of knowledge to guide the dialog system.
- At the next level the actual dialog originates from the control and knowledge, and this has form and connectivity. This is termed the *protocol* layer since it primarily concerned with the internal structure of the dialog.
- At the next level the messages which constitute the dialog originate from the form and connectivity, and this results in activity at the level of physivcal modalities and psychological acts. This is termed the *message* layer since it primarily concerned with the actual message structure.

Two additional layers are shown to complete Figure 9:-

- At the top the *cultural* layer captures the social infrastructure which which the dialog is taking place and where overall cultural pressures influence individual agents' intentions.
- At the bottom the *physical* layer captures the actual transmission medium along which messages pass.

Seven of the aspects of dialog listed in Figure 2 are represented in Figure 9. The eighth is *access to information sources* , and this may be represented by considering the communication paths from the dialog system to other systems. Figure 10 shows the layered structure in three different types of communicating entity: an information source; an intelligent dialog system; and a person. The information source is possibly another dialog system or person. However, it may also be some form of data or knowledge base in which the intentionality layer is virtually non-existent. The intelligent dialog system may itself be weaker at the intentionality layer than a person, but will be expected to have some activity at this level to justify the term "intelligent." In conversation-theoretic terms (1980), Figure 10 may be seen as representing the process whereby

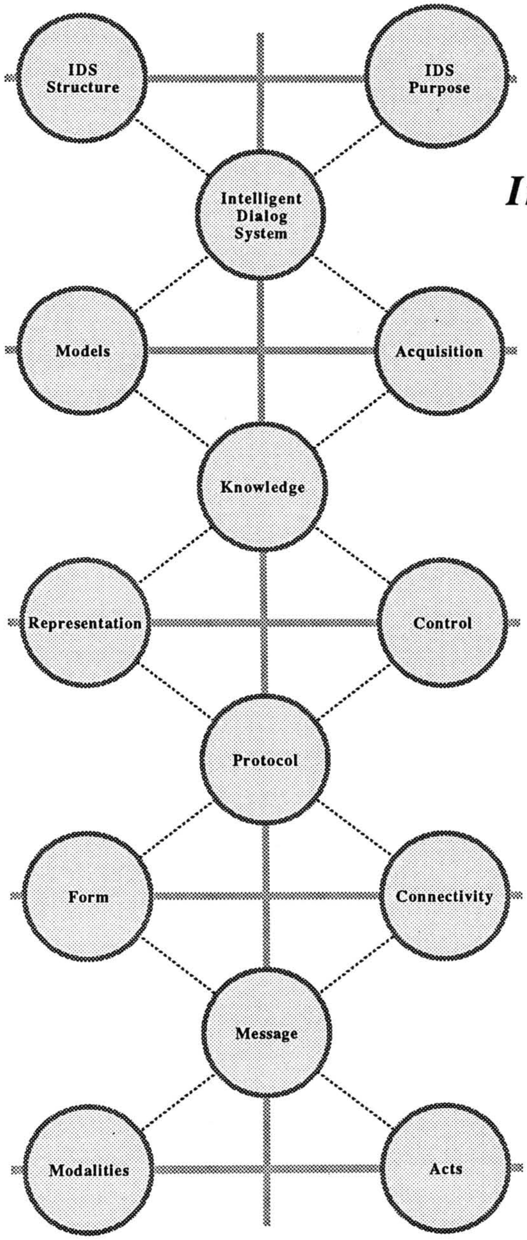


Fig.9 Hierarchy of layers in an intelligent dialog system

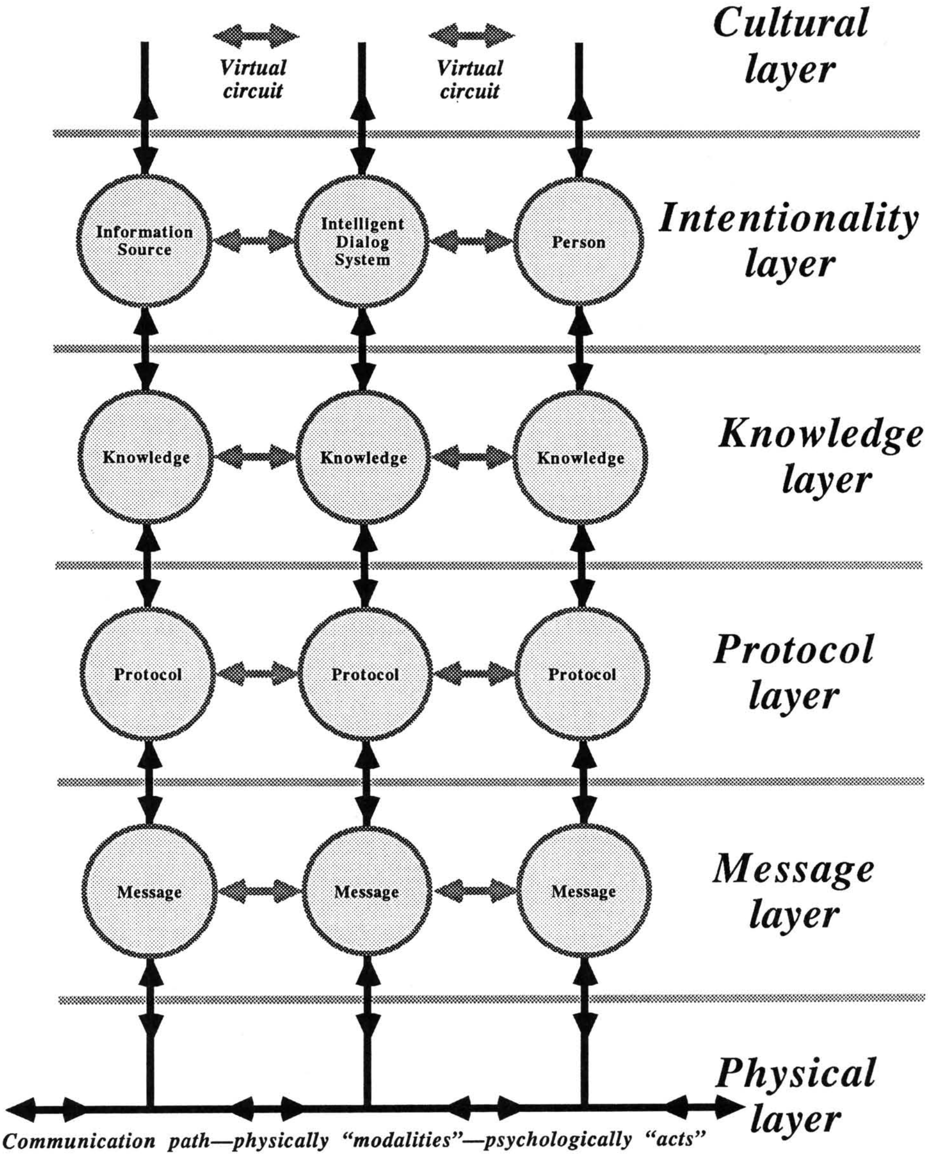


Fig.10 Layers and virtual circuits in the interaction of an intelligent dialog system with people and information sources

agents use a shared communications medium to establish relations between their otherwise unshared intentionality and knowledge systems. In communication-theoretic terms, the *virtual circuits* shown represent the conceptual paths through which communication of messages, interaction of protocols, exchange of knowledge, interaction of intentions, and formation of culture, appear to take place.

Thus the aspects of dialog singled out for evaluation fit naturally into a layered model (Taylor 1987). Dialog takes place as a manifestation of the actions of agents based on their intentions leading to goals, plans and communicative actions. The actions are generated based on knowledge, and structured through a protocol into messages.

10 Conclusions

The fifth generation developments of complex, knowledge-based decision-support systems involving multiple tasks, multiple users, and multiple modalities over-extends current guidelines for the design of human-computer interaction. We need to begin to replace empirical guidelines with deeper systemic theories that extend to a far wider range of complex syustems. This paper has presented an analysis of the systemic princples underlying human-computer interaction in order to provide a deep, systemic theory of software ergonomics.

11 Acknowledgements

Financial assistance for this work has been made available by the National Sciences and Engineering Research Council of Canada. The work on a systemic model described in this paper has benefited from detailed discussions with Jack Edwards and Martin Taylor.

12 References

Ayres, R.U. (1968). Envelope Curve Forecasting. **Technological Forecasting for Industry and Government.** pp. 77-94. New Jersey: Prentice Hall.

Brown, G.S. (1969). **Laws of Form.** London: George Allen & Unwin.

Crane, D. (1972). **Invisible Colleges: Diffusion of Knowledge in Scientific Communities.** Chicago, Illinois: University of Chicago Press.

Davis, R. & Lenat, D.B. (1982). **Knowledge-Based Systems in Artificial Intelligence.** New York: McGraw-Hill.

Edwards, J.L. & Mason, J.A. (1987). Evaluating the intelligence in dialogue systems. **International Journal of Man-Machine Studies, 27**(1), to appear.

Fano, R.M. (1965). The MAC system: a progress report. Sass, M.A. & Wilkinson, W.D., Eds. **Computer Augmentation of Human Reasoning.** pp. 131-150. Washington D.C., USA: Spartan Books.

Fuchi, K., Sato, S. & Miller, E. (1984). Japanese approaches to high-technology R&D. **Computer**, 17(3), 14-18 (March).

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. (1979). The role of the behavioural sciences in programming. **Structured Software Development Vol.2**. pp. 57-68. Maidenhead, UK: Infotech International.

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. (1984a). A framework for the fifth generation. **Proceedings of National Computer Conference**, 53, 453-459. Arlington, Virginia: AFIPS Press.

Gaines, B.R. (1984b). Perspectives on fifth generation computing. **Oxford Surveys in Information Technology**, 1, 1-53.

Gaines, B.R. (1986a). Sixth generation computing: a conspectus of the Japanese proposals. **ACM SIGART Newsletter**, No.95, 39-44 (January).

Gaines, B.R. (1986b). Socio-economic foundations of knowledge science. **Proceedings of 1986 International Conference on Systems, Man and Cybernetics**. Vol. IEEE 86CH2364-8, pp. 1035-1039. Atlanta (October).

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.

Nelson, T. (1980). Interactive systems and the design of virtuality. **Creative Computing**, **6**(11), 56-62 (November).

Marchetti, C. (1981). Society as a learning system: discovery, invention and innovation cycles revisited. Report RR-81-29 (November). Laxenburg, Austria: International Institute for Applied Systems Analysis.

Moto-oka, T. (1982). **Fifth Generation Computer Systems.**. Amsterdam: North-Holland.

Pask, G. (1980). Developments in conversation theory—Part I. **International Journal of Man-Machine Studies, 13**(4), 357-411 (November).

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

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

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

Smith, D.C. (1983). Visibility in the Star user interface. **Proceedings of Graphics Interface '83**, 181. Ottawa: National Research Council of Canada.

STA (1985). Promotion of Research and Development on Electronics and Information Systems That May Complement or Substitute for Human Intelligence. Tokyo: Science and Technology Agency.

Taylor, M.M. (1987). Layered protocols for computer-human dialogue: I Principles. **International Journal of Man-Machine Studies, 27**(1), this issue (July).

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

Address of Author

Professor Dr Brian R. Gaines, Killam Memorial Research Professor
Department of Computer Science, University of Calgary, Alberta, Canada T2N 1N4

I BENUTZERGRUPPEN UND IHRE AUFGABE