Gaines, B. R. (1984). From ergonomics to the fifth generation: 30 years of human-computer interaction studies. Computer Compacts 2, 158-161.

From Ergonomics to the Fifth Generation: 30 Years of Human-Computer Interaction Studies

Brian R. Gaines

Department of Industrial Engineering, University of Toronto, Ontario, Canada M5S 1A4

From the earliest days of computer systems it was realized that computers were essentially tools to be used by people and that good human factors were an essential part of system design. In the early years the struggle to generate reliable, low-cost hardware and software dominated the industry and human factors played a minor role in computer science and technology. In the past decade, however, advances in VLSI and software engineering have made advanced computer systems increasingly widely available and manufacturers have turned to human factors for product differentiation. In particular, human-computer interface requirements have a dominant role in fifth generation computer specifications. This paper surveys the development of human-computer interaction research from the 1940's through to the present day. The goals and expectations of the research, trends in the literature, the status of the results, and the directions of future development, are discussed.

1. In the Beginning

In this description of the development of human-computer interaction (HCI) it is appropriate to review computer history and the growth of human factors studies concerned with all aspects of computing. It is tempting to commence in 1879 with the Merrifield committee's concern that Babbage's analytical engine might be abused by people who used it for *Sisyphean* tasks. They take logarithms as a comparison and note: 'Much work has been done with them which could more easily have been done without them...more work has been spent on making tables than has been saved by their use.' [17] Any computer manager who has spent six years laboriously rolling the stone of a new installation of one generation to a peak of cost-effective service and is then faced with starting all over again with an incompatible next generation machine knows the meaning of a Sisyphean task. We have been guilty of that abuse and have learned to pay for emulation, compatibility, portability, enhancibility, modularity, integratibility and the like.

Probably, a proper starting point for concern with human factors lies with Mauchly who, in discussing EDVAC programming in 1947 notes the importance of ease of use of subroutine facilities, remarking: 'Any machine coding system should be judged quite largely from the point of view of how easy it is for the operator to obtain results.' [16] Replace 'machine' by 'virtual machine' and you have an aphorism which lies at the heart of HCI design today. However, in the early years the problems of making any form of operational computer far outweighed such ease of use considerations, and we have to move on some twenty years to see the beginning of human factors studies of HCI as we know them today.

Brian Gaines received his BA in Mathematics and Psychology from Cambridge University (UK) and a PhD in Psychology from the same university. He is Professor of Computer Science at York University and has adjunct appointments at the Universities of Toronto and Calgary. His primary research interests are in the fields of expert systems and human-computer interaction.

2. Interactive Systems

Shackel's 1959 paper [24] on the ergonomics of a computer console is an isolate. Ten years later, in surveying work on man-computer interaction, Nickerson [21] remarks on its paucity and quotes Turoff to the effect that psychology should be able to contribute greatly to the design of interactive systems: '...when one looks at some of the current systems of this nature, it becomes quite evident that the evolution of these systems has not been influenced by this field.' It is salutary to note that Nickerson wrote a paper 24 years later entitled 'Why interactive computers are sometimes not used by people who might benefit from them' [22], and I expect that he, and many at this meeting, will be prepared to echo Turoff's remarks today.

The first time-shared interactive systems, MAC, JOSS and BASIC, came on the air from 1963 onwards and stimulated interest in the human factors problems of non-specialist users. In 1967 Mills remarked: '...the future is rapidly approaching when 'professional' programmers will be among the least numerous and least significant system users.' [18]

Hansen [13] at the 1971 FJCC seems to have made the first attempt to tabulate some user engineering principles for the design of interactive systems.

3. The Early Literature

The landmark year was 1969: Ergonomics had a special issue based on papers to be given at an International Symposium on Man-Machine Systems; the IEEE Transactions on Man-Machine Systems reprinted the same papers to give them wider circulation; and the *International Journal of Man-Machine Studies* commenced publication. As editor of *IJMMS* I can attest to the difficulty of obtaining true human factors material for publication in those days. As a scientific discipline the field did not yet exist, but what we could pass on was a wide variety of user experience of interactive systems applied to many tasks. The 1969 issues contain papers on teaching systems, learning machines, natural language processing, speech recognition, radiological reporting, automated

psychological testing, and air traffic control. Computers are stimulating and the world was alive with imaginative computer applications from the earliest days onwards. It just took a long time for our scientific knowledge and professional skills as psychologists to begin to catch up with our creative imaginations as computer users.

Sackman's *Man-Computer Problem Solving* [23] in 1970 and Weinberg's *Psychology of Computer Programming* [36] in 1971 did much to stimulate interest in the possible applications of human factors principles in computing science. I would date the beginnings of experimental psychological interest in HCI as 1973 with the publication of Sime, Green and Guest's paper [31] on the 'Psycological evaluation of two conditional constructions used in computer languages.' The same year saw the publication of Martin's *Design of Man-Computer Dialogues* [15] and Wasserman's paper at the NCC on 'The design of 'idiot-proof' interactive programs' [34]. The choice of a less insulting term for naive, non-professional, or casual users has been a continuing problem.

4. Growth in the Literature

The mid-1970's was the beginning of the era of the personal computer, and the availability of low-cost computers with graphic displays led to their increasing use in psychological studies, and a boom in the associated literature. The decline in computer costs and the decreasing differences in computer facilities led to increasing commercial interest in good human factors as a marketing feature. *Ease-of-use* and *user-friendliness* began to be seen as saleable aspects of computer systems, and human engineers as product generators. Labor organizations intensified commercial interest as they promoted legislation relating to human factors of computer systems in the workplace, particularly the ergonomics of displays [10]. The papers from commercial sources further expanded an already swelling literature.

Conference organizers on almost any computing topic felt it timely to have a session on the human factors associated with it. More and more sessions at human factors meetings were devoted to HCI topics. The 1980's saw many books on HCI [1,2,3,11,12,25,26,29,30,33] the monthly publication of *IJMMS* and two new journals on human factors in computing, *Behavior and Information Technology* and *Human-Computer Interaction*.

5. Fifth Generation Computing

The culmination of this interest may be seen in the Japanese announcement in 1981 of a program of development for a Fifth Generation of computing systems, and the funding of the ICOT research center in Tokyo. As Mota-Oka, Chairman of the Managing Committee, notes: 'In these systems, 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.' [19]

Karatsu, Chairman of the Social Needs Committee, reiterates the HCI theme: 'Until today, man had to walk toward the machine with effort to fit precisely. On the contrary, tomorrow, the machine itself will come to follow the order issued by man.' [19].

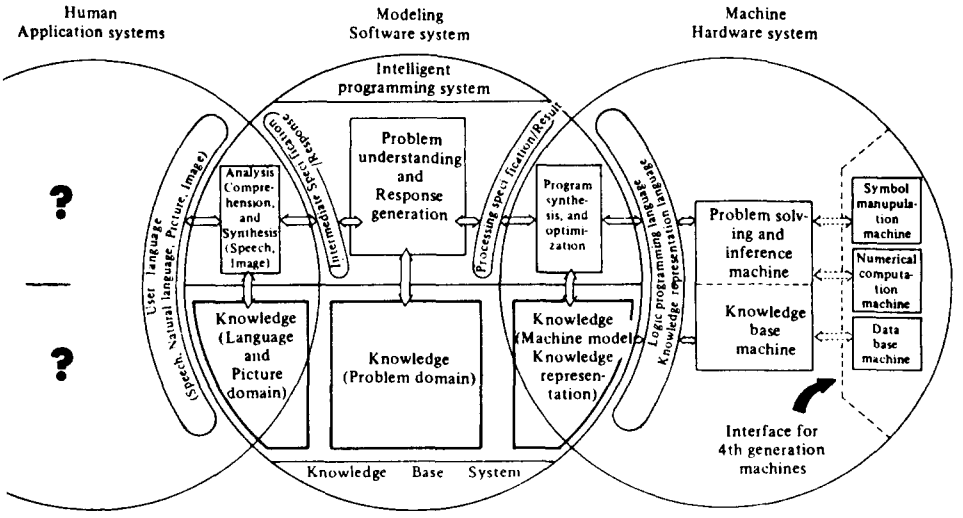
Fig. 1 shows the 'conceptual diagram' of fifth generation computers, and it is notable that the user interface is one of 'speech, natural language, pictures and images' to a system processing 'knowledge' rather than information. Fuchi, the Director of the ICOT Research Center, writes of *The Culture of the Fifth Generation Computer* [14], and we can see it as being user natural in its human factors. The next generation of computers will operate comfortably within the culture of person-person interaction; HCI will no longer be different.

We may see the Japanese proposal as a natural response to advances in computer technology that have given us massive power in hardware and software at low cost [7]. The technology which limited us for so long has now outstripped our demands and we can expect a shift from technology-push economics in computer systems to those of market-pull. The HCI is what the customer sees as a computer and is where the market requirements are being expressed. We will increasingly build systems top-down from user needs rather than bottom-up from technology availability.

However, my remarks above about our creative imaginations being well in advance of our scientific knowledge and skills applies with force to the fifth generation conceptual diagram. The hardware and software rings are full of structure which is expanded in detail in the accompanying text. The human ring is empty and the Japanese program has no activities designed to fill it. Logic tells us that the third ring should contain the psychological structure of the user but it does not yet tell us what this is.

Thus, this conference takes place at a historic time when the human-centered strategy of fifth generation computing has captured imaginations around the world. Hopefully, it may provide the focus for activities that will fill the third ring and give us foundations for the human role in next generation computer systems.

Fig. 1. Conceptual Diagram of Fifth Generation Computer Systems (from [19] p.29).



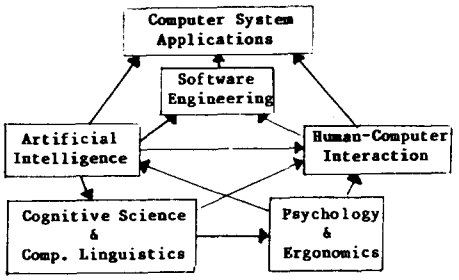


Fig. 2. Influences Between Disciplines

6. Artificial Intelligence, Psychology and HCI

There has long been a close, but ambivalent, relationship between studies of artificial intelligence (AI) and those of HCI. Their key joint roles in the fifth generation program makes it important too examine this relation and those to other areas such as cognitive science and software engineering. Abstract definitions are probably of less value than a simple example that exaggerates the differences in approach. Consider a problem of overcoming the barriers to database access by casual users:

- the AI approach to this problem might be to build a better database system LMwith understanding of the user, his requirements, and natural language PMcommunication, i.e. to put all the load on the computer and make it clever LMenough to deal with the casual user;
- the applied psychology approach might be to develop a training program for LMcasual users that gave them as rapidly and effectively as possible the PMskills to use the database, i.e. to put all the load on the person and make KMhim clever enough to cope with the computer;
- the HCI approach might be to determine where the problems lie in the RMinterface between user and computer and design a communication package that KMhelps him formulate his requests in a way natural to him but which can be MMtranslated easily into a form natural to the database, i.e. to remove as NMmuch of the load as possible from both systems and share what is left QMbetween them.

In practice, a pragmatic system designer will take what he can get from all three areas and put together a working system. However, the example shows, in very simplistic terms, both the close relationships between the disciplines and their differing orientations. Fig. 2 incorporates related disciplines into an influence diagram indicating the rich structure now underpinning person-computer applications.

AI and HCI are now major influences on computer applications and both have influenced software engineering: AI by the development of symbolic languages for knowledge processing and HCI by developments for dialog engineering. AI has also had a direct influence on HCI by the introduction of intelligent interfaces and natural language processing. It has also had an indirect influence through its promotion of interest in cognitive science and computational linguistics which provide tools for HCI. These areas have also become influential in psychology and ergonomics providing foundations for HCI. Completing the loop, psychology in its turn has influenced thinking in AI where the goal has been to emulate human intelligence rather than create truly *artificial* intelligence.

The diagram is over-simplified and cases can be made for counter-influences to those shown but it serves to show the rich environment for HCI studies, theory and applications that now exists. We are already at a stage where some of the wilder speculations of a few years ago [5] are becoming reality. Fifth generation systems, incorporating advances in all these areas, will provide us with a new medium for communication that subsumes those now available and expands our modes of existence [27].

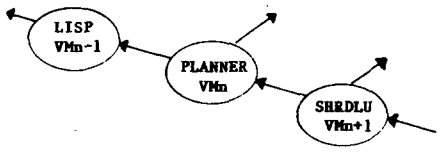


Fig. 3 Virtual Machine Hierarchy

7. The Virtual Machine Hierarchy and HCI

The concept of *virtuality* has proved an extremely useful one in computer science y9[35]; we can view any computer system as either a programmable machine with certain features together with a program, or as a (programmable) virtual machine (VM) with extended features. VM's form a natural hierarchy, for example the natural language interface of SHRDLU [37] is a program in PLANNER which is written in LISP (see Fig. 3).

The extra arrows in Fig. 3 indicate that PLANNER could have been written in some language other than LISP and SHRDLU could have been in something other than PLANNER. The de-coupling between levels of the VM hierarchy is important to a user who need be aware only of the level which he accesses. Nelson [20] has emphasized that the user sees the virtual machine not the underlying one, and Smith has noted this as a design principle of the Xerox Star: '...if everything in a computer system is visible on a display screen, the display becomes reality.' [32]

If the VM hierarchy is extended to show the people involved in its design, implementation and use, a surprisingly large number of HCI studies may be set in an integrated framework [6] (see Fig. 4).

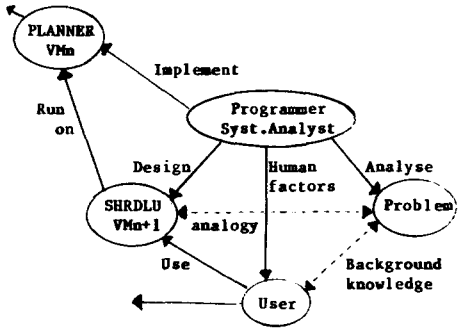


Fig. 4 Extended Virtual Machine Hierarchy

The user is now seen to be part of the VM hierarchy, perhaps a secretary creating a VM for an executive who wishes to send mail but cannot use a word processing VM.

The systems analyst & programmer team is seen as mediating between the problem and the two levels of the VM hierarchy, at one of which the solution is designed and at the other of which it is implemented.

The remaining lines indicate other relations: that the systems analyst has to take into account the human factors of the user; that he generally assumes the user to have some background in the problem; and that there is a relationship of analogy between the problem and the VM solving it [4].

In the context of expert systems the problem becomes that of mimicing the expert, analysis is knowledge engineering [28], and human factors are embedded in a dialog shell [9].

Most HCI studies can be set in this framework. The user may use other VM's and import expectations of these to his understanding of VMn. An analyst/programmer team can be examined taking into account the four relationships shown, and the expansion of these when the task requires design and implementation of multiple levels of the VM hierarchy. We can see that the technical problem of whether a VM is matched to a problem, e.g. whether Pascal or LISP is better for natural language processing, is confounded by many human variables: the programmer may know Pascal better than LISP; or he may know LISP well yet not understand the problem; or he may understand both but not take into account the user interface; and so on.

The concepts in this diagram can be formalized to any required degree within a category-theoretic framework [8], although our knowledge of the human components is only just beginning to justify such formality [28].

8. Conclusions

The past thirty years has seen a swing in human factors in HCI from lip-service and regrets that they are neglected to a wide range of in-depth studies both academic and commercial. It has also seen a swing from human factors based primarily on introspection to those based on experimental studies [6]: ethological where programmers and users are studied in their natural habitats; experimental where they are boxed in a laboratory micro-world and their behavior monitored; and statistical where the dynamics of user populations are analysed.

The growth in the literature, new journals and specialist conferences in the 1980's shows that these studies are beginning to bear fruit. The empty third ring in the Fifth Generation diagram indicates that we still have much to do and discover. Only when we know as much about the natural phenomena of people as we do about the artificial phenomena of hardware and software will human factors achieve its desired maturity. Perhaps the natural/artificial distinction will by then have disappeared as we merge with our artefacts into true man-machine symbiosis. There is a Chinese curse, 'May you live in exciting times' – we surely do.

References

1. Badre, A. and Shneiderman, B. (eds.), *Directions in Human Computer Interaction* (Ablex, New Jersey, 1982).
2. Coombs, M. J., and Alty, J. L. (eds.), *Computing Skills and the User Interface* (Academic Press, London, 1981).
3. Degano, P. and Sandewall, E. (eds.), *Integrated Interactive Computing Systems* (North-Holland, Amsterdam, 1983).
4. Gaines, B. R., Analogy categories, virtual machines and structured programming, in Goos, G. and Hartmanis, J. (eds.), *GI-5. Jahrestagung Lect. Notes in Comp. Sci.*, 34 (Springer, Berlin, 1975) 691-699.
5. Gaines, B. R., Man-Computer Communication - what next?, *International Journal of Man-Machine Studies*, 10(3) (1978) 225-232.
6. Gaines, B. R., The role of the behavioural sciences in programming, in ON*Structured Software Development*, Vol. 2 (Infotech International, Maidenhead, UK, 979) 57-68.
7. Gaines, B. R., A framework for the fifth generation, *Proceedings of National Computer Conference*, 53 (AFIPS Press, Arlington, Virginia, 1984).
8. Gaines, B. R. and Shaw, M. L. G., Dialog shell design, *Proceedings of INTERACT '84* (North-Holland, Amsterdam, 1984).
10. Grandjean, E. and Vigliani, E. (eds.), *Ergonomic Aspects of Visual Display Terminals* (Taylor & Francis, London, 1980).
11. Green, T. R., Payne, S. J. and van der Veer, G. C., (eds.), *The Psychology of Computer Use* (Academic Press, London, 1983).
12. Guedi, R. A., ten Hagen, P. J. W., Hopgood, F. R. A., Tucker, H. A. and Duce, D. A. (eds.), *Methodology of Interaction* (North-Holland, Amsterdam, 1980).
13. Hansen, W. J., User engineering principles for interactive systems, *Proceedings of the Fall Joint Computer Conference*, 39 (AFIPS Press, New Jersey, 1971) 523-532.
14. Hirose, K. and Fuchi, K., *The Culture of the Fifth Generation Computer* (Kaimeisha, Tokyo, 1984).
15. Martin, J., *Design of Man-Computer Dialogues* (Prentice-Hall, New Jersey, 1973).
16. Mauchly, J. W., Preparation of problems for EDVAC-type machines, in Randell, B. (ed.), *The Origins of Digital Computers: Selected Papers* (Springer-Verlag, Berlin, 1973) 365-369.
17. Merrifield, C. W., Report of a committee appointed to consider the advisability and to estimate the expense of constructing Mr. Babbage's Analytical Machine, in Randell, B. (ed.), *The Origins of Digital Computers: Selected Papers* (Springer, Berlin, 1973) 53-63.
18. Mills, R. C., Man-machine communication and problem solving, in Cuadra, C. KNA. (ed.), *Annual Reviews of Information Science and Technology, 2* (Interscience, NY, 1967).
19. Moto-oka, T. (ed.), *Fifth Generation Computer System* (North-Holland, Amsterdam, 1982).
20. Nelson, T., Interactive systems and the design of virtuality, *Creative Computing, 6(11)* (1980) 56-62.
21. Nickerson, R. S., Man-computer interaction: a challenge for human factors research, *IEEE Transactions on Man-Machine Systems, MMS-10(4)* (1969) 164-180.
22. Nickerson, R. S., Why interactive computer systems are sometimes not used by people who might benefit by them, *International Journal of Man-Machine Studies, 15(4)* (1981) 469-483.
23. Sackman, H., *Man-Computer Problem Solving* (Auerbach, Princeton, 1970).
24. Shackel, B., Ergonomics for a computer, *Design, 120* (1959) 36-39.
25. Shackel, B. (ed.), *Man/Computer Communication* (Infotech International, Maidenhead, UK, 1979).
26. Shackel, B. (ed.), *Man-Computer Interaction: Human Factors Aspects of Computers and People* (Sijthoff & Noordhoff, The Netherlands, 1981).
27. Shaw, M. L. G. and Gaines, B. R., Fifth generation computing as the next stage of a new medium, *Proceedings of National Computer Conference, 53* (AFIPS Press, Arlington, Virginia, 1984).
28. Shaw, M. L. G., Knowledge engineering for expert systems, *Proceedings of INTERACT '84* (North-Holland, Amsterdam, 1984).
29. Shneiderman, B., *Software Psychology* (Winthrop, Cambridge, Massachusetts, 1980).
30. Sime, M. E., and Coombs, M. J. (eds.), *Designing for Human-Computer Interaction* (Academic Press, London, 1983).
31. Sime, M. E., Green, T. R. G. and Guest, D. J., Psychological evaluation of two conditional constructions used in computer languages, *International Journal of Man-Machine Studies, 5 (1)* (1973) 105-113.
32. Smith, D. C., Visibility in the Star user interface, *Proceedings of Graphics Interface '83* (National Research Council of Canada, Ottawa, NN1983) 181.
34. Wasserman, T., The design of idiot-proof interactive systems, *Proceedings of the National Computer Conference, 42* (AFIPS Press, New Jersey, 1973) 34-M38.
35. Weegenaar, H. J., Virtuality and other things like that, *Proceedings of IEEE COMCON, CHI388-8/78* (1978) 287-293.
36. Weinberg, G. M., *The Psychology of Computer Programming* (Van Nostrand Reinhold, New York, 1971).
37. Winograd, T., *Understanding Natural Language* (Edinburgh University Press, 1972).