INTELLIGENT USER INTERFACES

William E. Hefley
Department of Engineering and Public Policy
Carnegie Mellon University
Pittsburgh, PA 15213, U.S.A.
wh0i@andrew.cmu.edu

Dianne Murray
Social and Computer Sciences Research Group
Department of Sociology
University of Surrey
Guildford. GU2 5XH. U.K.
dianne@soc.surrey.ac.uk

ABSTRACT
We consider the significant work in both the human-computer interaction and artificial intelligence fields in the area of intelligent interfaces, placing into a structure the disparate strands of research work. We treat design and implementation issues, covering the relationship between user interface management systems (UIMS) architectures and interface design, and providing schematic architectures for building intelligent systems. We discuss such work in its historical context and suggest that UIMS architectures be used as a basis for constructing intelligent interfaces.

KEYWORDS:
Intelligent Interfaces, Intelligent User Interfaces, User Models, User Interface Management Systems, Adaptive Intelligent Interface Systems

INTRODUCTION

Significant work has been underway in both the human-computer interaction and artificial intelligence fields in the area of intelligent interfaces [36]. Much of this work is concentrated on dialogue understanding and on user modelling to support user assistance, providing usable presentations, error remediation or tutoring. Other work focuses on adaptive or adaptable interfaces. We provide here a working definition of Intelligent Interfaces, presenting a taxonomy of intelligent systems and showing how user modelling techniques for intelligent interface design is the foundation for a variety of Intelligent Systems and contributes to building specific application interfaces.

Development of systems which effectively make use of these forms of knowledge must address these issues: how do we *model* the user/operator in complex time-critical environments; how can this knowledge be brought to bear in the design of interfaces to enhance human-computer interaction and overall system effectiveness? There are many difficulties and problems in dynamic modelling of user intentions and behaviour: how to obtain information on the user, strategies for representing such information in an appropriate form, and mechanisms for using information in effective dialogues with users. There are also many more general issues in modelling aspects of user behaviour, including those of privacy and the ethics of maintaining user information but space does not allow discussion of these important aspects.

We hope that this will place into a structure and hierarchy the disparate strands of research work in intelligent user interfaces, all of which provide computer-based intervention in order to mediate user tasks and interactions. We treat design and implementation issues, covering the relation between UIMS architectures and interface design, and providing schematic architectures for building Intelligent Systems. We site this in the historical context of user interface management systems architectures and use UIMS concepts as a basis for intelligent interfaces research.

A HISTORICAL PERSPECTIVE

In many complex systems, the interactive component with which an operator interacts has now become an agent to control the system, regardless of whether that system is a supervisory control system, such as nuclear power plant, or a management information system, or a CAD/CAE system. Such systems demand that operators perform in two capacities: as a supervisor under normal operations and as a manual controller during emergency or degraded operations [18]. These capacities require operators to have both a high level integrated overview of the system and a more detailed perspective of its functions.

The principal user activity in complex systems is operating system controls based on remembered procedures, whilst concurrently monitoring system state and consequences obtained from the system or other operators [31]. These activities are mediated by efforts to recall or update known procedures, to ensure that correct controls are being operated, or to carry out explicit monitoring strategies. As in many systems, performance range can be from expert to novice and it is in these complex systems that the processes of human-computer interaction are crucial to successful performance.

Research work in human-computer interaction has made it clear that interaction with computers or complex machines does not take place at just the physical level but in a very much broader environmental context. Figure 1, based upon work by Rasmussen (25), shows five broad levels of interaction which can be distinguished in a wide range of human-computer systems.

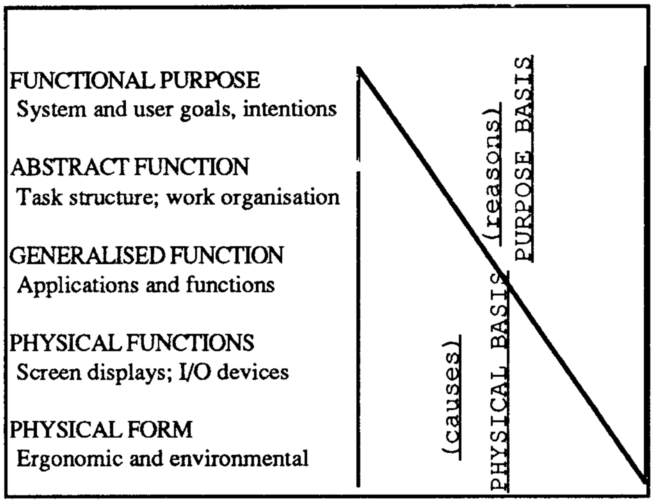

Figure 1. Levels of Abstraction (Rasmussen, 1987b)

Several perspectives for describing these processes have emerged and the perspective we adopt is to view interaction between human operators or users and computer-based systems as a cooperative endeavour which occurs because of a purpose or intention on the part of the human. The human-computer dialogue is a communication exchange between the human and the computer and can be viewed as *situated action*, wherein actions are situated in particular social and physical circumstances [35] within the context of specific task-oriented situations. The dialogue can encompass many interaction aspects - how information is requested by the user, just when it is required, and how it is presented. Further issues are the way in which request and presentation displays are controlled by the application system and how the style of dialogue between user and machine is chosen and managed. This latter aspect can also include interpretation of user errors, modifications and excursions and in many such situations, human error can be viewed as operator-task-system mismatch [24].

To reduce the complexity of a user's role in complex tasks, two approaches may be taken: to increase user skill or to reduce task complexity. Not only is preventing errors in high-performance human-system interfaces critical, but providing means to improve an operator's skills is the key to solving the dilemma of providing users with appropriate guidance in ways which will enable them to activate appropriate mental schemas for initiating effective processes. User guidance can play a crucial role in clarifying dialogues when user errors occur [9] and, as Smith & Mosier [33] point out, can serve not only as means of enhancing the user's current knowledge, but also as a learning mechanism. Sheridan [32] also suggests that controlled active participation in error recovery can indeed be a form of learning for the operator. Hefley [12] suggests that this is true not only in error recovery, but also in Level 3 (unfamiliar) and Level 4 (unforeseen) situations.

We believe that it is essential to develop computer systems to accommodate to the changing skill basis, experience and needs of different users or operators and to define the requirements of individual users. It is also evident that different groupings of users interact in varied fashions with computer systems and often require different interface modalities to make one approach more suited to them than another. Users are often classified as expert or novices, frequent or infrequent users. Despite the fact that it is difficult to develop eminently suitable strategies for dealing with even a small part of the whole complexity of human-computer communication, we believe that the endeavour of intelligent interfaces is moving toward this goal with some degree of success, as the papers in these proceedings demonstrate. There are three potential roles for the user interface: to assist in the correct and effective use of the system's capabilities, to be proactive in the user's problem solving process, and to provide training [1]. We attempt to define the structures, meanings, and situations that form an adaptive, intelligent user interface in order to better explore intelligent interaction between human and machine.

UNDERLYING ARCHITECTURES

There are a number of ways in which we might approach a classification of intelligent interfaces. However, a primary foundation is an underlying architecture, grounded in the design and implementation of the user interface in user interface management systems (UIMS) as shown by the influential Seeheim model [23] and the more recent Slinky metamodel [10]. A review of architectures used in UIMS, decision support systems, and knowledge-based systems provides an understanding of the components of many current systems. Future architectures must build on the strengths of these approaches and incorporate suitable extensions in order to become adaptive intelligent systems: knowledge-based systems which reason about and interact with other dynamic agents, including their users, in real time. We will discuss some aspects of underlying architecture and then try to illuminate various strands in intelligent interfaces work before bringing them together in a comprehensive overall architecture.

User Interface Management Systems

UIMS are high-level interactive software applications intended to design, specify, prototype, execute, evaluate, and maintain end-user interfaces [11]. Their area of application is in assisting the computer-based system to communicate with the user but a less visible focus, though no less important aspect from the user's perspective, is to provide separation of functionality between the application and user interface components of a system. An abstract, logical model of a UIMS, widely known as the Seeheim model [23], describes an architecture in which the user conducts a dialogue with the user interface (Figure 2). The UIMS is a software module having three logical components between the user and the application program. Green [10] describes each of these logical components and their functions as:

- **presentation:** to provide the presentation of the interface to the user
- **dialogue control:** to control of the structure (or sequencing) of the dialogue between the user and the application program
- **application interface model:** to define the semantics of the functionality of the underlying application

UIMS architectures encourage such separation and support the incorporation of new input/output technologies within the presentation layer. It is also possible to extend such component isolation to incorporate agents of varying types, whether these be application demons, interface filters, modelling components or intelligent self-actuating agents.

Interactions with UIMS

We believe that an initial architecture for intelligent systems can be found in the 3-way model of human-computer interaction (Figure 4) as originally described by Card (6). Newer notions to be incorporated are those of a metamodel for prescriptive design of interactive systems and the Arch model of the runtime architecture based on both the data exchanged and the internal functions of the system (Figure 3) [37]. This latter satisfies the condition of 'buffering' from the effects of changes but the overall purposes of a metamodel are twofold: to derive architectures, and to evaluate such architectures in terms of proposed goals. The functionality deemed important is shown in the operations necessary for interactive systems:

- control and re-organisation of domain data
- performing domain tasks
- provision of task sequencing
- provision of multiple view consistency
- decisions on appropriate media
- choice of interaction objects
- provision of physical interaction with the user
- conversion between domain formalisms and user interface formalisms

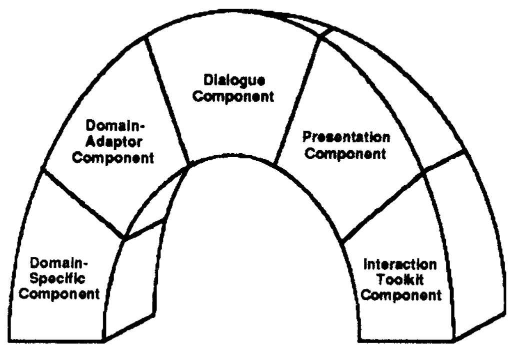

Figure 3. The ARCH model: interfaces between components

Functions which may be added to this for intelligent interfaces are those concerned with intelligent presentation, assistance, and higher-level management of the user's interaction, all accessing user and knowledge data bases. Linkages evident between Card's original model (and some supervisory control models [31]) are those of *tasks* whilst linkages between the metamodel and intelligent systems are those *domains* [37] and the presentation component. We posit that intelligent interfaces are really an instance of domain adaptors and that the domain specific component can be easily seen as the site of the user model data and the home of both domain-specific knowledge and dynamic knowledge-bases. The embedded user model can be linked to the domain adaptor component to effect the upward layer of the dialogue component. Other layers closer to the user interface itself can perform recognition (e.g., bridging between user actions and discourse, and the goals/intentions

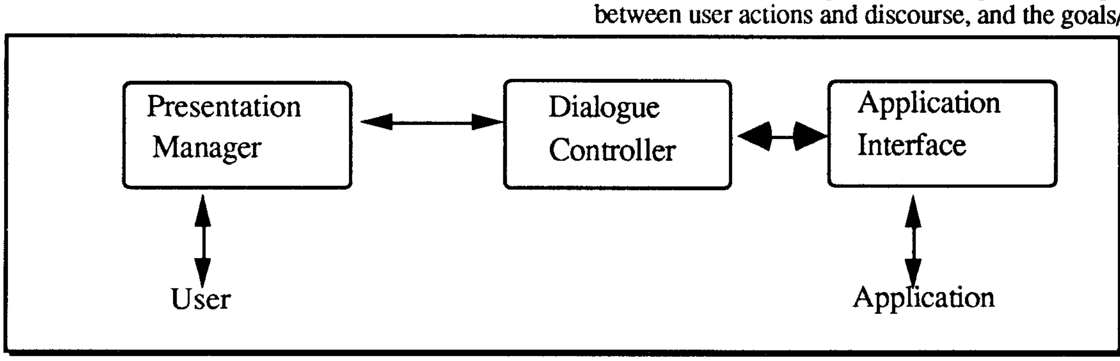

Figure 2. Components of the Seeheim model for UIMS

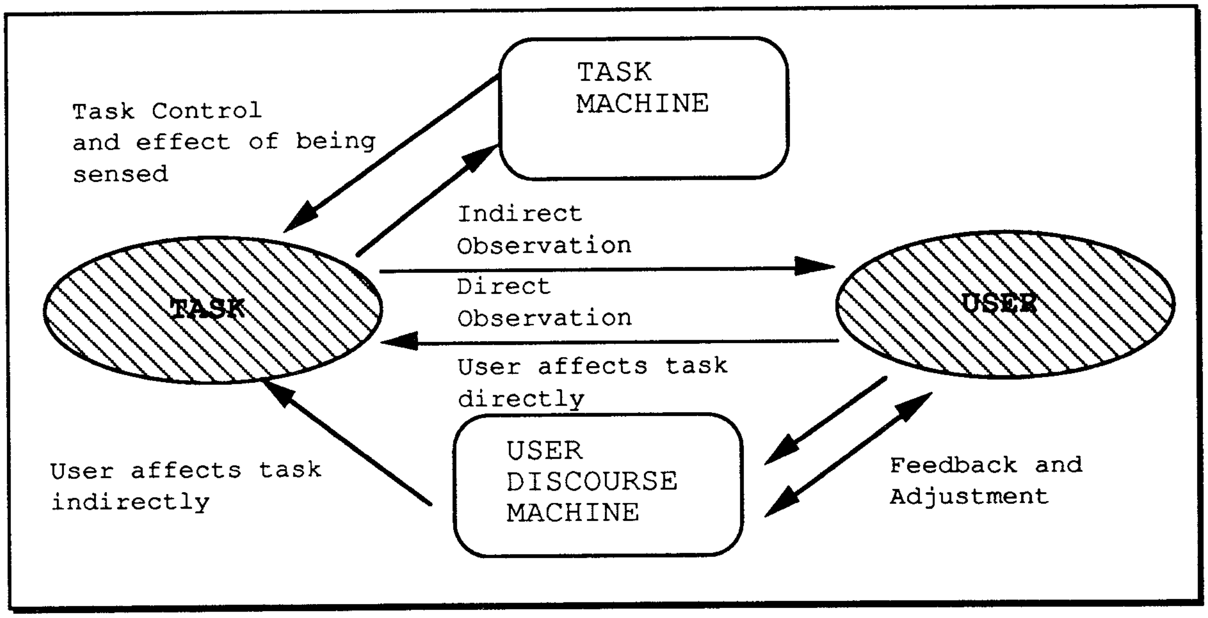

Figure 4. User Discourse Machine

of users), presentation, explanation, user tailoring and so on. We suggest that a robust intelligent interface will have several embedded layers within it, in addition to the provision of adaptivity and intelligent presentation. These include specific levels for:

- **Domain adaptation:** goal management, higher level (i.e., cognitive) user modelling
- **Discourse management:** management of the interaction, action/presentation-level user modelling

CLASSIFICATION

A potential classification is by domain area whilst another, which may be more fruitful, is by a subdivision into wider classification of knowledge based systems, decision support systems and adaptive systems.

Knowledge-based systems

Knowledge-based systems are employed in domains where decision-makers must repeatedly make decisions about which they have insufficient or incomplete knowledge. Both expert systems, which utilise a knowledge base of heuristics used by experts, and knowledge-based systems, which use an augmented knowledge base and additional algorithms and calculation procedures, are typified by several system components:

- an inference engine for rule processing, which may include a scheduler (to control the order of rule processing) and an interpreter (for applying rules)
- a consistency enforcer to adjust previous conclusions when new data or knowledge alters the bases of support for current conclusions
- a knowledge base comprising facts and heuristic rules
- a data store for recording intermediate results
- a justifier that rationalises and explains the system's behaviour

The interfaces which support such systems are typically a user interface/language processor (to enable communication between the user and the system); a knowledge acquisition interface (to assist in building the knowledge base), and external interfaces to the external world.

A pertinent application of knowledge-based systems is in intelligent computer-aided instruction (ICAI) or intelligent tutoring systems (ITS) which actively diagnose the state of a student's knowledge, providing motivation to increase that knowledge through tutorial strategies. In a well-established field, there are three distinct, but related, instructional processes which are accomplished through four components: a domain knowledge base; student model (state of the student with respect to domain knowledge); tutorial or instructional modules, and a user interface. An important feature of ITS is that of student modelling. In fact, the field of user modelling may stake a strong claim to have its roots in the mechanisms and representational strategies of modelling student knowledge.

Recent advances in distributed artificial intelligence (DAI) [14] incorporate the perspective of computational actors or agents [7, 16] which are processes encapsulating a dynamic representation of the system and its situations, perhaps including a self-representation or a representation of internal goals. Young [39] views expert systems as agents with embedded knowledge of their domains and communication with their users.

Decision support systems

Decision support systems (DSS) have been employed in situations which are ill-structured and complex, but which require a decision-maker possessing some expertise to make decisions. The computer-based DSS supports the user's intuition or assists in solving a problem whose solution typically contributes to a decision which must be ultimately made by the user.

Sprague & Carlson [34] present a model of DSS use which is influenced by both task and environment. They describe three components of a DSS: dialogue management, model base management system, and database management system. The dialogue management component supports all communication between the user and the DSS. The DSS implements a set of models within the model base management system which address the expected range of situations and draws upon the database management system for data manipulation capabilities.

Evolutionary perspectives on DSS suggest the extension of DSS models with text base and rule base components [2]. Other recent efforts suggest that DSS must provide a framework for facing new problems or situations which are different from those expected [17].

Adaptive Intelligent Systems

Whilst an intelligent system builds on facts and the heuristic knowledge of an expert, together with techniques for reasoning about unstructured situations, an adaptive intelligent system has the added capability to learn over time from experience. An intelligent system must focus resources on the specific problem being addressed but must also be data-driven, constantly alert to unforeseen events. It has been suggested that such a system must also be self-aware: achieved through monitoring its own actions, knowledge, needs, goals and intentions.

This section presents a functional architecture (based in part on Hefley [12] and Murray [21]), relying on user modelling, for an adaptive intelligent system oriented towards providing a collection of agents which monitor and mediate the interactions between the user and the underlying application. The proposed architecture draws heavily on the UIMS models, Sprague & Carlson's DSS architecture and Card's 3-way model of human-computer interaction. As in UIMS, the architecture separates the user interface from the application, while implementing an enhanced task model in the form of an intelligent decision support system. The task model is the system's model of the user, containing a description of the tasks the user performs with the user interface and providing a basis for prediction and monitoring of users actions.

The user model is a model of an individual user's behaviour, responses and intentions to provide data for predictive and evaluative functions. In order to adapt at an effective level, this model should contain data on personal features as demonstrated by both overt and implied behaviour. Further user details can be inferred from usage patterns and logged sequences or could be explicitly provided by an external source. This model [19, 20, 22] may be of a similar construction to the inference engine of an expert system, encompassing production rules and an evaluative function or may be separated from the remainder of the system in a functionally organised fashion. By continually monitoring the interaction, making predictions of users' intentions and comparing actual performance against what is known about the user's skills and natural preferences, adaptive interactions maybe effected.

Many of the concepts described here can be found in isolated instances throughout the literature. Figure 5 shows an architecture of an adaptive intelligent system which attempts to integrate concepts of adaptation—to the domain and to the user. Previous research into problem-solving capabilities embedded within user interfaces has identified several important issues: information display, task specific natural language interface, explanation, discourse modelling, planning, and personalisation [8]. Rouse and his colleagues have identified architectural issues and presented conceptual designs for adaptive user interfaces [26, 27, 28, 29]. Williges, Williges, and Elkerton [38] distinguish between flexible interfaces, where the user can tailor the interface or the interface supports several interaction styles, and adaptive interfaces, which adapt over time to accommodate the user and their interaction.

CONCLUSION

The areas of application of such an architecture are many and all fall within of intelligent user interfaces research. Whilst many architectures and schemas for the various system exist, we hope that this integrated model will provide a basis for looking both at the functionality of the systems which we can expect to see in future years and to assist in construction, in a separable fashion, of realistic systems which build upon existing technology, paradigms and research from many contributing disciplines.

We believe that the model given will allow construction of:

- **Intelligent Front Ends (IFE):**

  expanding on early work in building intelligent interfaces to existing database, knowledge based and expert systems.

- **Natural Language (NL) interfaces:**

  emphasising those which have developed user modelling techniques of interest to interface designers.

- **Intelligent Tutoring Systems (ITS):**

  emphasising those which have relevance to user support, rather than detailed consideration of the merits of pedagogical or student knowledge modelling strategies.

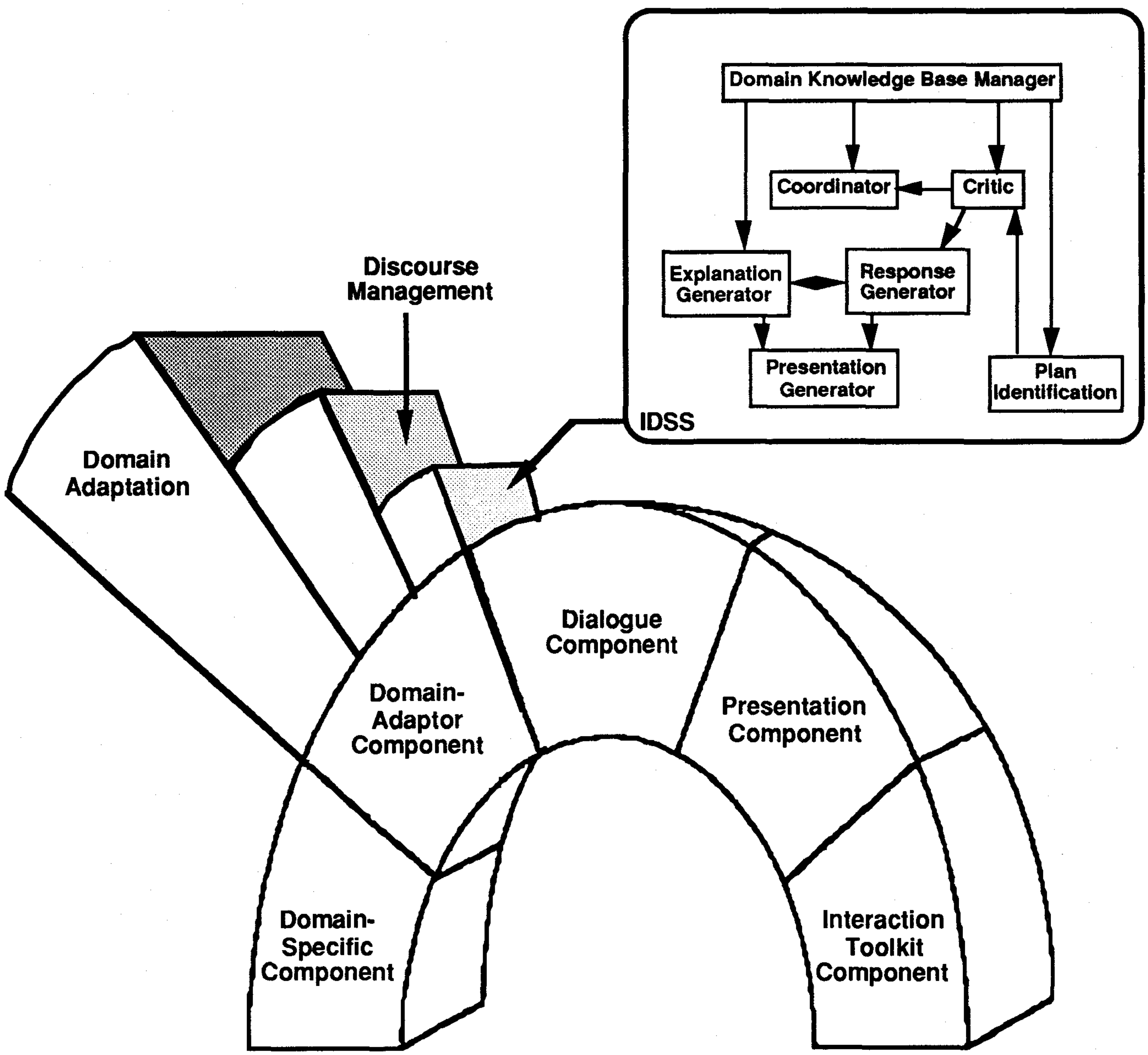

Figure 5. Architectural model for an adaptive intelligent human-machine interface

- **Intelligent Help and Support Systems (IHSS):**

  distinguishing between 'assistance' and 'tutoring' in advisory systems and focusing on NL interfaces to explanation systems which can act as computer critics, coaches, and mentors.

- **Intelligent Multimedia Presentation Systems (IMMPS):**

  designing presentations which express information using a combination of the available presentation techniques and media in a way which achieves the communicative purposes and supports users in performing their tasks [30].

- **Decision Support Systems (DSS):**

  emphasising user support through interface adaptability (information aiding) and presenting intelligent decision support (integration aiding). Intelligent decision support systems (IDSS) are being explored as a possible means of improving human-machine systems in complex environments [13].

- **Adaptive Interfaces:**

  focusing on adaptation and self-adaptive systems rather than exclusively on interface customisability and statistically reactive systems [3, 4, 5, 15].

- **Cooperative Intelligent Agents and Dialogue Assistants:**

  expanding dialogue through agent-based interaction incorporating knowledge representation and inferencing strategies.

All of these systems are described in the contents of the papers selected for this conference. We hope that a greater focus on the practicalities of building such systems, of evaluating their usefulness and determining just when intelligent interfaces are needed will expand this burgeoning field and lead to a greater understanding of the elements contributed by both human and machine partners in our human-computer dialogues.

REFERENCES

1. Alty, J.L., & Guida, G. (1986) The use of rule-based system technology for the design of man-machine systems. In G. Mancini, G. Johannsen, & L. Martensson (Eds.), *Analysis, Design and Evaluation of Man-Machine Systems* (pp. 21-41) Oxford: Pergamon Press.

2. Belew, R. K. (1985) Evolutionary decision support systems: an architecture based on information structure. in L. B. Methlie, & R. H. Sprague (Eds.), *Knowledge Representation for Decision Support Systems* (pp. 147-160) Amsterdam: North-Holland.

3. Benyon, D.R., Innocent, P.R., Murray, D.M. (1987) System adaptivity and the modelling of stereotypes. In: B. Shackel and H-J. Bullinger (Eds.), Proc. INTERACT '87, Second IFIP Conference on Human-Computer Interaction. Amsterdam: Elsevier Science Publishers.

4. Benyon, D.R., Milan, S. and Murray, D.M. (1987) Modelling users' cognitive abilities in an adaptive system, In: J. Rasmussen and P. Zunde (Eds.), Proc. 5th Symposium EFISS, Risø National Laboratory, Denmark, November 1987. New York: Plenum Publishing.

5. Benyon, D.R. and Murray, D.M. (1988) Experience with adaptive interfaces, The Computer Journal, 31(5).

6. Card, S.K., (1984) Human factors and the intelligent interface. In: G. Kohl, G. and S.S. Nassau (Eds.), Human Factors, Proc. Symposium sponsored by the Metropolitan Chapter of the Human Factors Society, New York, November 15th, 1984.

7. Doran, J. (1989) Distributed artificial intelligence and the modelling of socio-cultural systems. In L. A. Murray, & J. T. E. Richardson (Eds.), *Intelligent Systems in a Human Context* (pp. 71-91) Oxford: Oxford University Press.

8. Fox, M. S. (1983) The intelligent management system: an overview. In H. G. Sol (Ed.),*Processes and Tools for Decision Support.* (pp. 105-130) Amsterdam: North-Holland.

9. Gilmore, W. E., Gertman, D. I., & Blackman, H. S. (1989) *The User-Computer Interface in Process Control.* Boston: Academic Press.

10. Green, M. (1985b) Report on dialogue specification tools. In G. E. Pfaff (Ed.), *User Interface Management Systems.* (pp. 9-20) Berlin: Springer-Verlag.

11. Hartson, H. R., & Hix, D. (March, 1989) Human-computer interface development: concepts and systems for its management. ACM Computing Surveys, 21(1), 5-92.

12. Hefley, W. E. (1990) Architectures for adaptable human-machine interfaces. In W. Karwowski & M. Rahimi (Eds.), Ergonomics of Hybrid Automation Systems II (Proceedings of the Second International Conference on Human Aspects of Advanced Manufacturing and Hybrid Automation) (pp. 575-585) Amsterdam: Elsevier Science Publishers.

13. Hollnagel, E., Mancini, G., & Woods, D. D. (1986) *Intelligent Decision Support in Process Environments.* Berlin: Springer-Verlag.

14. Huhns, M. N. (Ed.)(1987) *Distributed Artificial Intelligence* London: Pitman.

15 Jennings, F., Benyon, D.R. and Murray, D.M (1991) Adapting systems to individual differences in cognitive style. Acta Psychologica, December.

16. MacLean, R. (1989) Enaction formalisms. C. Tully (Ed.), *Proc. 4th International Software Process Workshop.* Moretonhampstead, Devon, UK. (pp. 11-13) New York: ACM Press.

17. Migliarese, P. G.(1985) Knowledge representation and organizational noise: an organizational framework about decision support systems. In L. B. Methlie, & R. H. Sprague (Eds.), *Knowledge Representation for Decision Support Systems.* (pp. 237-244) Amsterdam: North-Holland.

18. Mitchell, C. M., Stewart, L. J., Bocast, A. K., & Murphy, E. D. (1982) *Human Factors Aspects of Control Room Design: Guidelines and Annotated Bibliography* NASA TM 84942. Greenbelt, MD: Goddard Space Flight Center.

19. Murray, D.M. (1987) Embedded user models. In B. Shackel and H-J. Bullinger (Eds.), Proc. INTERACT '87, Second IFIP Conference on Human-Computer Interaction. Amsterdam: Elsevier Science Publishers

20. Murray, D.M. (1987) A survey of user modelling definitions and techniques, NPL DITC Report 92/87. Teddington, U.K.

21. Murray, D.M. (1988) Building a user modelling shell. In P. Zunde (Ed.), Proc. 6th Symposium EFISS, Georgia Tech., Atlanta, Georgia, USA, October 1988. New York: Plenum Publishing.

22. Murray, D.M. (1989) Modelling for adaptivity, Proceedings of 8th Interdisciplinary Workshop, Informatics and Psychology, Schärding, Austria, May 1989 Amsterdam: North Holland.

23. Pfaff, G. E. (Ed.) (1985) *User Interface Management Systems.* Berlin: Springer-Verlag.

24. Rasmussen, J. (1987a) The definition of human error and a taxonomy for technical system design. In J. Rasmussen, K. Duncan, & J. Leplat (Eds.), *New Technology and Human Error* (pp. 23-30) Chichester, UK: Wiley.

25. Rasmussen, J. (1987b) Mental models and their implications for design. In Proc. 6th Workshop on Informatics and Psychology, June 1987 (Austrian Computer Society).

26. Rouse, W. B., & Rouse, S. H. (1983) *A Framework for Research on Adaptive Decision Aids* Technical Report AFAMRL-TR-83-082 Wright-Patterson AFB, OH: Air Force Aerospace Medical Research Laboratory.

27. Rouse, W. B., & Morris, N. M. (1986) Conceptual design of a human error tolerant interface for complex engineering systems. In G. Mancini, G. Johannsen, & L. Martensson (Eds.), *Analysis, Design and Evaluation of Man-Machine Systems.* (pp. 281-286) Oxford: Pergamon Press.

28. Rouse, W. B., Geddes, N. D., & Curry, R. E. (1987-1988) An architecture for intelligent interfaces: outline of an approach to supporting operators of complex systems. Human-Computer Interaction, 3 (2), 87-122.

29. Rouse, W. B. (1988) Adaptive aiding for human/computer control. Human Factors, 30(4),431-443.

30. Roth, S. R. and Hefley, W. E. (in press) Intelligent multimedia presentation systems: research and principles. In M. Maybury (Ed.), *Intelligent Multimedia Interfaces* Cambridge, MA: AAAI Press.

31. Sheridan, T. B. (1981) Understanding human error and aiding human diagnostic behaviour in nuclear power plants. In J. Rasmussen, & W. B. Rouse (Eds.), *Human Detection and Diagnosis of System Failures.* (pp. 19-35) New York: Plenum Press.

32. Sheridan, T. B. (1988) Human and computer roles in supervisory control and telerobotics: musings about function, language and hierarchy. In L. P. Goodstein, H. B. Andersen, & S. E. Olsen (Eds.), *Tasks, Errors and Mental Models.* (pp. 149-160) London: Taylor & Francis.

33. Smith, S. L., & Mosier, J. N. (August, 1986) *Guidelines for Designing User Interface Software* Technical Report MTR-10090, ESD-TR-86-278 Bedford, MA: The MITRE Corporation.

34. Sprague, R. H. J., & Carlson, E. D. (1982) *Building Effective Decision Support Systems* Englewood Cliffs, N.J: Prentice Hall.

35. Suchman, L. A. (1987) *Plans and Situated Actions* Cambridge: Cambridge University Press.

36. Sullivan, J. W. & Tyler, S. W. (eds.). (1991). Intelligent User Interfaces. Reading, Mass : Addison-Wesley.

37. The UIMS Tool Developers Workshop. (1992) A metamodel for the runtime architecture of an interactive system. ACM SIGCHI Bulletin, 24(1), 32-37.

38. Williges, R. C., Williges, B. H., & Elkerton, J. (1987) Software interface design. In G. Salvendy (Ed.), *Handbook of Human Factors.* (pp. 1416-1449) New York: Wiley.

39. Young, R. M. (1989) Human interface aspects of expert systems. In L. A. Murray, & J. T. E. Richardson (Eds.), *Intelligent Systems in a Human Context.* (pp. 20-34) Oxford: Oxford University Press.