The ICADS Model in Retrospect

Jens Pohl, Leonard Myers, and Arthur Chapman

CAD Research Unit
Design and Construction Institute
School of Architecture and Environmental Design
Cal Poly, San Luis Obispo, California, USA

Abstract

This paper reviews work, involving the development and implementation of a prototype working model of an intelligent computer-aided design system (ICADS), conducted at the CAD Research Unit over the past four years. In the ICADS model, images drawn by the architect are analysed in background to establish higher level architectural objects, such as spaces, windows, doors and furniture. Combined with non-geometric attributes obtained from prototypical building type and contextual site/neighborhood knowledge bases, these design objects serve as a high level representation of the current state of the design solution.

Domain experts, functioning as intelligent design tools, continuously evaluate the current state of the progressively evolving solution model to test solution validity, confirm design program compliance and propose alternative solution strategies. Conflicts among the domain experts are resolved within a blackboard-like coordination and control system. The domain experts and the blackboard, together constituting an Expert Design Advisor, are implemented in a production rule environment utilizing a frame-based representation structure.

Key Words

artificial intelligence, blackboard, computer-aided design, cooperating agents, expert system, knowledge base

Background

Over the past several years the CAD Research Unit has gained some modest insights into the characteristics of the design activity and, in particular, the manner in which this activity might be supported in a computer-based environment. In some cases, what initially appeared to be relatively straightforward design tasks that were thought to be readily supportable on a computer, turned out to be much more complex. For example, the development of rules that arbitrate among several conflicting solution proposals was found to be heavily influenced by design context assumptions. These assumptions are easily overlooked and, if known, they are often difficult to represent appropriately in a computer-based environment. In other cases, what appeared at first sight to be potentially unmanageable was sometimes found to be surprisingly simple.

We found, for example, that computer-based domain experts are able to make meaningful evaluations of meagerly defined early design proposals, as long as the design proposals are represented at a high level of abstraction (i.e., as real world objects).

Some five years ago, when two of the authors first formulated their concepts of a computer-aided architectural design system, the design professions at large were generally well satisfied with the first wave of drawing automation systems. These early CAD systems focused almost entirely on the production aspects of design practice. The notion that the computer might be capable of supporting the human designer in the reasoning process that essentially underlies the artifacts that are represented in drawings, was met with a great deal of scepticism.
Several reasons may be cited for this genuine disbelief that a CAD system could be any more than a drawing production automator. First, design practitioners by and large were not willing to recognize design decision making as a distinct activity apart from the communication of these decisions by means of drawings. Over the centuries manual design practice has become synonymous with sketching and drawing. Even today designers, whether they be architects or engineers, tend to place primary importance on the images that they produce. When asked to describe what they do, they invariably argue that the visualization of the evolving design solution is in fact the kernel of the design activity.

Second, the cognitive functions that underlie the design reasoning process are extremely complex and, as yet, poorly understood. What is not well understood cannot be clearly described and is therefore typically relegated to less importance in a society dominated by the rationalistic approach (Kuhn 1962).

Third, the general unwillingness of design practitioners to recognize the fundamental distinction between the mental processes and the communication processes in the design activity was greatly reinforced by the failings of the design methods movement in the 1970s. It became clear in the 1980s that the operations research approach to problem solving had little relevance to the realm of design problems, where both the problem specifications and the solution objectives change dynamically during the decision making process (Rittel and Webber 1984, Simon 1984). Although the advocates of the operations research approach did not claim that their methods could provide a comprehensive solution to design problems, the profession nevertheless expected a simple and usable theory that could be readily applied in practice. When such a theory did not emerge the architect, in particular, decided that the design activity would for the foreseeable future rightly remain the province of the human designer.

Finally, early CAD drawing systems did not include any of the prerequisistes of a reasoning environment. They were single process, sequential systems that lacked linkages to external information resources and included only the most primitive object representations. Typically, a fairly advanced design solution was entered by the user in almost entirely geometric terms. In most cases the drawing was represented inside the system in terms of point and line objects. Later, solid modelling systems became commercially available with surface and shape objects, such as spheres, cones and slabs. However, even these more sophisticated object representations were and continue to be far removed from the real world objects such as for example site, building, space, wall, and opening that the architect reasons about.

The Development of the ICADS Model

Within this environment, in 1986, the CAD Research Unit at Cal Poly assembled an interdisciplinary team of architects, engineers and computer scientists to develop a prototype working model of a computer-aided architectural design system that could serve as a testbed for exploring the potential of computer-designer partnerships. With initial funding from IBM Corporation (IBM/GPD, San Jose, California), by the end of 1988 the CAD Research Unit team had developed a description of the design activity and established an implementation strategy for a working model of an intelligent computer-aided design system, to become known as the ICADS model (Pohl et al. 1988a, 1988b).

The ICADS implementation strategy targeted three areas for elevating a computer-aided drawing environment to a computer-aided design environment: representation; information; and, intelligent assistance. With the mandate to reach the goal of a working prototype as soon as possible, it was decided to utilize off-the-shelf software systems where feasible. Accugraph Corporation (El Paso, Texas) became an early partner in the ICADS project by making their MountainTop CAD package available as the CAD drawing environment of the ICADS model. IBM Corporation contributed several networked RT workstations, systems software (ie., AIX (Unix), X-Window and the C language) and the SQL/RT (Oracle) relational database system. In addition, the CAD Research Unit adopted the CLIPS expert system shell, developed by the Software Technology Branch of the Johnson Space Center at NASA, as the principal knowledge engineering tool (NASA 1989).

Development of the first ICADS working model (DEMO1) commenced in early 1989. Conceived as a CAD system shell (Fig.1), integrating a CAD drawing environment with internal and external information resources and an expert design advisory facility (Fig.2), the main development tasks centered on the interfaces that would allow these components to interact with each other and the human designer in a highly cooperative environment.

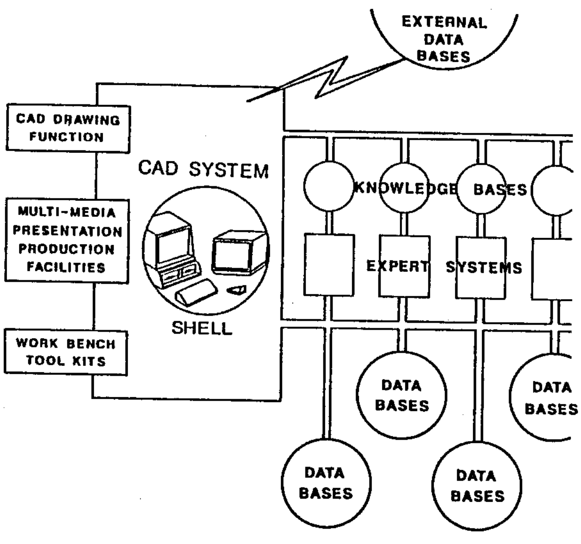

Fig.1: A CAD System Shell

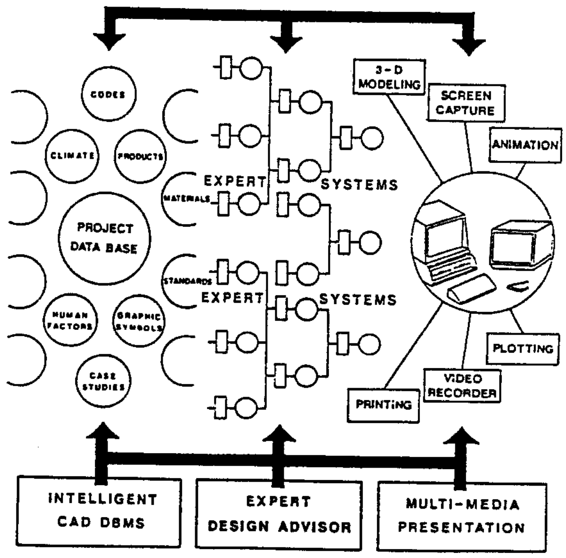

Fig.2: The Conceptual ICADS Model

With the adoption of the MountainTop 2-D CAD drafting package, a Geometry Interpreter was developed to map the point/line geometric representations of the drawing environment to the real world objects (e.g., neighborhood, site, building, floor, space, wall, window, and door) that the architect reasons about. The 2-D limitation, which made the geometric database interpretation approach feasible, allowed the CAD Research Unit to satisfy the representation requirement (one of the three target areas) fairly quickly. The other two target areas, information and intelligent assistance, necessitated a more substantial research and development effort.

The CAD system shell concept shown in Fig.1 clearly indicates the dominance of data and knowledge bases within the proposed computer-aided design environment. It suggests that a computer-based design system must by necessity be a knowledge-based system. Reasoning is an information intensive process that develops solutions based, at least partly, on knowledge of past similar problems applied to the context of the current design space (Gero et al. 1988, Oxman and Gero 1988, Oxman 1990, Gero 1990). In the ICADS model past knowledge is represented in the form of prototypical building type and site/neighborhood knowledge bases (Pohl et al. 1989). A subset of this information with the addition of a number of project specific parameters, such as site conditions, owner objectives and external constraints, constitutes the initial design program (i.e., specifications). In addition, the ICADS model provides access to a variety of more general reference information (e.g., catalogs, material and construction system characteristics, and environmental factors).

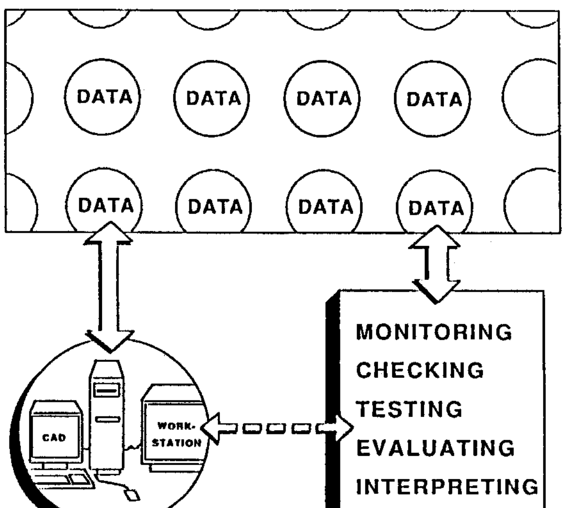

Fig.3: An Expert Advisory Facility

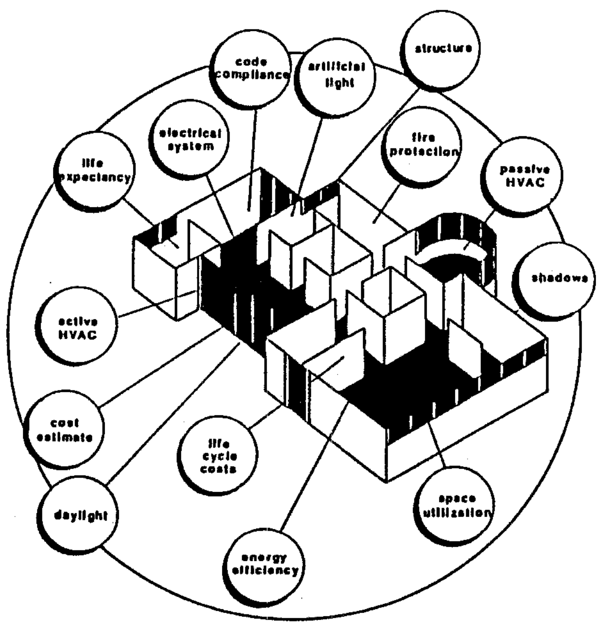

Fig.4: Typical Advisory Domains

The third target area of the ICADS implementation strategy called for the availability of intelligent design assistance comparable to the participation of consultants or experts in real world design projects. It was agreed that the implemented expert advisory facility should be capable of operating in background, continuously monitoring and evaluating the current state of the design solution, interrupting the human designer

only under critical circumstances (Fig.3). A second requirement was concurrency. Each expert should be able to perform its evaluations and develop its proposals without regard to operational priority or sequence. In other words, the advisory facility should function much like a real world project meeting with several consultants and the architect in attendance. In particular, it was agreed that every effort should be made to preserve the non-dictatorial nature of the dialog of such a real world meeting. Suggestions by one expert should lead to countersuggestions and consensus conclusions, rather than vetoes and categorical overrides.
These requirements are met in the ICADS model by a cooperative decision making model consisting of a blackboard control system and several independently executing domain experts. The blackboard contains two experts: a Conflict Resolver that makes countersuggestions to the evaluation results produced by domain experts; and, a Message Router that receives and sends information to all participants in the dialog. With the domain experts and the Conflict Resolver executing as separate processes on different machines virtually no scheduling takes place in the blackboard. Since all information related to the current state of the design solution is readily available to the dialog participants, reactions to proposals posted by any of the participants are spontaneous and virtually concurrent.

The ICADS Model as a Testbed

Implementation of the first version of the ICADS model, DEMO1, was completed in the latter part of 1989 (Pohl et al. 1989). Over the past two years the efforts of the CAD Research Unit have focused on two research directions: improvement and extension of the ICADS testbed model; and, exploration of the design environment provided by the DEMO1 version. Our findings to date can be divided into four categories: apparent validations; acceptable current restrictions; fundamental limitations; and, promising directions for further research.

Apparent Validations

The continuous responsiveness of the ICADS environment to the actions of the designer, through its ability to monitor and evaluate changes in the progressively evolving solution model and its concurrent conflict resolution capabilities, is clearly the most striking and pleasing feature of the model to any new user. Fears that the monitoring activities of an expert advisory facility may be disruptive and, therefore, not welcomed by the designer appear to be unfounded. In fact, the opposite has been observed. After working in the ICADS environment for only a short time the designer progressively comes to rely more and more on the evaluation capabilities of the system.

The cooperative decision making model which underlies the operation of the expert advisory facility appears to be a good match to the observed behavior of users during the early design stages. First, the iterative dialog between the domain experts and the blackboard, which may take from less than a second to several minutes, tends to reinforce the user's confidence in the validity of her own highly iterative design activity. Second, the ability of the system to bring expertise from multiple domains to bear on the current state of the solution supports the human designer precisely where she is most vulnerable. While the subconscious neural processes in the human brain that associate patterns and retrieve information are highly parallel, the conscious logical processes are essentially sequential (Caudill and Butler 1990). Accordingly, the human designer experiences difficulties dealing concurrently with more than two or three variables during the early design stages when the developing conceptual solution vacilates.

The availability of prototype information, even to the limited extent that such knowledge bases have been included in DEMO1, appears to be a fundamental requirement for any computer-based design environment. Particularly during the early design stages, designers tend to rely heavily on prototypical information from past design experiences to develop and explore solution alternatives. Schoen further categorizes this information as repertoires of functional types, references, spatial gestalts, and experiential archetypes (Schoen 1988).

The ability to monitor the interactions that occur on a continuous basis within the design advisory facility appears to be strongly supportive of the designer's need for experimentation. This became even more noticeable when the original static design interface was replaced with a dynamic version, in which the user could watch the values describing the current state of the design solution change during the dialog between the domain experts and the Conflict Resolver.
Work involving the development of domain experts in the ICADS project, to date, suggests that production rule-based expert systems are capable of representing the algorithms and heuristics required for the evaluation and generation of partial design solutions in narrow domains. It is likely that a great deal of headway can be made in this area very quickly. In most engineering fields there exists a substantial body of specific and generalized problem solving expertise that can be readily transferred into expert systems.

Acceptable Current Restrictions

The current version of the ICADS working model, DEMO2, comprises some 20 separate processes executing concurrently and communicating with each other through Unix socket connections. An extended version (being developed jointly with Battelle PNL, Lawrence Berkeley Laboratory, and the University of Oregon under the AEDOT project sponsored by the US Department of Energy) will add another four substantial processes to the current testbed model (Battelle 1988). These processes range from major procedural programs, such as the MountainTop CAD drawing package, and a large relational database management system (i.e., Oracle), to expert systems with between 50 and 300 rules.

The hardware environment available in support of the ICADS project has moved in recent months from the IBM-RT workstation platform to the new IBM-RS/6000 platform, with at least a tenfold performance increase. Specifically, the current ICADS model executes about twice as fast on one RS/6000 workstation than it does when distributed over six networked RT workstations. While the response time in both of these hardware environments is acceptable for demonstration purposes, a production version would be well served by a further tenfold increase in computing power. Indications are that the required computing power will be available at the desktop within a three to five year period (Bairstow 1987, Stroustrup 1990).

The scope of the ICADS testbed model has been limited in several respects (Pohl et al. 1988b, 1989).

- the body of information represented in each prototype knowledge base is restricted to a vertical segment of the data that are typically available for a building type, and the regional context of a site;

- the functionality of the implemented model supports only a narrow horizontal segment of the normal building design activity undertaken by architects, namely, the development of floor plans;

- the range of expertise provided by the system is restricted to subsets of the structural, acoustic, lighting, thermal, cost estimating, and space planning (adjacencies) domains;

- the ability of the user to interact privately with an individual domain expert is limited to a one-way conversation (i.e., the user cannot send information through the domain expert to the blackboard, nor can the results of the interaction be merged with the current state of the design solution).

Observations to date appear to indicate that these restrictions are not having significant impact on the testbed functions of the working model.

Fundamental Limitations

The Geometry Interpreter that maps the low level (point/line) geometric representation of the drawing in the MountainTop CAD environment to the high level objects that the system uses as the basis of all internal reasoning functions, is the weakest component of the ICADS model. It was designed to be a temporary bridge between the CAD drawing environment and the expert advisory facility, until a more direct and robust linkage could be developed. Work commenced at the beginning of 1991 on the implementation of a front-end module that will allow objects, such as site, building, and spaces, to be extracted directly from the textual descriptions of the design program (i.e., project specifications). The approach is similar to the 'design with features' concept that is favored in mechanical engineering and manufacturing design (Dixon et al. 1985a, 1985b, Simmons 1985, Pohl et al. 1991).

The coordination and arbitration of the results produced by the domain experts is a complicated undertaking. In the ICADS model this task is performed by a coordinating expert in the blackboard control system. While future versions of the ICADS model are likely to expand the coordinating expert to several interacting components, in the current version there is only one component, the Conflict Resolver. Many obstacles are encountered in this area. Compared to the considerable depth of knowledge available in narrow domains, little is known about the integration of narrow domain solutions into a single comprehensive solution model (Newell 1989).

There are good reason for this deficiency. First, the collection of knowledge in this area requires a level of design team continuity that cannot be achieved in practice. Particularly in the construction industry, design teams rarely stay together for more than one project. Therefore, the sample size is almost always too small to form the basis of a reliable set of generalizations. Second, building designers consider uniqueness a desirable attribute of a design solution (Schoen 1990, Archea 1987). Therefore, the application of general rules to specific design cases is greatly influenced by the interpretations of the designer.

In respect to the ICADS model this leads to the observation that the blackboard control system can only serve as a meaningful integrator of the domain experts if it is allowed to operate in partnership with the designer. This allows the coordinating expert to solicit the assistance and guidance of the designer, and the designer to redirect the coordinating expert, whenever the problem space changes significantly during the design session. The ICADS testbed model supports this requirement in a very limited manner only. Interaction between the user and the Conflict Resolver is restricted to situations where the Conflict Resolver cannot persuade the domain experts to reach consensus. This is a serious limitation of the current model that is under investigation at the present time. Future versions of the ICADS model must support a much higher level of interaction between the user and the coordinating expert(s) in the blackboard.

The Building Type and Site/Neighborhood knowledge bases that are included in the current ICADS model are limited in several respects. First, they are not sufficiently rich to support the wide spectrum of associations that designers typically make during the early design stages (Schoen 1988). During this phase designers draw upon past experience to relate aspects of the design problem to several varieties of prototypes, such as vertical prototypes, horizontal prototypes, and domain prototypes. The Building Type knowledge base is, at best, a single vertical prototype describing the design parameters and typical solution strategies of a specific type of occupancy (i.e., a Rural Community Center). Neither of the two knowledge bases contain information about horizontal prototypes such as the solution alternatives that pertain to the sloping curved access of an underground parking garage, or domain prototypes such as the structural guidelines that apply to buildings located in a seismic zone. Clearly, future versions of the ICADS model should support a richer set of prototypes.

Second, the prototype knowledge bases available in the current ICADS working model do not, in fact, represent integrated design solutions. The information they contain is mostly structured in terms of domains, referred to as 'information categories'. For example, the information category, 'structure', comprises three 'design issues': Structural System; Constructional System; and, External Facade. Each of these is linked to an hierarchical set of attribute groups and individual attributes. While some integration of design issues occurs at this lower level due to sharing of attributes across multiple information categories, there are no relationships that integrate design issues or information categories directly at a higher level. Strictly speaking, therefore, these currently available knowledge bases contain loosely linked narrow domain solutions. They could form the basis of integrated prototype design solution models, with the availability of an automatic knowledge acquisition facility in the blackboard control system.

Although proportionally more time has been spent on the development and implementation of the prototype knowledge bases then any other task, in the ICADS project, this remains an area with many unknowns. The structure of the knowledge bases is still largely untested. For example, the six domain experts that are included in the current ICADS testbed model utilize less than 15% of the information included in the Rural Community Center prototype. Additional domain experts will be required to validate the content of both the Building Type and Site/Neighborhood knowledge bases.

An overriding concern of the current implementation of the expert advisory facility in the ICADS model is that it assumes that the conflict resolution set required for the coordination of a representative number of domain experts can be largely predefined. This assumption is obviously not valid. The very nature of design as a process that involves creativity and intuition, operating in a design space that expands dynamically, defies the establishment of the type of detailed symbolic description that is required in support of a comprehensive design reasoning system.

The reasoning capabilities of the current ICADS working model, while perhaps satisfactory within the limited scope of the Expert Design Advisor, cannot be readily extrapolated to a much larger model involving two or three dozen domain experts and a comparable number of knowledge bases. Any attempt to predefine the necessary conflict resolution set, even if this were possible, would serve to constrain the design space and restrict rather than enhance the creative ambience of the design environment.

A better approach may be found in considering design as a largely unstructured environment in which conflict resolution is accomplished by agents that are endowed with simple tools. These agents gain knowledge of the environment through the application of their tools to the conflicts encountered. Work in this area has been conducted by others in respect to robots (Brooks 1990, Lozano-Perez et al. 1990). Existing industrial robots that function perfectly in the carefully structured environment of automated factories, tend to perform poorly in the unstructured environment of ordinary factories. Recognition of the fact that the unstructured environment simply contains too many potentially significant details to be structured into a symbolic model, suggests the need for an alternative approach: the progressive acquisition of knowledge through the interaction of simple mechanisms with each other and a complicated world.

Application of these principles to the ICADS model would suggest a conflict resolution structure that differs in several ways from the current expert advisory facility. It is clear that the conflict resolution structure must be generated in a more dynamic manner and that it must provide the capability for making decisions that are customized by the user, or user group. However, this does not mean that all of the functions of the Conflict Resolver need to be developed specifically for each user. Indeed, there are many requirements and constraints that are quite static. As a result, it would appear plausible that a new or modified conflict resolution facility could be realized through a variety of development techniques, including: predefined universals; design project constraints; preferences learned during previous use; and, implications deduced from changes in the structure. It should be noted that in order to provide for these new functions, it will be necessary to expand the scope of the conflict resolution structure beyond that of only handling conflicts. New knowledge will need to be added to allow the system to reason about its own functions.

Promising Directions for Further Research

The current ICADS working model provides a convenient environment for experiments in expanding the conflict resolution structure. In the current model the Conflict Resolver is depicted as a single component. However, it can be implemented as a set of processes that communicate with the entire system, including each other, through the existing blackboard structure. In addition, it can be implemented as a miniature blackboard structure itself, or as a combination of blackboard and non-blackboard structures. The underlaying framework of the working model does not limit the agents to any particular purpose, it merely provides the mechanism for communication. As a result, parts of the conceptual conflict resolution can be realized through different types of software units. They can execute on different kinds of hardware, and they can have their own control mechanisms. Furthermore, the automatic means by which the system as a whole is configured makes it possible to quickly create a new combination of blackboard connections (Myers et al. 1991).

Work has been in progress since the middle of 1991 to extend the horizontal segment of the normal building design activity currently supported by the ICADS model into the program development stage. This is typically the earliest stage of the design activity during which the designer collects programmatic information and defines solution objectives. The Program Development Module must be highly interactive to provide the designer with adequate opportunities for participating in the development of the initial design criteria. It is expected that this module will lead directly into the Predesign Module that has been described in a previous publication (Pohl et al. 1991). Once these additions are in place the extended ICADS testbed model will support a more comprehensive set of design functions. This support will extend from the earliest definition of project specifications, including the development of design criteria, guidelines, and strategies, through the manipulation of symbolic solution objects for the purpose of developing preliminary site and space layouts, into the currently supported floor plan generation stage.

Extensions already implemented in the recently completed DEMO2 version of the ICADS model include a limited domain expert interaction capability. At the present time this capability applies to only three of the six domain experts, and is restricted to graphic output of domain expert proposals. It is planned to enhance the GXI Client/Server, which manages these conversations between the user and an individual domain expert, so that data can be sent from client to client (Pohl and La Porta 1991). Eventually, the ICADS model should be expanded to allow the designer to send the results of a private design experiment performed with a particular domain expert to the system at large, for validation by the expert advisory facility.

Conclusion

The ICADS project has been moving concurrently in two directions during the past year. On the one hand, the capabilities of the current working model, although limited in many respects, are being incorporated into several potentially usable and commercially viable products. This commercial interest is driving efforts to enhance and extend the ICADS model into functional areas that are essential for real world applications. On the other hand, recognition of the need for dynamic redefinition of the knowledge base during the design activity is redirecting our reasearch interests to explore collaborative designer-system learning strategies. This research direction is likely to gather increasing momentum during the next several years and may lead to new ICADS models, substantially different from the current version.

References

Archea J. (1987); 'Puzzle-Making: What Architects Do When No One Is Looking'; Kalay (ed.) Principles of Computer- Aided Design: Computability of Design'; Wiley, New York.

Bairstow J. (1987); 'Personal Workstations Redefine Desktop Computing'; High Technology, March (pp.18-23).

Battelle PNL (1988); 'Advanced Energy Design and Operation Technology Research: Recommendations for a US Department of Energy Multiyear Program Plan'; Battelle Pacific Northwest Laboratories, Richland, Washington, December.

Brooks R.A. (1990); 'A Robot that Walks: Emergent Behaviours from a Carefully Evolved Network'; Artificial Intelligence at MIT - Expanding Frontiers, Winston and Shellard (eds.), MIT Press.

Caudill M. and C.Butler (1990); 'Naturally Intelligent Systems'; Bradford/MIT Press.

Dixon J.R., E.C.Libardi, S.C.Luby and M.Vaghul (1985a); 'Expert Systems for Mechanical Design: Examples of Symbolic Representations of Design Geometries'; Winter Annual Meeting, ASME, Miami, FL, November (pp.29-46).

Dixon J.R. and M.K.Simmons (1985b); 'Expert Systems for Mechanical Design: A Program of Research'; ASME Design Engineering Division, 11th Design Automation Conference, Cincinnati, OH, September.

Gero J. (1990); 'Design Prototypes: A Knowledge Representation Schema for Design'; AI Magazine, August (pp.26-36).

Gero J., M.Maher and W.Zhang (1988); 'Chunking Structural Design Knowledge as Prototypes'; Working Paper, The Architectural Computing Unit, Department of Architectural and Design Science, University of Sydney, Sydney, Australia.

Kuhn T.S. (1962); 'The Structure of Scientific Revolutions'; University of Chicago Press, Chicago.

Lozano-Perez T., J.L.Jones, E.Mazer and P.A.O'Donnell (1990); 'Task-Level Planning of Pick-and-Place Robot Motions'; Artificial Intelligence at MIT - Expanding Frontiers, Winston and Shellard (eds.), MIT Press.

Myers L., J.Pohl and A.Chapman (1991); 'Computer-Based Intelligent Design Assistance: Concepts and Strategies'; First International Conference on Artificial Intelligence in Design, Edinburgh, Scotland, June 25-27.

NASA (1989); 'CLIPS Architecture Manual (Version 4.3)'; Artificial Intelligence Section, Lyndon B.Johnson Space Center, TX, May.

Newell A. (1989); 'The Quest for Architectures for Integrated Intelligent Systems'; IJCAI Conference Proceedings, Kaufmann.

Oxman R.(1990); 'Prior Knowledge in Design: A Dynamic Knowledge- Based Model of Design and Creativity'; Design Studies, 11(1) (pp.17-27).

Oxman R. and J.Gero (1988); 'Designing by Prototype Refinement in Architecture'; Gero (ed.) Artificial Intelligence in Eng- ineering Design, Elsevier, Computational Mechanics Publication, 1988 (pp.395-411).

Pohl K. and J.La Porta (1991); 'Graphical Interface Builder: A Client-Server Model'; InterSymp-91, Baden-Baden, Germany, August 12-16.

Pohl J., A.Chapman, L.Chirica and L.Myers (1988a); 'ICADS: Toward an Intelligent Computer-Aided Design System'; Technical Report, CADRU-01-88, CAD Research Unit, Design Institute, School of Architecture and Environmental Design, Cal Poly, San Luis Obispo, CA.

Pohl J., A.Chapman, L.Chirica, R.Howell and L.Myers (1988b); 'Implementation Strategies for a Prototype ICADS Working Model'; Technical Report, CADRU-02-88, CAD Research Unit, Design Institute, School of Architecture and Environmental Design, Cal Poly, San Luis Obispo, CA.

Pohl J., L.Myers, A.Chapman and J.Cotton (1989); 'ICADS: Working Model Version 1'; Technical Report, CADRU-03-89, CAD Research Unit, Design Institute, School of Architecture and Environmental Design, Cal Poly, San Luis Obispo, CA.

Pohl J., L.Myers, A.Chapman, J.Snyder, H.Chauvet, J.Cotton, C.Johnson and D.Johnson (1991); 'ICADS Working Model Version 2 and Future Directions'; Technical Report, CADRU-05-91, CAD Research Unit, Design Institute, School of Architecture and Environmental Design, Cal Poly, San Luis Obispo, CA.

Rittel H. and M.Webber (1984); 'Planning Problems are Wicked Problems'; Cross (ed.) Developments in Design Methodology, Wiley, New York (pp.135-1440.

Schoen D.A. (1988); 'Designing: Rules, Types and Worlds'; Design Studies, 9(3), July (pp.181-190).

Simmons M.K. (1985); 'Expert Systems in a CAD Environment: Injection Molding Part Design as an Example'; ASME Computers in Engineering Conference, Aug.4-8, Boston, MA (pp.77-82).

Simon H. (1984); 'The Structure of Ill-Structured Problems'; Cross (ed.) Developments in Design Methodology, Wiley, New York (pp.145-166).

Stroustrup B.(ed.) (1990); 'The Byte Summit: Future Directions'; Byte Magazine, September (pp.225-367).