Reflections on Computer-Supported Cooperative Design Systems

Wassim JABI
School of Architecture, New Jersey Institute of Technology, USA

Keywords: collaboration, groupware

Abstract: Computer-supported cooperative work has come to signify a process by which a computer system supports the activities of geographically dispersed participants and enables them to share ideas and artefacts. This paper aims to dispel this unified approach to the definition of and in turn the design of aids to cooperative work. The paper starts by surveying the history of the field of computer-supported cooperative work and design. Then, the paper presents three tasks and their associated aids. The first system helps designers to create and discuss a hierarchical building program and share schematic drawings. The second system allows a jury to anonymously review a set of submissions. The third system allows a hierarchically organised group of participants to search, retrieve, organise and share a set of digital media assets. All systems are web-based and use the same underlying object-oriented technology. The paper provides a brief case study that describes the main features of the three pieces of software as an example of the need for a variety of approaches in the support of cooperative work and design. The paper concludes by advocating an object-oriented, domain-specific approach to creating computer-supported cooperative systems based on the analysis of the task at hand.

1 INTRODUCTION: A HISTORY OF COMPUTER-SUPPORTED COOPERATIVE WORK AND DESIGN

To understand the current framing of computer-supported cooperative design as a set of technologies that enable geographically dispersed users to work collectively on a common problem, one must trace the evolution of the field back to the original ideas advocated by its *pioneers*. Unfortunately, since much of the current work derives its paradigm from those early ideas, many of the misconceptions of the early work still find their ways into current thinking. Thus, in the following section, I will trace as closely as possible the evolution of the field starting in the 1960s and ending in the year 2001. I chose to ignore the period 1970-1985 in which little if any work has been concerned with the topic at hand. I also find myself obliged to omit from this review the exploding body of work since 1995 in keeping with the relatively limited scope of this paper. I chose 1995 as a year to halt my survey because much of the work since then, while significant, builds on a solid foundation of ideas founded between 1986 and 1995. However, I do include a bibliographical list that will enable the reader to pursue and fully investigate that body of work on his/her own.

1.1 Computer-Supported Cooperative Work: The 1960s

One can trace the beginnings of the field now known as computer-supported cooperative work (CSCW) to the work of Douglas Engelbart in the 1960s. Specifically, to his seminal paper *Augmenting Human Intellect: A Conceptual Framework,* which was published as a research report to the Air Force Office of Scientific Research in October 1962 (Engelbart 1962). In this report he describes how computers can help a team solve problems and make decisions more efficiently. In his introduction, he outlines some of the disciplines, including design that might benefit from technology augmentation. However, he immediately rejects the idea of developing domain-specific solutions to solve a domain-specific problem. He derides these solutions as *isolated clever tricks*. Later in his report, in a section titled *Team Cooperation*, he outlines his vision of how the system will work:

> *Let me mention another bonus feature that wasn't easily foreseen. We have experimented with having several people work together from working stations that can provide intercommunication via their computer or computers. That is, each person is equipped as I am here, with free access to the common working structures. There proves to be a really phenomenal boost in group effectiveness over any previous form of cooperation we have experienced. They can all work on the same symbol structure, wherever they might wish. If any two want to work simultaneously on the same material, they simply duplicate and each starts reshaping his version -- and later it is easy to merge their contributions. The whole team can join forces at a moment's notice to 'pull together' on some stubborn little problem, or to make a group decision. Most points of contention are resolved quite naturally, over a period of time, as the developing structure of argument bears out one, or the other, or neither stand.*

A year later, at the 1963 Spring Joint Computer Conference held in Detroit, Michigan, Steven Coons for the first time writes about the need for a Computer-Aided Design System that enables synchronous interaction with multiple users (Coons 1963). It this vision that laid the foundation for what we currently call the field of Computer-Supported Cooperative Design (CSCD):

> *The Computer-Aided Design System should be capable of carrying on conversations with, and performing computations for several designers at several consoles substantially all at once. In this way each designer can be immediately aware of what the other designers are doing, and thus avoid one of the truly severe problems of intercommunication that designers face today.*

These pioneering visions of CSCW and CSCD set the research agenda tone for the next forty years. However, given that Engelbart and Coons dealt with theoretical situations and visionary solutions, their intuitive response was to specify a utopian and unified system. That is, in their writings, they viewed the problems of cooperative work and design as a congruent set that can be addressed with one system and one approach. Engelbart mistakenly believes that merging different solutions is an easy task and resolving conflicts can occur naturally. Similarly, Coons falsely believes that synchronicity and social-awareness are the only needed features in addressing the problems that designers face. This paper will challenge these assumptions through a review of the work done since 1962 as well as the design and implementation of three collaborative systems that will act as case studies in collaboration, competition, team hierarchy, access privileges, social awareness and information filtering.

1.2 Computer-Supported Cooperative Design: 1986-2001

Most of the computer-aided design research efforts in the 1970s and early 1980s concentrated on the problems of automating the design process as well as the geometric and conceptual modelling and visualisation of buildings (Jones 1970; Mitchell 1977; Turner 1988; Stiny 1989). The implications of technological augmentation to cooperative design had to wait until the technology caught up in the second half of the 1980s with the advent of hypermedia, the maturity of computer networks, and the invention of the internet. The CSCW field became widely known to researchers with the first conference on the subject in 1986 held in Austin, Texas. The definition of cooperative design as a type of cooperative work came two years later at the CSCW 88 conference held in Portland, Oregon. At that conference Susan Bødker, Pelle Ehn and others published a paper titled *Computer Support for Cooperative Design* in which they theoretically defined cooperative design as a type of cooperative work (Bødker et al. 1988). However, in contradiction to Englebart's unified vision of computer augmentation to group work, they warned in their paper against the generalization of solutions in one domain of cooperative work to another. It is after the publication of this paper that researchers, especially in the field of architectural design, began looking at computer-supported cooperative design as closely related to, but distinct from the field of computer-supported cooperative work.

The convergence of computer-aided design and the study of teamwork start to take place by the late 1980s and early 1990s. Bryan Lawson, in the second edition of his book *How Designers Think*, comes tantalizingly close to defining the then nascent field of computer-supported cooperative design (Lawson 1990). The last two chapters of his book are titled *Designing with Others* and *Designing with Computers* and although he does not fully extrapolate the fact that designing with computers might enable us to design with others, one cannot help but assume that he was starting to form that connection.

In the same year, William J. Mitchell also whets our appetite by coining the intriguing term *society of design* in the last chapter of the book *The Electronic Design Studio* (Mitchell, McCullough and Purcell 1990). The term, inspired by Marvin Minsky's 1985 *society of mind*, hints at Mitchell's thinking about the convergence of CAD and network technologies to provide users with various modes of interactions and exchange of artefacts. He does not write specifically about design collaboration in that chapter; however he sets the stage for his later work in that area. In 1991, Carl Tollander publishes a chapter titled *Collaborative Engines for Multiparticipant Cyberspaces* in Michael Benedict's book *CyberSpace: First Steps* (Benedikt 1991). In this chapter Tollander describes a collaborative cyberspace as metric space influenced by the actions of multiple participants. He defines collaboration as a "set of operations over a set of cyberspaces, in which a group of selective influences under the control of the participants direct the evolution of space."

A year later, in 1992, Jerzy Wojtowicz, James Davidson and William Mitchell co-author a paper in the proceedings of the Association for Computer-Aided Design in Architecture (ACADIA) titled *Design as Digital Correspondence* (Wojtowicz, Davidson and Mitchell 1992). It is the first paper in an ACADIA conference that discusses the potential of computers as aids for collaborative design. The paper discusses a case study in which students in two studios at institutions a continent apart were asked to co-design a project using a shared electronic bulletin board to post design artefacts (using the FTP protocol), review and exchange comments using electronic mail and speakerphone, and conduct an electronic, long distance jury. As evident in the title and the case study, the authors believed that cooperative design could be conducted in similar ways to holding a conversation with back and forth communication. While they did include synchronous collaboration in their project, the emphasis was clearly on asynchronous contributions to a common repository of information. This may have been due to the difference in time zones as well as the relative immaturity and lack of availability of real time synchronous software and corresponding infrastructure.

Starting in 1993, the field of computer-supported cooperative design begins to dramatically expand. As an example, the CAAD Futures '93 conference included a special section on "Cooperative Design" in which three papers were published (Maher, Gero and Saad 1993; van Bakergem and Obata 1993; Bhat, Gauchel and Van Wyk 1993). Conferences in Europe, especially those addressing Design Decision Support Systems and Cybernetics begin to focus on Collaborative Design (Jabi and Hall 1995a; Jabi 1996a, 1996b). In Asia, the *CAAD Futures* 1995 conference in Singapore included two separate sessions on cooperative design with a total of fourteen papers, including one by the author of this paper (Jabi and Hall 1995b). In North America, ACADIA also witnessed a similar expansion in papers submitted on the topic. Since the ACADIA 1995 conference, computer-supported cooperative design has been formally identified as a research topic that is solicited in the call for papers. The 1999 conference included nine papers on the topic; a peak number for ACADIA (Ataman and Bermudez 1999). The 2000 proceedings included five papers while the 2001 proceedings included four papers (Clayton and Vasquez de Velasco 2000; Jabi 2001).

Currently the number of published papers has expanded to the point where it is difficult to track and document within the scope of a conference paper such as the one at hand. However, the field still has very few books dedicated to the subject. In 1995, Jerzy Wojtowicz edited a relatively modest manuscript titled "Virtual Design Studio" that included essays reflecting on the topic of conducting design studios at a distance (Wojtowicz 1995). The manuscript also included documentation of the collaborative Virtual Village Studio Project (VVS) conducted by Harvard University, University of British Columbia, MIT, and the University of Hong Kong. The project to date remains as one of the largest prototypical implementations of cooperative design studios. While not directly addressing computer-supported cooperative design issues, William Mitchell's book "City of Bits", published in 1996 in both print and online, is considered a seminal work on the nature and the potential of an interconnected world where processes that used to require real space are replaced with counterparts in a virtual space (Mitchell 1996). Many of the ideas and warnings he includes in his book can be applied to the design of computer aids and collaborative environments. Four years later, Mitchell publishes a sequel titled "E-Topia" in which he turns his attention to the evolution of the city under the influence of the various technologies especially e-commerce (Mitchell 2000).

Lastly, in 2000, Stephen Scrivener, Linden Ball and Andrée Woodcock edited and published the proceedings of the CoDesign 2000 conference held at Coventry University (Scrivener, Ball and Woodcock 2000) that included fifty papers on the topic at hand. The book is an indication of the maturity of the field and is a good survey of the various sub-topics and themes currently being pursued in this area of research.

2 MISCONCEPTIONS ABOUT CSCD

While it is dangerous and unwise to paint the field of CSCD with a broad brush, it is useful to point out common misconceptions plaguing the field and subsequently the design of corresponding computer aids. In this section, I will briefly list some of the most common misconceptions, explain them and discuss ways of avoiding them.

1) *Cooperative design is one thing*. The first misconception is the one mentioned by Douglas Engelbart and Steven Coons; mainly that cooperative design is a single activity that needs to be supported using a single system or solution. We currently know that design cannot be easily defined or reduced. It is certainly more complex, subtle, and multi-faceted than initially thought. Integrated systems that claim to support the full spectrum of collaborative design usually fail due to both their complexity as well as inability to deal with certain aspects of design. They tend to force a particular view of design that participants may or may not agree with. Once a participant feels that a system is demanding a particular method to be followed, that user will reject the system and subsequently allow it to fail.

2) *The more the better*. A second misconception is that the problems of cooperative design are solved by providing more information to everyone. The unfortunate side-effect of this approach has been what we call information overload. That is, the abundance of information has helped obscure relevant information rather than make it more readily accessible. As an example, we all now complain about the information overload

present on the Internet that hinders our search for relevant information. We are beginning to see systems that emphasize information filtering.

3) *We are all team players*. A third misconception is that the term cooperation is synonymous with collaboration, teamwork, and group effort. That is, those participants in a cooperative design environment are always engaged in a positive and helpful mode of co-work. However, we all know that internal design reviews in practice and academia, architect-client meetings, and architect-consultant meetings are usually a mix of collaborative and adversarial work. Systems should not only support conflict resolution, but perhaps encourage conflict to bring out the desired solution. Gaming theory has realized the potential of conflict, yet we rarely see that applied in a CSCD system.

4) *Two heads are better than one*. A fourth misconception is that group work is always preferable to individual work and that synchronous collaborative work is always beneficial. If we use traditional modes of co-design as an example, we can easily derive the fact that drawing on the same piece of paper at the same time by multiple users is certainly not a useful method of co-designing. We also know from field research that group discussions work best when individuals bring significant work done previously in a private context to the meeting (Jabi 1996a).

5) *The open floor plan*. A fifth misconception is that by bringing everyone to the same shared workspace cooperation will take place naturally. What researchers sometime forget is that team members are not all in the same hierarchical position. In almost any cooperative system, there will be a need to include concepts of privacy, access privileges, and group hierarchy.

3 CASE STUDIES

In the interest of dispelling the above misconceptions, this section includes a description of three case studies of computer software that supports cooperative activity in the field of architecture. While the reader may come to the conclusion that some of these systems do not support design activity, it is important to note that design is not necessarily confined to the act of creating drawings and artefacts that represent built structures. Creative design activity is multi-faceted and includes the activities supported by the systems described here. All systems described here use a three-tier object-oriented, web-based scheme coupled with a relational database for server-side storage. The first system integrates this technology with Java-based tools for synchronous web-based collaboration.

3.1 Case Study 1: Collaborative Space Programming and Design

This tool allows members of a design team to collaboratively specify a hierarchical spatial program for an architectural project. The represented artefacts have built-in intelligence that allows them to respond to user actions and manage their own sub-artefacts. Furthermore, the software goes beyond outlining functions to support synchronous collaborative design by linking each item in the spatial program to a detail page that allows file uploading, real time group marking of images, and textual chat. Thus, the software offers a seamless transition from the largely asynchronous definition of an architectural program to synchronous collaboration when needed. These sessions are artefact-based in the sense that they get automatically initiated once participants visit the same architectural space in the program hierarchy. By being artefact-based, the system allows sub-groups to meet privately and work on parts of the overall shared workspace un-distracted by what others are doing. A detailed description of this tool has been published in (Jabi 1998, 1999, 2000).

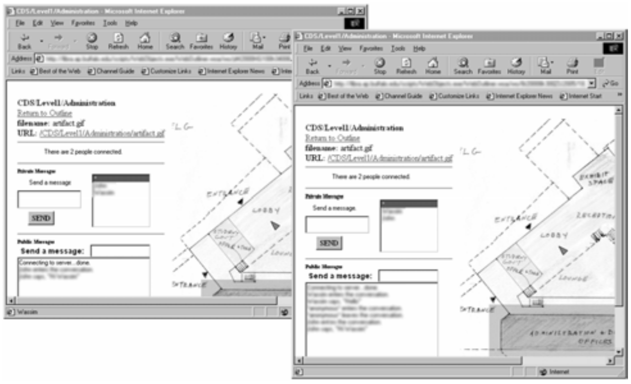

Figure 1 Screen Shot of the Synchronous Aspects of the System

3.2 Case Study 2: Double-Blind Peer Review System

This tool allows a group of participants to submit competitive entries for double-blind peer review. Examples include submissions to conferences, design competitions and juries. The system allows the participants to register their entry and submit it anonymously using only a unique registration number. Reviewers are invited and added to the system by system administrators. Reviewers are then assigned entries to review online and are offered an online form that allows them to score the entry, make other recommendations and give public feedback to the authors of the entries as well as private feedback to the administrators (Figure 2). While the same entry may be assigned to multiple reviewers, reviewers do not know who else is reviewing that entry or even other entries. Thus they are unaware of the current standing of the entry. This social-unawareness built into the system ensures fairness and lack of bias by reading other reviewers' reaction to the same entry. The system then tallies the comments and average scores and sorts the papers such that the administrators can make an informed decision. The system illustrates that a common goal, the assembly of a set of successful entries, is sometimes achieved by competition rather than collaboration. The system clearly defines a hierarchy of participants (entry authors, reviewers, and administrators) and ensures social unawareness and privacy when needed (Figure 3). The system was successfully used to review the entries to the ACADIA 2001 Conference (Jabi 2001).

Figure 2 Screen Shot of Review Submission Form

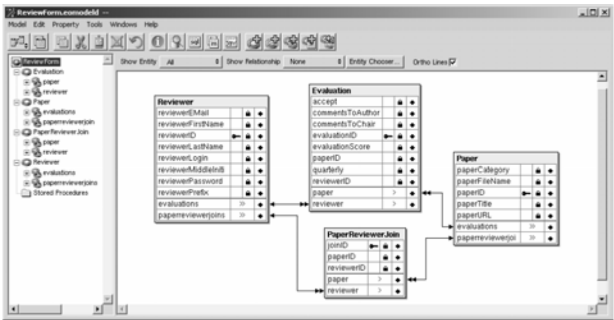

Figure 3 Database Schema Illustrating the Objects for Entry (Paper), Reviewer and Evaluation

3.3 Case Study 3: A Digital Asset Management System

ViSTA, a Virtual Slide Tray Archive, is a digital asset management and display system designed for education (Figure 4). The system enables instructors to search, select, sort and save in virtual sets (called trays) a database of digital assets (Figure 5). They can then project those trays for in-class presentations as well as allow registered students to view them at will from any internet-connected computer. Once registered using a password, the students' ViSTA homepage automatically displays the courses they are registered for and the associated trays for that particular course. Students can also create and modify their own personal trays organised in any fashion they want. The system illustrates the need for both asynchronous and synchronous modes of work as well as a clear definition of group hierarchy. In addition, the system allows participants to peruse shared workspaces as well as create and control access to private workspaces.

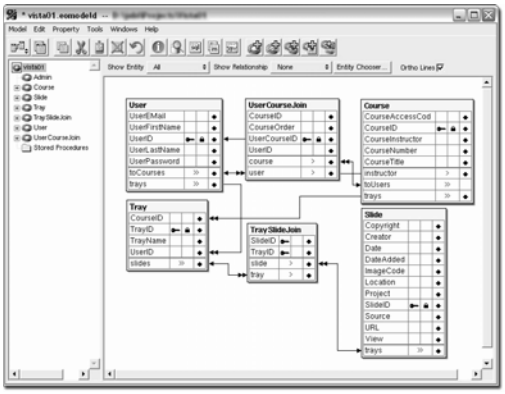

Figure 4 Database Schema for the ViSTA System

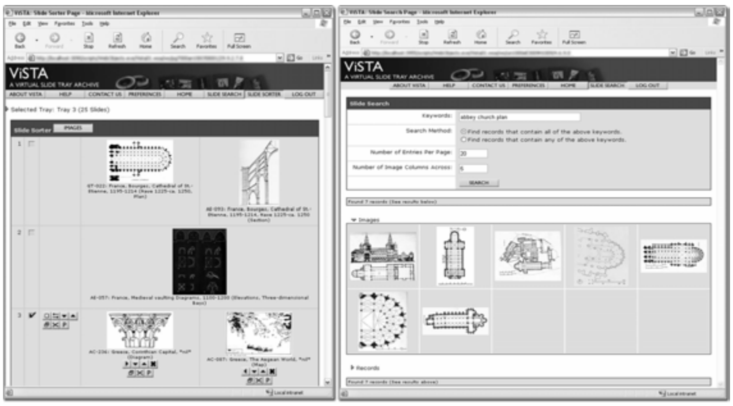

Figure 5 Screen Shot of Search and Sorting Functions in ViSTA

It is important to note here that all three systems not only use the same underlying object-oriented framework, but that they also share common reusable components. If designed properly, these components can provide a rich cellular palette that allows for the rapid development of full systems that address different functions.

4 CONCLUSION

Through a survey of the history of computer-supported cooperative work and design and the study of three example implementations, this paper sets to challenge the unified and integrated approach to supporting cooperative design. Just as large integrated CAD systems have failed to support design except in the rarest and most restricted of domains, integrated CSCD systems if pursued will fail to address the complexities and subtleties of cooperative design. Instead, this paper argues for a domain-specific and task-specific approach that solves one or a small number of problems at once. Doing so requires careful task and user analysis. That does not mean that solutions are not generalisable. On the contrary; the case studies illustrate how a fundamentally similar object-oriented framework can be re-used and re-adapted for solving different problems. By following a component-based approach, one can easily re-assemble a new system by selecting, modifying and adding standardized components. The breaking down of the design of the system into components is what will ensure its ability to adapt to change. It is this inability to adapt that causes large integrated systems to ultimately fail. Finally, the reusability of components like everything else will depend on good design.

REFERENCES

Benedikt, M. 1991. *Cyberspace: First Steps*. Cambridge, MA: MIT Press.

Bermúdez, J. and O. Ataman. 1999. Media and Design Process. *Proceedings of ACADIA 1999:* ***Media and Design Process***, eds. O. Ataman and J. Bermudez, Salt Lake City, UT: ACADIA.

Bhat, R.R., J. Gauchel and S. Van Wyk. 1993. Communication in Cooperative Building Design. *Proceedings of CAADFutures 1993,* eds. U. Flemming and S. Van Wyk, 481-493. New York: North Holland.

Bødker, S. et al. 1989. Computer Support for Cooperative Design. *CSCW 88, Proceedings of the Conference on Computer-Supported Cooperative Work*, ed. I. Grief, 377-394. New York: ACM Press.

Clayton, M.J. and G.P. Vasquez de Velasco. 2000. Eternity Infinity and Virtuality in Architecture. *Proceedings of ACADIA 2000: Eternity, Infinity and Virtuality,* Washington, DC: ACADIA.

Coons, S. 1963. An Outline of the Requirements for a Computer-Aided Design System. *Proceedings of the 1963 Spring Joint Computer Conference,* 299-304. Baltimore, MD: Spartan Books.

Engelbart, D. 1962. *Augmenting Human Intellect: A Conceptual Framework*, SRI Project Number 3578, Menlo Park: Stanford Research Institute.

Jabi, W. and T. Hall. 1995a. The Role of Computers in Synchronous Collaborative design. *Proceedings of the 14th International Congress on Cybernetics*, 71-76. Belgium: International Association for Cybernetics.

Jabi, W. and T. Hall. 1995b. Beyond the Shared Whiteboard: Issues in Computer-supported Collaborative Design. *Proceedings of CAADFutures 1995*, 719-725. Singapore: National University of Singapore.

Jabi, W. 1996a. An Outline of the Requirements for a Computer-supported Collaborative Design System. *Open House International*, vol. 21, no 1, 22-30.

Jabi, W. 1996b. Domain-specific Tools for Collaboration in Architectural Design. *Proceedings of the 3rd International Conference on Design and Decision Support Systems in Architecture and Urban Planning*, 248-259. Eindhoven: Technical University of Eindhoven.

Jabi, W. 1998. The Role of Artefacts in Collaborative Design. *Proceedings of CAADRIA 1998*, 271-280. Osaka, Japan: CAADRIA, Kumamoto University.

Jabi, W. 1999. Integrating Databases, Objects and the World-wide Web for Collaboration in Architectural Design. *World Wide Web as Framework for Collaboration, Proceedings of the Focus Symposium in Conjunction With the 11th International Conference on Systems Research, Informatics and Cybernetics*. Baden-Baden, Germany: The International Institute for Advanced Studies in Systems Research and Cybernetics.

Jabi, W. 2000. WebOutliner: A Web-based Tool for Collaborative Space Programming and Design. *Proceedings of ACADIA 2000: Eternity, Infinity and Virtuality,* 195-202. Washington, DC: ACADIA.

Jabi, W., ed. 2001. Reinventing the Discourse: How Digital Tools Help Bridge and Transform Research, Education and Practice in Architecture*. Proceedings of ACADIA 2001: Reinventing the Discourse,* ed. J. Wassim, Buffalo, NY: ACADIA.

Jones, J.C. 1970. *Design Methods: Seeds of Human Futures*. New York: John Wiley.

Lawson, B. 1990. *How Designers Think: The Design Process Demystified*, 2nd ed. Cambridge: Butterworth Architecture.

Maher, M.L., J.S. Gero and M. Saad. 1993. Synchronous Support and Emergence in Collaborative CAAD. *Proceedings of CAADFutures 1993,* eds. U. Flemming and S. Van Wyk, 455-470. New York: North Holland.

Mitchell, W.J. 1977. *Computer-Aided Architectural Design*. New York: Van Nostrand Reinhold.

Mitchell, W.J. 1996. *City of Bits: Space, Place, and the Infobahn*. Cambridge, MA: MIT Press.

Mitchell, W.J. 2000. *E-topia: Urban Life, Jim - But Not As We Know It*. Cambridge, MA: MIT Press.

Mitchell, W.J., M. McCullough and P. Purcell. 1990. *The Electronic Design Studio: Architectural Knowledge and Media in the Computer Era.* Cambridge, MA: MIT Press.

Stiny, G. 1989. What is a Design? *Proceedings of ACADIA 1989: New Ideas and Directions for the 1990s*, ed. C. I. Yessios, Gainesville, FL: ACADIA.

Turner, J.A. 1988. A Systems Approach to the Conceptual Modelling of Buildings, *CIB Conference on CAD*, ed. Per Christensen, TC 5.2 and 7.8. Lund Sweden.

Van Bakergem, W.D. and G. Obata. 1993. Making the Problem Visible: Project Specific Information in Collaborative Design. *Proceedings of CAADFutures 1993,* eds. U. Flemming and S. Van Wyk, 471-480. New York: North Holland.

Wojtowicz, J. 1995. *Virtual Design Studio*. Hong Kong: Hong Kong University Press.