1 Introduction

In the following, we describe our vision of a collaborative engineering environment, the Collaborative Design Studio, consider issues and questions relating to the effective design and evaluation of such systems, and describe proposed studies for evaluating and evolving the current environment. This work is being undertaken at the Geometric and Intelligent Computing Laboratory at Drexel University. While preliminary development of the studio has proceeded using relatively informal input from professional design engineers, we feel it is essential to incorporate direct evaluation by intended users of the system to ensure that it accurately meets their needs and fits into their work environment. Such information will be obtained through surveys and observation of subjects as they complete specified tasks, analyzed, and used to guide further development of the system.

2 Current Vision of a Collaborative Design Studio

The fundamental goal of the Collaborative Design Studio project is to enhance the design engineering process through the integration of computer-aided design and engineering tools, communication tools, and archiving functions. Design context is automatically extracted from CAD tools for inclusion in communications, and all design-related communications are automatically archived in a searchable database. We are attempting to achieve this while incorporating commercial off-the-shelf (COTS) tools, with underlying integration functions being transparent to users.

The proposed environment should provide a number of enhancements to the design engineering process. A central advantage will be the enabling of efficient and effective distributed design efforts, with geographically separated engineers able to work together as effectively as those housed together. The capabilities of the studio are being specifically targeted to this increasingly important work mode; however, they should prove advantageous in traditional settings as well. A significant benefit should be a reduction in design development time, facilitated by the automatic inclusion of design context into communications. Even frequent and informal forms of communication (such as e-mail) will include precise and unambiguous reference to parts and features in the current design, making it easier for engineers to have detailed discussions without requiring face-to-face meetings. The integration of communication tools and design tools allow these contextual references to initiate the automatic display of the relevant design model or diagram, freeing the receiver of a communication from the task of locating and loading the appropriate file. In addition, the automatic inclusion of specific, unambiguous design references should reduce the number of errors or misunderstandings generated by misinterpretation of designer-produced verbal descriptions of the design feature and context to which the communication applies. Finally, the automatic archiving of all communications, together with embedded design context information, should greatly aid many design activities. Design reuse and adaptation will be facilitated by designers’ having access to all communications related to previous design efforts, which can provide valuable insight into tradeoffs and compromises made which might not be reflected in formal documentation. During protracted design efforts, such an archive may provide the only access to ratio-
Design Studio consists of three groups of tools: CAD/CAE tools, collaborative work tools, and archiving tools. These are integrated through a number of software modules. The design of the environment has been guided by two requirements: standard commercial tools should be incorporated whenever possible, so as to avoid expecting designers to use proprietary software in place of effective and familiar alternatives; and the integration of tools should be as nearly transparent as possible, placing few additional burdens on users. The current prototype utilizes the Structural Dynamics Research Corporation I-DEAS suite as its CAD/CAE package, Netscape Communicator as the email client, Collaborative Virtual Workspace\(^1\) (CVW, developed by the Mitre Corporation) as a videoconferencing tool, and Oracle as the database archive (the search interface to this archive being web-based, allowing the use of any web browser). However, the integrating modules have been designed with an emphasis on “package independence”, to allow substitution of alternate packages with relatively minor effort (as long as the substitutes provide a sufficiently rich feature set — for example, CAD/CAE software with an application programming interface permitting the extraction of design information).

Design context is extracted from the current design for embedding into a communication by the client communication module. This module communicates with the CAD/CAE software (using the Open I-DEAS API in the case of SDRC I-DEAS) and creates an XML document incorporating the extracted design data organized according to a design-communication specific message model. This message model is central to the integration of the components in the studio: all design and communication context is captured in these messages, which provide the fundamental level of organization of information. A designer initiates a communication through a script which first calls the client module to extract the design context and create the XML document and then starts the appropriate communication tool, with options that associate the generated XML document (as an attachment in the case of email) and automatically include the archiving server as a participant (or recipient) in the communication. The user will observe little of this background processing, being presented with the selected communication tool as in the normal (unintegrated) setting. The client communication module also functions on the receiver side, using the attached XML design context document to display the appropriate design part in the CAD package.

The archiving server extracts design context from the XML document associated with a received communication and uses it to drive storage of the communication in the archive. The server also permits web-based search access to archived communications, returning retrieved information in the form of XML documents which are used by the communication module to display the associated design files.

While we feel the Collaborative Design Studio will provide many obvious benefits for designers, there are still a number of questions to be addressed. One particular issue is the amount and level of design context information to include into the XML documents. The message model has been designed to be extensible, permitting varying types of context to be included; however, this leaves open the question of exactly which information to include, which cannot be adequately answered without some form of input from the intended users of the system. In addition, it will be important to understand the role of the various tools used by designers when they employ the integrated suite in the Collaborative Design Studio. Some questions to be addressed are: to what extent does integration enhance the design process, and how can this be measured; how is the pattern of use of the tools changed when integration is incorporated; and how does access to archived communications influence design efforts. The issue of how we hope to obtain this information is addressed below.

4 Formulating Evaluation Questions and Procedures

In thinking about ways to facilitate use of an engineering design environment, there are two levels of design criteria that are important. One level is the interface or “tactical” level of design. The second is the interaction or “strategic” level. Interface level design criteria address issues such as how interface components and widgets should be built or “shaped.” For example, aircraft cockpit instruments need to show certain necessary information clearly. Similarly, in a wide range of circumstances, knobs, switches and dials need to be easily discriminable by sight and/or by touch. In contrast, interaction level design criteria address the goals

---

\(^1\)http://cvw.mitre.org
of the user and focus on the way the user thinks about the task. Certainly, a concern with interface features is important. Badly designed components can make the user’s task more difficult, lead to unnecessary errors, and even prevent successful task accomplishment. However, optimizing interface features without attending closely to the overall flow of interaction can lead to unworkable or unusable systems.

Adopting the goal of understanding the designers’ tasks and how they proceed in solving their research problems immediately commits one to developing explicit answers to several questions. Exactly who are the target users? What knowledge are they assumed to have, and what are their skills? What specific tasks will such users want to complete and are there generic types of tasks or task characteristics? How will successful user performance be measured or assessed? As a number of sources on Task Analysis [8, 9] have pointed out, the user’s task can have a significant impact on how tools are used or not used. Consider the use of computer aided design (CAD) systems in architecture. It is possible to divide architectural design into two broad categories of activity. One type of activity is associated with preparation of a model or design representation to illustrate the design to a customer. Another type of activity is associated with sketching out and developing a novel design concept. Even casual observation of usage patterns of architectural CAD systems reveals task differences. While it is not unusual to find someone trained in the use of an architectural CAD system putting it to use in polishing and preparing drawings for presentation, it is quite a challenge to find an architect using such a CAD system while creating an original design concept.

As pointed out by Dym [4], there are three identifiable types of engineering design activity. These are routine, variant, and conceptual. Routine design involves taking an existing design and making minor tweaks in the artifact being designed to develop an improved artifact. Variant design involves using existing components or subassemblies to create a new version of an existing artifact. Conceptual design involves coming up with a new design concept. Eventually existing components or subassemblies may become components of the new artifact design is that of a new artifact.

One place to begin identifying task-critical functionalities for a human centered engineering design environment is with a realistic understanding of current procedures, practices and activities that the environment is meant to augment or replace, as discussed in Hewett and DePaul [6]. For example, there are a variety of reasons for believing analogical thinking is a key process involved in creative advances in science and engineering [7]. Of particular relevance to thinking about the creation of design environments is work on the use of analogy, both in individual problem solving [2] and in generating new insights during the social process of science and engineering [3].

Clement [2] found that the steps used by scientists in developing a problem solution are not necessarily derivational or formal. He gave a mechanics problem to 10 participants sophisticated in physics but with no extensive experience solving sample problems from the class of problems used. The expertise level of participants in the study ranged from graduate students to a Nobel laureate in physics. Participants were asked to “think aloud” while solving the problem. Examination of video tapes from these sessions revealed that only one participant used anything resembling an abstract principle to derive a problem solution. The typical solution procedures involved memory search for an analogous situation with similar characteristics. The search was conducted through association or by examining dimensions of similarity. Next the typical problem solver developed a possible solution on an analog which had been judged to be appropriate. Once the solution had been worked out for the analog, the individual then proceeded to test the solution on the original problem.

Dunbar [3] reported on a year-long study conducted in four molecular biology laboratories, where he followed several projects. The processes studied included “planning the research, executing the experiments, evaluating the experimental results, attending laboratory staff meetings and public talks, planning further experiments, and writing journal articles (p. 365).” Dunbar reports that, “analogies were an important source of knowledge and conceptual change (p. 381).”

The researchers Dunbar studied used three different classes of analogy. Local analogies were those within the same domain as the research being conducted and were drawn from a previous experiment to a current one. This type of analogical reasoning typically occurred when a scientist would attempt to map a currently unsuccessful experiment onto a similar experiment that had been successful. Local analogies seemed to be one of the main mechanisms for driving the research program forward. Regional analogies involved mapping a system of relationships from a similar domain onto the problem domain being studied. This type of analogical reasoning was typically “employed when the scientists were working on elaborating their theory and planning a new set of experiments (p. 383).” These regional analogies seemed to be one of the main mechanisms for allowing the scientists to fill in gaps in their own knowledge and to suggest new questions to ask about the entities being studied. Long-distance analogies involved mapping a concept from a very different domain onto the problem domain being studied. Typically, “long-distance analogies were used to highlight features of the research that were salient,” and were used to, “bring home a point or to educate new staff members of a laboratory (p. 383).”

Pulling together the diverse threads just reviewed, it is possible to see readily that a collaborative engineering de-
sign environment offers a wide field of questions which need to be addressed and simultaneously creates an environment in which a number of questions about the nature of design and collaborative design can also be formulated and answered. There are those questions which are concerned with the interface level details, i.e., are the tools usable and useful given their intended purpose. In addition there are those questions about facilitating interaction, both a single designer’s interaction with the tool set and the social interaction among designers.

Cutting across these considerations is a concern with the role of analogical thinking in both individual and collaborative engineering design. Thus we propose exploring the apparent parallel which exists between Dym’s categorization of engineering design activities and Dunbar’s comments about the types of analogy that were used in the research laboratories which he studied — i.e., are routine, variant and conceptual design characterized or driven by the use of local, regional, and distant analogies, respectively. Our preliminary hypothesis is that the role of analogy in collaborative engineering design is analogous to its role in scientific research.

In considering the nature of the collaborative design environment, these inquiries into characteristics of the engineering design process will guide the identification of the features of the message model which will be used to archive design decisions and provide design context during communication.

5 Proposed Approaches to Evaluation of the Collaborative Design Studio

5.1 Survey of working engineering designers

We have under development a survey which we propose to circulate to working engineering designers. In this survey we explore both the types of knowledge which engineering designers feel they use and the nature of how they store, retrieve, and/or communicate that information to other designers. Once the survey document has been pilot tested to improve its wording, etc., we intend to disseminate the questionnaire to a large number of design engineers in New Jersey, Pennsylvania, Delaware and Maryland in conjunction with our collaborators with the Delaware Valley Industrial Resource Center (DVIRC)². Not only will this survey provide us with normative data about what designers currently think about what they do, it will also be used to seek volunteers to participate in further studies in which experienced engineers are the most appropriate participants. In short, the survey provides us with a “map of the terrain”, an indication of tasks that need to be studied and questions to be answered.

5.2 Thinking out loud studies

We currently anticipate the need for one or more studies in which engineers and/or student engineers participate which will allow us to learn more about the weaknesses of the tools and environment at the interface level when infelicities or outright errors are identified for us by people engaged in a design task — e.g., is there an unusable tool or feature? In addition, we anticipate the need for one or more studies focusing on the interaction level of analysis so that we can improve the cognitive engineering of the environment and tools — e.g., have we structured the environment in such a way that the designer or design team are forced to hold more information in memory than a human being can maintain in an active state?

Protocol analysis, or thinking out loud, is a technique to obtain subjects’ verbal reports about a task being performed concurrent with the process of completing that task. It has been shown that this technique can produce valuable data while not altering subjects’ mental processes [5]. Subjects are instructed to verbalize their thoughts while performing the task being studied — i.e., to think out loud. It should be noted that this is distinctly different from having participants explain their actions while they are engaged in the task; in particular, it is relatively unobtrusive to the problem-solving process (except in certain exceptional cases, such as time-critical activities). The technique gives insight into subjects’ mental processes, course of thought and problem-solving approach in addressing the specified task. This real-time verbal report will also be supplemented with retrospective reports and interviews. The latter provide an indication of what participants “think they were doing” during the activity; the former gives evidence of what they “actually were doing.”

It is generally desirable that subjects employed in such a study have prior familiarity with the process being studied to gain any reliable insight into that process. Thus if the process of designing a mechanical artifact is being studied, the participants of the study should be people familiar with mechanical design, perhaps industrial designers or design engineers. If in addition what is being evaluated is not the CAD software itself, but the integration of that software into an environment including other applications, then the participants should be familiar with the use of a CAD package (though not necessarily the specific package used in the study, since most CAD applications are similar in behavior, at least in those components likely to be relevant for the study). Considering that each test group should consist of at least 10 teams of subjects to guarantee an acceptable level of confidence in the results, it becomes clear why such

²dvirc.org
For our particular problem, the main task assigned to users would be to solve a design problem as a group, collaboratively, while being spatially separated. The chosen problem needs to be relatively simple, able to be solved in a matter of hours, while still requiring active collaboration of members of the design team. The activity to be observed is how CSCW tools are used in the process of design, and how their integration with design authoring tools influences the activity — in particular, is such integration beneficial, and what kinds of information about the design should be incorporated into communications. A control group of designers will work on a design problem under “normal” conditions, in which design authoring and CSCW tools are both present but not bound together in any way; another group will be given access to the Collaborative Studio, with communication tools integrated with CAD software. Experimenters will observe subjects, recording what kind of information is being exchanged, how, in which cases, and for what purposes participants use the communication tools at their disposal, and other pertinent data. The observers will record in addition how long it takes to complete the task, the frequency and amount of information exchanged, and the classification of that information.

For our particular problem, the main task assigned to users would be to solve a design problem as a group, collaboratively, while being spatially separated. The chosen problem needs to be relatively simple, able to be solved in a matter of hours, while still requiring active collaboration of members of the design team. The activity to be observed is how CSCW tools are used in the process of design, and how their integration with design authoring tools influences the activity — in particular, is such integration beneficial, and what kinds of information about the design should be incorporated into communications. A control group of designers will work on a design problem under “normal” conditions, in which design authoring and CSCW tools are both present but not bound together in any way; another group will be given access to the Collaborative Studio, with communication tools integrated with CAD software. Experimenters will observe subjects, recording what kind of information is being exchanged, how, in which cases, and for what purposes participants use the communication tools at their disposal, and other pertinent data. The observers will record in addition how long it takes to complete the task, the frequency and amount of information exchanged, and the classification of that information.

5.3 Interaction Process Analysis

Another aspect of the Collaborative Studio design that we intend to study is the categorization of exchanged communications. We have developed a preliminary set of categories for the types of design communication that could be exchanged: request for change, change notification, information request, information reply, etc. However, the study will help to improve and augment the classification. The communication tools require the user to select the type of communication being sent, choosing ‘other’ if no available choice is appropriate (and supplying a user-defined type). However, there is often a difference between what a subject says he or she is doing, and what is actually being done. Thus protocol analysis will be used to try to determine the intention which lies behind the sending of each communication, which should help identify possible classification schemas. Linguistic analysis will also be performed on the subjects’ verbal reports to determine patterns, significant and preferred words (such as action verbs [1]), and the like.

6 Conclusion

The Collaborative Design Studio has been designed to provide features and functions likely to prove valuable in any engineering environment, whether traditional or distributed. However, until user studies are undertaken, the value of the system as a whole and the relevance of specific features remain unproven. To guide further development, we have proposed several evaluation strategies, which will serve as a primary source of information for the evolution of the design environment.

Acknowledgements. This work was supported in part by National Science Foundation (NSF), Knowledge and Distributed Intelligence in the Information Age (KDI) Initiative Grant CISE/IIS-9873005 and Grants ENG/DMI-9713718 and CISE/CDA-9729827.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the other supporting government and corporate organizations.

References


