Building a Model for Augmented Design Documentation[1]

Ana Cristina Bicharra Garcia[2]
H. Craig Howard[3]

Abstract. Project-specific knowledge is the rationale behind the project data and specifications, including the design decisions that link elements of basic data, design data, project-specifications, domain knowledge, and general knowledge to explain the design. This information should be available in design documentation, but usually it is missing. The paper describes an approach for improving design documentation in which the computer acts as an apprentice to the designer to capture the rationale during the design process. The initial focus of the work is on HVAC (Heating, Ventilation, and Air Conditioning) design. We are using videotape analysis of design sessions along with structured interviews to develop a model of design rationale in this domain.

INTRODUCTION

The life cycle of the civil engineering facilities can be measured in decades, a long period during which the facility may undergo substantial changes. Moreover, most of the facilities are highly complex and require substantial time from initial planning to the final construction. The agencies involved in the realization of these facilities are also numerous, making information exchange and communication during the facility development life cycle very vital. Huge amounts of data and knowledge are produced and consumed during the various project phases. However, the final design documentation for the facility contains primarily drawings and numbers, with very little, if any, knowledge of planning, design and other decisions (and their bases) which went into making the facility the way it is. Generally, the documentation does not record the intermediary decisions nor the reasons behind those decisions. This information is what we call the *design rationale* behind a project or *project-specific knowledge* (Howard 89, 91). The lack of design rationale in project documentation hinders the understandability of projects and the transfer of expertise.

---

[1] Published in the Proceedings of *Artificial Intelligence in Design '91*, Edinburgh, U.K.

[2] Graduate Research Assistant, Department of Civil Engineering, Stanford University, Stanford, CA 94305-4020, USA; phone: (415) 725-2580; e-mail: bicharra@cive.stanford.edu.

[3] Assistant Professor, Department of Civil Engineering, Stanford University, Stanford, CA 94305-4020, USA; phone: (415) 723-5678; e-mail: hch@cive.stanford.edu.

The project documents are developed by the designers, but they are consulted by many different users. During the development of a project, the documentation is used as a communication medium among the different trades involved in the project. Plan checkers use the documentation to verify whether the project is in accordance with the codes and standards. Contractors use the documentation as a plan for realizing the artifact. Designers use the documentation to support redesign and also as a repository of experience.

In the era of automation, computers offer an attractive environment in which the designer can develop and document a project. A number of research projects have started in this direction. Some propose to build an integrated environment able to offer all the tools needed by the designer, including analysis programs and e-mail communication, in order to record all actions that took place in a specific design case, working as a notebook. They suggest that this historic data would allow design rationale reconstituting (Lakin 89). Other researchers are trying to create environments in which designers could explicitly enter their decisions and the rationale for them, but again in a noninterpreted form, e.g., (Conklin 88, Lee 91, Fischer 89). Still other researchers are interested in explaining all the decisions by going to first-principle knowledge, e.g., (Gruber 91, Baudin 89). On the other hand, some researchers emphasize the difficulty of capturing design rationale in a computer due to the complexity of the way people explain design, e.g., by using gestures (Tang 89).

As our approach for capturing the design rationale, we have developed a conceptual architecture for an intelligent design interface which acts as an *assistant* and *apprentice* to the designer. Whenever the designer proposes a design action that differs from the apprentice's expectations, the interface will ask for the designer for justifications to explain the differences. Subsequent queries for design rationale are answered using a combination of the interface's domain knowledge and the designer-supplied justifications. The conceptual architecture, to be implemented in a prototype called *ADD* (Augmented Design Documentation model), is described in Section 2.

The implementation of the ADD prototype will focus on the preliminary design of HVAC systems (Heating, Ventilation and Air Conditioning) for commercial office buildings. The task consists of choosing elements (such as ductwork, pipes, heaters, and chillers) and configuring them to provide the proper environmental control. By focusing on preliminary design, we are concentrating on the more significant decisions made early in the design process. In the case of HVAC system, the decisions being documented include the selection of a main HVAC system type, the equipment for providing cooling and heating, the duct and pipe selection, and finally the rough location of those components into the floor plan. Section 3 describes the process we are using to learn about project-specific knowledge in the HVAC domain by videotaping design sessions and interviewing designers. Section 4 discusses some of our initial observations from the interaction with designers.

THE ADD MODEL

The ADD (Augmented Design Documentation) model is based on the metaphor of having a computerized apprentice following the designer in the design process. The role of the designer is to create the specification of an artifact using his or her knowledge and to answer questions the apprentice might ask. The role of the apprentice is to follow the designer's decisions and ask whenever they are not clear. The apprentice should try to come with its own hypotheses in order to verify whether it is actually understanding the designer's reasoning. The objective is to minimally disturb the designer, but maximally understand what the designer is doing. By understanding a decision, we mean being able to check whether the decision is consistent with the knowledge that the apprentice contains about design in a specific domain.

When the designer and the apprentice propose the same design action, the apprentice records the decision with connections to its own rationale. All the decisions and their supporting rationale are stored in a case knowledge base. As soon as an inconsistency appears, the apprentice activates a knowledge acquisition agent to request more knowledge for its case knowledge base until it can reproduce the designer's decision. When another user wants to query the case knowledge base to understand more about a design, a case interpreter creates a browsable form that the user can examine to see the decisions and explore the associate knowledge. This information flow is depicted in Figure 1.

Figures 2 and 3 depict the two views of design rationale that ADD must support. The first view is the chronological progression, in which the design process is strictly a sequential set of decisions. Each decision helps to set up the scenario for the next one. The second view emphasizes the logical relationships between design rationale, the design decisions, and the design data. Every decision reflects or is caused by a change in the device form. A decision is a selection or elimination of alternatives that should be carried in the design process. The decision is made in a specific situation of the design process, what we call design scenario, using essentially experiential knowledge. The design scenario contains the history of the decisions already made, the design parameters already instantiated, and the decisions that have been deferred. Each alternative, direct or indirectly, represents a proposed change over the device form.

Figure 1: The ADD model.

Figure 2: Chronological View of Design Rationale

Figure 3: Logical View of Design Rationale

Figure 4 describes the apprentice process as implemented in ADD. The designer posts an action (i.e., a design decision) based on the current state of the design case. Meanwhile, ADD's hypothesis generator proposes its own action based on its design knowledge and on the current design scenario. The reconciler is responsible for comparing those design versions, and if they are the same, the reconciler updates the design by incorporating the hypothesis generator's knowledge as the explanation for the user's action. If the design decisions are different, the reconciler activates the acquisition mode to elicit knowledge that justifies the differences. The new knowledge is inserted, and a new cycle of hypothesis generation starts. This cycle continues until ADD is able to reproduce the user's decision.

We are assuming that if the system is able to reproduce the designer's actions, it is because the system has enough knowledge to explain those actions. We do not believe that the entire set of decisions that leads to a design can be matched by coincidence. We are also assuming that the user is able to explain the difference for its actions in an operational way. ADD offers a menu driven interface in which the user can use pre-existing concepts or create new concepts for explaining an action. The knowledge acquisition is done by presenting the user with the system's hypothesis and supporting rationale. The knowledge acquisition model includes two-way knowledge flow. The system is teaching the user about its knowledge and how it processes that knowledge, while the user is explaining his or her reasoning processes. Therefore, ADD does not have to start with a complete and powerful design knowledge base, but rather one that provides basis for dialogue with the designer.

Figure 4: Knowledge Flow in ADD

In order to develop a decision hypothesis, ADD relies on three sources of information: an HVAC system design knowledge base (domain KB), a code knowledge base (code KB), and a catalog data base. All three sources of information are biased towards the HVAC designer view; i.e., only information that is relevant for the HVAC system design will be represented. In the domain KB, we find the previously created building description and the evolving artifact specification. Both pieces of information define the scenario for the design task. This specification reflects the way designers look at the building by describing the components, the relations among the components, and the attributes characterizing each object. For example, a building is described composed of floors, which are composed of zones, which in turn are composed of rooms. Each of these objects needs an attribute to represent ceiling height. The domain KB will probably represent this information in a network of frames to attain the desired semantic expressiveness.

Also in the domain KB, we find the design constraints, including both internal and external constraints. Internal constraints are imposed by either the function or the behavior of the artifact in the environment such as "the dependency relation between number of ducts and ceiling space." External constraints are imposed by other design agents, such as the owner's constraint on cost. These constraints can be implemented using dependency relations among objects, as illustrated in SALT (Marcus 88) a knowledge acquisition system for configuring elevators. The domain KB will also contain heuristic rules to support the decision-making process. From our observations of design sessions, preliminary design decisions are heavily supported by heuristic reasoning.

Figure 5 illustrates a small portion of the domain KB's contents. Figure (5.a) illustrates the environment in which the design will be developed including the values for the artifact parameters, (5.b) illustrates the classification model of the HVAC system, (5.c) external and internal constraints are presented and finally (5.d) a design heuristic rule example.

The code KB contains the restrictions imposed by law; these constraints can not be relaxed by ADD. The catalog DB represents the available standard components. Our goal is to connect ADD with available manufacturers' databases, but initially we will create our own catalog data base. The components are described in terms of their attributes such as capacity and price. Since we are using frames to describe the artifact and environment, we will build the catalog using frames too.

In addition to domain knowledge, ADD needs to understand the user's needs in requesting design explanations. Five types of users access design documents: the original designer, another designer, the owner, the owner's representative, designers of other design trades and the plan checkers. Their needs are different, depending on their background knowledge and their goals in accessing the documents. These differences are observed in the objects and relations contained in the design explanations requested by each one of them. We are currently identifying those needs for designing the explanation.

METHODOLOGY FOR DEVELOPING ADD

Our starting point for implementing the ADD model is observing how designers develop designs for HVAC systems. We are trying to identify the important aspects of the design process and the rationale generated. The methodology we are using for this knowledge engineering work includes videotape analysis and structured interviews. We are working with three different companies (two contractors and one consulting firm). The contractors are known as cost oriented companies while the consulting companies are quality oriented. We have videotaped design sessions and meetings during the conceptual design stage. Those sessions are showing us which type of knowledge is important for justifying a design; i.e., which objects, attributes and relations should be present in design explanations in order to satisfy the goals of the users in accessing the documentation. We are looking for evidence of rationale normally used and provided by designers. The structured interviews with designers are providing additional background on the HVAC design process and the designers' strategies.

Videotape Analysis

We are videotaping four types of sessions:
1. The designer explains a previous project to the knowledge engineer.
2. Two designers develop the main concepts of the design.
3. The designer develops the design with interruptions by the knowledge engineer.
4. The designer develops the design in the presence of designer of other trades.

Initially, we viewed the tapes without having any expectations of what would be important. After some initial observations, we built an initial framework for analyzing those taped sessions. We tried to clarify how much information should be acquired from the tapes, the characteristics of that information, and the user interface needs. The main objective of this framework is to define design rationale for the domain of HVAC system design. The framework is divided into three levels: analysis by session, analysis by decision (within a session), and analysis by alternative (within a decision).

Figure 5: Domain Knowledge Base

1. Analysis by Session

o Type of session

Participants. The session participants (HVAC designers, other design specialists, owners, contractors, etc.) determine the type of interactions that take place within the session.

Session format. This parameter indicates the level of consciousness in the designer's reasoning process. For example, a designer working alone frequently takes big steps in the decision process without annotations. If the knowledge engineer interferes to ask for an explanation, the design process is a bit clearer, but the design process altered. If the designer is discussing a design with a peer from the same company, there are frequently reasoning jumps that are difficult for the uninitiated to follow. If the designer is explaining the project to other trades, the amount and type of information used are different than would be used in discussions with a peer. In summary, this parameter measures the changes in the type and granularity of information used during the session.

o Purpose.

The purpose of a design session is very subjective. It is related with the session output, which sometimes is indefinite. For the intention of this research, the purpose of the session is classified in three main categories: (1) generating an initial design concept, in which at least the type of HVAC system is selected; (2) identifying constraints that will necessitate changes in the initial design; and (3) producing the schematic design (how everything fits in the building). Purpose (1) and (3) fits the synthesis type of session, while (2) fits the analysis sessions.

o Session summary.

The summary includes a synopsis of what happened during the session, the design strategy, the amount of background knowledge used, the understandability of the explanation raised during the session, the jargon used, and the types of inference needed.

o Research output.

This parameter identifies the type and content of the design information conveyed in a specific design session, for example, the designer's strategy, the designer's heuristics, and the structure used to explain the design. By structure, we mean the objects and relations contained in the explanation.

o Designer behavior facing new knowledge or constraints.

The idea here is to record the effects of new factors as they are discovered. Does the designer simply check whether previous decisions are still valid or he will start again his analysis? How much does the new factor influence previous design? What type of adjustment is required?

o Interface needs.

Discussion media. This illustrates the environment required for design development during a given session. This parameter defines the tools necessary for integrating the design interface. For example, we observed that designers need to have a calculator available for them. We also noticed that they need to look at working drawings during the design process to understand the design.

Language needs. Does the discussion remain inside the domain? What type of common sense knowledge is invoked?

o Number of decisions taken and the names of those decisions.

These are links to the decision analysis records described below.

o Final comments.

2. Analysis by Decision

o Transcript of the design session portion that leads to the decision.

The detailed representation of the decision includes a transcript from the videotape of the designers' discussion.

o The decision.

This parameter describes the type of decisions that were taken, which may include no decision and eliminating or adding a set of alternatives.

o What triggers the decision?

This parameter identifies the type of knowledge that triggering the decision including a recognized pattern, the building data, the sequence of the decisions, heuristic rules, or a set of constraints that need to be satisfied. It helps to identify the domain design strategy.

o Sub-decisions.

This parameter points to the intermediate decisions that are needed for building the specific decision that is being analyzed.

o Related decisions.

This parameter links the current decision to other decisions that triggered this decision or are triggered by it. The relation may be chronological or logical (causal). This parameter together with the sub-decision parameter helps to organize the decision graph of a particular design case.

o What different types of knowledge are used for generating the decision?

The knowledge may include such things as designer heuristics, constraints from building codes, or solutions from previous similar cases. If the designers draw on their experiences in specific cases, we need to record the criteria for similarity and the methods for transferring solutions.

o Number of alternatives considered and names of those alternatives.

These are links to the alternative analysis records described below.

o Final comments.

3. Analysis by Alternative

o Transcript of the portion of the design session that leads to the identification of the alternative.

The detailed representation of the alternative includes a transcript from the videotape of the designers' analysis.

o What is the alternative?

This parameter describes the type of decisions that were taken including no decision and eliminating or adding a set of alternatives.

o What triggers the alternative?

This parameter identifies the type of knowledge triggering the decision including a recognized pattern, the building description, the natural order of the decisions, heuristic rules, a set of constraints that need to be satisfied. It helps to identify the domain design strategy.

o Sub-decisions.

This parameter points to the intermediate decisions that are needed for making this candidate decision feasible.

o Related decisions.

This parameter links the current decision to other decisions that triggered this decision or are triggered by it. The relation may be chronological or logical (causal). This parameter together with the sub-decision parameter helps to organize the decision graph of a particular design case.

o What is the change proposed by this alternative?

Does it add or relax constraints, add or relax preferences, cause the solution to converge to that of a previous design, etc.?

o Use of previous solutions.

If information from a previous solution is used, does that previous case provide positive or negative examples? Similarly, what triggered the recognition of similarity—matching elements in the building descriptions, previous experiences with other project participants (especially owners), etc.? What type of knowledge is used for selecting the similar case? How was the solution transferred?

o How does the design change proposed by this alternative influence other trades?

o Final comments.

Even though this videotape analysis consists the major technique for knowledge acquisition, it is not enough. Many design sessions bring new concepts and discussions that are only superficially mentioned at that time. In order to investigate in more details what actually happens we make use of another knowledge acquisition technique: the structured interviews.

Structured Interviews

The purpose of the structured interviews is to give a better understanding about the design sessions and also about the domain model. The building (where the HVAC system fits), the artifact (HVAC system), the design process and codes are examples of information that composes the HVAC system design domain.

In our first meetings, we elicited some general information about the field. Those interviews helped us to define the scope of our project as well as to provide us literature reference. The second round of interviews has helped us to clarify the design session videotapes. We have been meeting with the designers and asked for clarification over the design session. We propose a scenario to the designer and ask for his or her reactions. Then, we fix all variables, but one, in that scenario. We change the value of the free variable and observe the designer's reaction in order to get a deep understanding in the role of such variable.

Figure 6 presents a sample scenario and the reaction of one of our designers. First, the designer examined the building initial condition and proposed a conceptual HVAC system for it, in our case, a built-up floor by floor system. Maintaining all design parameters fixed except for the height constraint, we observed that the designer changed his choice from floor by floor to central system. Repeating the exercise with varying numbers of floors, we verified that the change goes from built-up system to package units. Based on those observations, we verified with the designer the degree to which each isolated variable influenced his decisions.

Figure 6: Sample Scenario from a Structured Interview

INITIAL OBSERVATIONS FROM THE DESIGN SESSIONS

The initial analyses of the design sessions emphasized the importance of having the floor plan visually represented as the medium for developing the HVAC system design. The designers use simple electronic calculators, rulers for taking measures from the floor plan, and parts catalogues to aid in component selection. In later stages of the design (but also during conceptual design phase), they run heating load analysis programs to make sure that their estimates of the cooling and heating loads are realistic. The numbers, the decisions, and the explanations get more precise depending on the size of the project and the client's demand.

The designers we studied rely heavily on their experience from previous cases. In every videotape session, the designers have recalled at one previous case. They recall those cases by either according project description similarities or by contact with the same project participants (especially owners). The negative examples (poor design alternatives) are recalled more often than the positive examples. Our initial explanation for this phenomenon is that the positive examples are compiled in the designer's generalized rules of thumb, while the bad experience remains primarily connected with the case.

Another observation is concerned with the different level of abstraction and also different type of explanation depending on the type of design session. When explaining for the knowledge engineer, designers explain in a professorial format. When talking among each other (i.e., between designers of the same trade), they speak at a much higher level. When explaining the design for other trades, they try to explain the design by addressing only the portion that affects the other designer.

A last observation on those design tapes is that designers do take notes of their design process. Those notes are very concise and rely upon the designer's power to recall the entire situation in which a decision was made. Those notes work as pointers for their memory.

FUTURE WORK

The first steps of our research were towards understanding the design knowledge capture problem, as described in this paper. We are proposing a descriptive model of the design process in a specific domain, HVAC system design, for supporting the documentation of the rationale in design cases. The next steps are towards the implementation of this computational model and further development of the ADD model.

ACKNOWLEDGE

We would like to thank Jim Bishop, Steve Poe, Linda Wilkins and Grant Wong for participating in our experiments. We also would like to thank Paul Chinowsky for his comments on drafts of this paper. This work has been supported by grants from the National Science Foundation (MSM-8958316) and the Stanford Center for Integrated Facility Engineering.

REFERENCES

Baudin, C., Sivard, C., and Zweben, M. (1989). A Model Based Approach to Design Rationale Conservation, technical report, No. M.S. 244-17, NASA AMES Research Center.

Conklin, J., and Begeman, M. L. (1988). gIBIS: A Hypertext Tool for Exploratory Policy Discussion, *Computer-Supported Cooperative Work,* Association for Computing Machinery, pp.140-152.

Fischer, G., McCall, R., and Morch, A.(1989). Design Environments for Constructive and Argumentative Design, *CHI'89*, Austin, pp.269-276.

Gruber, T. R., and Russell, D. M.(1991). Design knowledge and design rationale: A Framework for representation, capture, and use, to appear in the special issue of *Human Computer Interaction on Design Rationale*, Summer 1991.

Howard, H. C., Wang, J., Daube, F., and Rafiq, T.(1989). Applying Design-Dependent Knowledge in Structural Engineering Design, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 3(2): 111-124.

Howard, H. C. (1991). Project Specific Knowledge Bases in the AEC Industry to appear in the *Journal of Computing in Civil Engineering*, 5(1): 25-41.

Lakin, F.,Wambaugh, J., Leifer, L., Cannon, D., and Sivard, C. (1989). The electronic design notebook: performing medium and processing medium, in *The Visual Computer,* 5: 214-226.

Lee, J., and Lai K.(1991). What's in Design Rationale?, to appear in the special issue of *Human Computer Interaction on Design rationale*, Summer 1991.

Marcus, S. (1988). Salt: A Knowledge-Acquisition Tool for Prpose-and-Revise Systems, in *Automating Knowledge Acquisition*, Kluwer Academic Publishers, Boston, pp.81-123.

Tang, J. C. (1989). Listing, Drawing, and Gesturing in Design: A Study of the Use of Shared Workspaces by Design Teams, PhD thesis, Department of Mechanical Engineering at Stanford University, April 1989.