Improving Design and Documentation by Using Partially Automated Synthesis

A. CRISTINA BICHARRA GARCIA[1], H. CRAIG HOWARD[2] AND MARK J. STEFIK[3]

Departamento de Informatica, PUC-RIO, Rio de Janeiro, Brazil[1], Department of Civil Engineering, Stanford University, Stanford,CA[2] and Information Systems and Technology Laboratory, Xerox Palo Alto Research Center, Palo Alto, CA[3]

[published at: Artificial Intelligence in Engineering Design, Architecture and Manufacturing--AIEDAM 5(8)

Special Issue on Design Methodology-- 5(8)]

Abstract

One of the products of engineering, besides constructed artifacts, is design documentation. To understand how design participants use documentation, we interviewed designers and typical documentation users and also took protocols of them both creating and using design documentation. Our protocols were taken from realistic projects of preliminary design for heating, ventilation and air conditioning systems (HVAC). Our studies of document creation and use revealed three important issues: (1) Design participants not only look up design facts; they frequently access documents to obtain information about the *rationale* for design decisions; (2) The design rationale that they seek is often missing from the documents; (3) design requirements change frequently over a project life cycle so that design documents are often inconsistent and out-of-date. Recognizing these documentation issues in design practice, we developed a new approach in which documents are no longer static records, but rather interactive design models supporting a case. We demonstrated the feasibility of the approach by constructing a running system and testing it designers on realistic problems. We also analyze the costs and benefits of creating and using documentation of design rationale and of the active documents approach in particular for routine, preliminary design in domains where community practice is widely shared and largely standardized. The approach depends on the feasibility of creating a parametric design model for the design domain.

1. Introduction

Design is an ill-structured task (Simon, 1981) in which requirements are formulated while the activity develops. In this activity, the design of an artifact becomes concrete as the problem becomes better specified (Coyne, Rosenman, Radford, Balachandran and Gero, 1990). Designers generally follow an opportunistic problem-solving strategy (Guindon and Curtis, 1988) to find a solution that satisfies the requirements imposed by the outer environment. An important product of design is documentation. In engineering domains, documentation consists of blueprints, manufacturing plans, meeting notes and reports containing the main decisions' output.

This paper is divided in four sections. Section 2 describes the documentation problem in the context of building design. Section 3 is the focus of this paper, presenting our empirical field studies of designers at work. From these studies we derive some criteria bearing on the success of any system for supporting documentation, and also the assumptions that guided our creation of the active document approach. Section 4 describes the ADD (Active Design Document) system, which was based on our observations of the design and documentation processes. It also presents data on ADD's performance in meeting documentation needs of its users. Section 5 presents our concluding remarks.

2. Problem Description

Preliminary design refers to the earliest stages of design. This stage is important for two reasons: Many designs are rejected before they are realized, especially in competitive bidding situations. Thus in many areas, there is more work on the preliminary design stage than in later stages. Secondly, most of the costs of a project are determined by decisions made in preliminary design.

A building design starts with an owner's desire to construct a building. The owner allocates a budget and schedule for the construction of the building that performs a specific function (such as an office or hospital). In general the owner specifies the location and might already have the space available. The owner might add other requirements to the project's design, either general requirements such as the basic budget or specific ones such as the use of a specific material or equipment (Luth, Krawinkler and Law, 1991).

A Heating, Ventilation and Air Conditioning (HVAC) system is responsible for the climate and air quality within a building. Its design is developed along with the other building subsystems, such as the architectural and structural ones. The building subsystems interact and impose constraints on each other.

Designing an HVAC system is a routine task. By "routine" we mean that designers work on many similar projects using much of the same knowledge from one design to another. Routine design can sometimes be characterized as parameterized design. By "parameterized design" we mean that the space of possible designs is characterized by a limited number of parameters that designers have in mind. Constraints on those parameters come from general laws such as building codes and physical laws as well as case-specific requirements derived from problem specifications. A parameterized design process can be characterized as a search for values of parameters. In addition to the constraints among the parameters, the designers need to consider constraints imposed by the other design trades and the evaluation criteria governing the design. A typical HVAC design is composed of approximately 150 parameters that influence each other. HVAC system design is both routine and based on a well-understood engineering field with much first-principles knowledge, as well as many heuristic rules that guide design.

Design documents must serve different kinds of people with different interests when manipulating the document as Figure 1 illustrates. An architect might use a document to check the aesthetics of a design; a city inspector might check to see whether a design complies with the city building codes; an owner's representative might check to see if a design satisfies the owner's requirements. They have different views of the HVAC system domain and different interests. A paper document is static, so it cannot adapt to the different documentation user needs. Consequently, documentation users either accommodate their needs to what the document offers or contact the designers for more information.

Figure 1: Designers generate design documents to communicate with many different people.

HVAC system designers generate design documents to communicate their concepts. Especially during preliminary design, documents are used for 2-way communication. There are many tradeoffs in design between cost, function, and aesthetics; also many assumptions and preferences still need to be articulated. Thus, a preliminary design is tentative and subject to negotiation.

In addition to the difficulties related to the multiple users, there are other problems inherent to preliminary design. Design documents are permanently incomplete. The amount of information to potentially document is greater than what it is feasible to document. Thus, a document user may want to look up information about the design or about design alternatives that were not considered by the designer. Furthermore, since a preliminary design is not finished, parts of the document become out-of-date and inconsistent with other parts. Thus, the incompleteness issue cannot be addressed simply by organization and indexing of the documentation.

3. Field Studies of Document Creation and Use

ADD’s approach for design rationale is based on field studies observing preliminary design practice. The purpose of our studies was to identify the parametric model underlying the HVAC system domain, as well as to identify the issues that a documentation system needs to address.

Project	**Design Method**	**Description of the project**	**Session Duration (hours)**	**Participants**
A	document	Design Check List from two of the interviewed companies		HVAC system designers
B	design practice videotape 1	Design of the HVAC system of a 5-story office building in the Bay Area	1:30	two HVAC system designers
C	design practice videotape 2	Design of the HVAC system for an existing 1-story office building in the Bay Area	1:14	one HVAC system designer and 3 architects
D	design practice videotape 3	Remodeling the HVAC system of a library	1:15	on HVAC system designer
E	design practice videotape 4	remodeling the HVAC system of a medical facility given changes in the zone's function	0:40	two HVAC system designers
F	design practice videotape 5	explain a colleague's HVAC system design from existing design documents	0: 50	one HVAC system designer
G	design practice videotape 6	Design an HVAC system for a small office building	0:20	one HVAC system designer and the owner
H	design practice videotape 7	design an HVAC system for a restaurant	1:32	one HVAC system designer
I	design practice videotape 8	Review the HVAC system design of a building laboratory	1:15	owner's representative
	design review sessions	for each design case, designers explain further issues	between 1 and 2 hours for each design session	HVAC designer and knowledge engineer
	document	Project Proposal of several projects		
	document	Design Review Documents of several projects		
	document	Designers' reply to design reviews		

Table 1: *Field Data Summary*

This section presents the analysis of the data that led to the definition of the underlying parametric design model, the engineering decision-making model, and the documentation users' needs, as well as the assumptions of the active design document approach. Since the documentation problem has two sides (i.e., the acquisition and retrieval of design documents), our field studies focused on the two sides of the problem. Section 3.1 discusses the field studies on the design process, while Section 3.2 discusses the field studies on design reviews in which users look for design explanation in documents. Section 3.3 combines results from these studies, develops criteria for evaluating a more automated approach to documentation, and presents some hypotheses that bear on the success of the approach.

We used methods drawn from the practice of building knowledge systems (Stefik, 1995). We started with unstructured interviews to gain an understanding about the domain. Most of the empirical data was derived from design protocols (Ericsson & Simon, 1984), (Newell & Simon, 1972). As described in the following, we collected protocol data by videotaping designers at work on design tasks. Designers were encouraged to "think aloud" as they worked. Cameras were positioned to keep track of their access to documentation materials. This procedure records data about what designers do in terms of what they look at or mark up and what they are thinking in terms of what they say. Written records are then made of the visual and audio events of the design sessions. Table 1 summarizes the sets of data collected in this research. Each row indicates a project or set of interviews.

3.1. Field Studies on Documentation Creation

From our preliminary interviews, we believed that a parametric model would be suitable for describing the preliminary design process for HVAC systems. We used the protocol data to identify the parameters and constraints of a preliminary design model for HVAC systems.

Dr. Garcia videotaped designers developing the main concepts of medium to small HVAC systems to be reported in the proposal packages. She asked them to think aloud while designing the HVAC system. In this section we describe the experiment, the participants in this experiment and their tasks.

The designers' comments during design session and interviews were divided into discussion units for the purpose of analysis. A discussion unit is a piece of the design participant's discourse that brings a new issue or issue argument into the discussion.

Figure 2: *The decision dependency network (non-directed graph) in the domain of HVAC system design.*

After the data were recorded, events were encoded and logged. Our goal was to characterize the decisions made by the designer in the process, and then, to abtract out from the process a model of underlying parameters and constraints. Figure 2 summarizes parameters and constraints that were derived collectively by combining protocol data from multiple sessions. Each parameter in the dependency network is characterized by sets of possible values. Each constraint characterizes relations between values of the connected parameters.

Data Collection

Three different companies participated in this experiment. Two of them are construction companies; i.e., companies that generally are not involved in the conceptual stage of the design, working instead from the preliminary design until the construction stages. The third company is usually involved in the conceptual stage.

In all of the videotaped design sessions, the designers were experienced HVAC system engineers. They are project managers in their companies, with the majority of their work dedicated to preliminary design. Given an initial set of design specifications, they are responsible for generating a rough design of the artifact (*design*) and writing a proposal (*design document*) to bid the project. Thus, the project managers receive initial documentation containing design specifications and requirements to be satisfied, and then they produce another document with additional specifications.

Task

The task consisted of generating an HVAC system concept that satisfied a set of requirements. Generally, the designer receives a set of drawings containing information about the building that the HVAC system will serve. In addition to specifying the environment for the HVAC system, the building drawings identify the architectural and structural constraints over the HVAC system design. Beyond the drawings, a written specification with additional requirements including evaluation criteria comprises the initial document package received by the designer. The type and amount of information in this initial documentation package are not standard. However, as a minimum, the package describes the building concept, including the function of the entire building and the majority of its space allocations. Figure 3 exemplifies a problem statement as received by an HVAC system designer.

Section 15880

AIR DISTRIBUTION

Part 1 -General

Description

A. Provide complete system of air handling units, fans, other moving air equipment, air outlets, dampers, filters, humidifiers, access panels, ductwork, fittings' hangers, attachments and other equipment to provide complete and operable systems.

B. Provide contacts and appurtenances for interface with Building Control System.

Quality Assurance

A. Applicator (Erector) Qualifications: Each type of equipment required under this section shall be the products of a single manufacturer.

...

D. Reference Standards:

1: ASTM - American Society for Testing and Materials

  a. A626 - "Commercial Quality Zinc Coated Carbon Steel."

...

Part 2 -Products

Acceptable Manufacturers

A. Damper Hardware

  1. Ventfabrics, Inc.

  2. Ventlok

B. Centrifugal Fans

  1. Pace

  2. New York Blower

...

FLEXIBLE DUCT

A. Not Permitted

....

Figure 3: *Partial sample of an HVAC design problem description from specification.*

In general, the tools used for supporting the decisions in the preliminary design were quite simple, consisting of a calculator, a ruler, catalogs and tables of relations. The designers made sketches, calculated loads, selected systems, and calculated equipment cost estimates. They also accessed catalogs for selecting components. Generally, they run a computer package for checking their “guestimate” of building cooling loads. In addition, the designers relied strongly on the building drawings for designing and explaining the HVAC artifact. The building drawings provide the environment that houses the artifact. It contains the major constraints for the artifact. In addition, they used the drawings for augmenting their explanations. They get a drawing as the input (building drawings), and they will provide a drawing as the output (artifact schematic design).

Issues in Encoding Protocol Data

Within psychology, information processing psychology arose in reaction to behaviorism, which focused on stimulus-response models for explaining behavior. Analysis of verbal protocols is a source of information about mental processes that take place between input and output. The goal of using protocol analysis in information processing psychology is to gain insights about the nature of mental processing and mental machinery. When a psychologist uses protocol analysis to study a person playing chess, solving physics and algebra problems, or interpreting sentences, he may be interested in short term memory, the nature of the errors made, the storage and chunking of memory elements, or in the time that it takes to perform particular operations.

Protocol analysis uses search as its framework. Behavior is modeled as a search through a problem space, typically a state space with states and operators. During this search, the subject accumulates information about the problem situation. Each step is characterized as the application of an operator. Roughly, to analyze a protocol is to use the verbal data to build a cognitive simulation of problem-solving behavior in terms of states and operators. In our analysis of protocols for parameterized design, a typical step is the use of some constraint to determine a value for a design parameter.

Our collection and analysis of protocols derives from information processing psychology (Ericsson & Simon, 1984), but takes its goals and and methods from knowledge engineering. The following material in this section is adapted with permission from (Stefik, 1995).

The analysis of protocols raises several questions: How objective is the encoding of protocols? What demarks an intermediate step in the protocol? How should a set of operators be determined? How do we distinguish big steps from little steps? Does it matter how the little steps are aggregated into big steps and little steps, major and minor phases? How finely should the steps be recorded? Does the validity and usefulness of the analysis turn crucially on the vocabulary of operators? Can techniques be developed such that the encoding of protocols will be reliably repeatable across sessions, across subjects, or by different coders? Under what conditions do a coder's expectations bias the encoding of protocols? This pot pourri of questions raises several methodological issues which have been the subject of investigation in information processing psychology.

The questions are resolved in different ways for knowledge engineering than in psychology because the goals of psychology and knowledge engineering are different. The goal of protocol analysis for information processing psychology is to determine parameters of human cognitive performance. The goal of the analysis for knowledge engineering is to identify knowledge used for a task. The essential similarity between the two fields is that they both use (mainly) verbal data on cases to provide specific data about steps in a problem-solving process.

A crucial difference is that a knowledge engineering team views protocol analysis as merely an intermediate stage in its process of characterizing a task and identifying the knowledge that is used. Protocol analysis is one step along the way to building a workable, extendable and maintainable knowledge system that performs a particular knowledge service usefully and reliably in a setting. Knowledge engineering is less concerned about whether verbalization itself alters the thinking process. It is less concerned about whether verbalization alters the validity of the knowledge elicited, and it employs additional tests of the organized knowledge to detect and correct inaccuracies.

A knowledge system is developed and tested using a large set of cases. In knowledge engineering, protocol analysis provides both an indication of what knowledge is needed and how that knowledge is used in context. A single protocol may provide useful indications of what knowledge is used and what constitutes correct performance. However, it is never used by itself. A knowledge system must ultimately perform correctly on additional cases that will be supplied and evaluated. For example, in our study of HVAC design, we combined the results of multiple sessions with different designers.

The relationship between a knowledge engineer and a domain expert is quite different from that between an information processing psychologist and an experimental subject. After all, a domain expert knows more about the domain subject matter than the knowledge engineer and they are working together in partnership. This changes the nature of protocol analysis in several ways. For example, the burden of categorizing and establishing the operation definitions need not rest solely with the knowledge engineer; rather, it is determined collaboratively and incrementally by the domain experts and the knowledge engineers together. Because the knowledge base evolves as it is tested on multiple cases, there is less emphasis on getting a single consistent protocol analysis at the beginning. In short, the purpose of the interview is quite different. In our interpretation of protocol data as a dependency network, we used structured interviews as backup and to fill in gaps in the reasoning that were missing from the protocol data itself.

For these reasons, many of the issues that arise in protocol analysis for information processing psychology are much less important in knowledge systems. For example, key techniques for unbiased and reliable encoding in information processing psychology involve the use of multiple trained encoders who are unfamiliar with the hypotheses and goals of the research. Since the goal in knowledge systems (and this project) was to develop a robust model these considerations were not as important.

A good illustration of this issue is in the treatment of implicit parameters. In some cases, even central decisions depend on implicit parameters that are not brought to the discussion. Designers either assume a value for those parameters or do not even mention them. For example, to calculate the required shaft space for running ducts, designers need to decide about the amount of air to bring or exhaust (cubic feet per minute—CFM) and the velocity this air is going to circulate (feet per minute—FPM). CFM and FPM are constrained by building codes. In addition, those factors indirectly influence the level of noise in the building. However, generally these parameters are not even mentioned in the discussion.

Constraints	Criteria	Goals
Environment Constraints: (Architect + structural constraints) building dimensions building shape building orientation (sun impact) glass information at each exposure number of floors building major function building available mechanical equipment area equipment location building shaft space building shaft location floor to floor height ceiling space position of main structural elements size of main structural elements building location schedule Artifact requirements: desired inside temperature desired inside humidity desired inside cleanliness need of back-up system durability limitations maintenance limitations Owner’s requirements: first cost limitations operating cost limitations tenant cost limitations owner's cost limitations owner's preference	minimum cost maximum efficiency maximum performance minimum maintenance maximum centralization maximum independence minimum noise	o Specifying the building zoning according to thermal, use and schedule similarities o Generation systems for cooling heating ventilation o Defining fuel for cooling heating ventilation o Defining Distribution systems for cooling heating ventilation o Defining terminal systems for cooling heating ventilation o Defining control systems for cooling heating ventilation o Specifying equipment o Specifying equipment locations

Table 2: *The structure of the HVAC system design problem.*

In general, we found that parameters and constraints used by HVAC designers did not have a great deal of variation across designers. This property is fundamental to our observation that HVAC design is routine; indeed that its practice is highly standardized. Table 2 shows some of the “routine” constraints, criteria, and goals that characterize designs for HVAC systems.

3.2. Field Studies on Documentation Retrieval

The second part of the experiment for defining design rationale in the domain of HVAC system design consisted of videotaping documentation users accessing design documents. In addition to the videotapes, official documents requesting explanation and interviews with different documentation users were used as the data sources for the analysis. Seven different projects were used in this experiment. They vary in their complexity of the designs (from small office building in the Bay area to large multi-purposed buildings) and the number of participants (from one to four participants).

From our preliminary study of the domain including interviews and our case studies, we observed that in addition to generating the design, the project manager is responsible for explaining it during and after the preliminary design to other design trades participating in the project. Table 3 illustrates the explanation needs of users of preliminary design documents. The document is manipulated by new HVAC designers who join the project later. They need to understand what has been designed so far and why in order to continue the design. Other HVAC designers also need to understand the entire design if they have to redesign the device. The other design professionals participating in the building design (e.g., architects and structural engineers) need to check whether the modifications they propose invalidate the HVAC system design. They also need to check whether changes imposed by the HVAC system designer affect their portion of the building. Owners check to see if the design is in accordance with their basic needs with respect to budget and quality. Owner representatives need to fully understand the design in order to verify whether their client's needs will actually be achieved. As part of the approval process, inspectors check to see that the building code was not violated. Finally, practitioners need to understand what was designed in order to detail it; however, they seldom need to understand the reasons for the design. Consequently, we categorized their needs as checking.

Information Requester	Role	Information Needs	Design Phase
Other HVAC designer	Continue design Re-design	Understand Understand	Preliminary Design Preliminary Design
Other Designer Trades	Accept design	Check	Preliminary Design
Owner Representative	Approve design	Understand	Preliminary Design
City Inspector	Approve design	Check	Preliminary Design
Practitioner	Detail design	Check	Design Development
Owner	Approve design	Check	Preliminary Design

Table 3: *Explanation consumers, their roles and needs.*

One of the most important data sources for studying the explanation needs of documentation users was the review sheet used by owner representatives to communicate with designers. A review sheet is a communication document where owner's representatives (experienced mechanical engineers) write their comments about the design idea proposed in the design documents. The review sheet contains requests for more explanation and suggestions for changes. Figure 4 presents a sample review sheet. In this document, an owner's representative requests design clarification and rationale for the designer's decisions (first column of the document). Designers answer the questions in the second column. The interviews with documentation users emphasized the importance and representativeness of review sheet in outlining the explanation needs.

Figure 4: *Official documents used for requesting design rationalization.*

The statistical unit adopted for this part of the experiment was “the request unit.” A request unit is either a question or an answer for a question concerning the design.

3.2.1. Method

The goal of this part of the experiment was to observe how document users used design documentation. More specifically, we wanted to identify the types of information that users seek when they access design documents. In some ways, having access to documents created by a designer is a substitute for having access to a designer. In analyzing the information sought by users, we did not want be limited to shortcomings of current paper documents, so we also taped sessions where users queried designers directly.

The participants’ task was to understand the design based on design documents and to formulate questions about what they did not understand or did not agree with in the HVAC system design. On the other hand, the HVAC system designer replies to the questions either by writing down the explanation and sending to the information requester or by answering the questions in a meeting. In this section we describe the videotapes used as data, the participants in the experiments, and the results from the data.

Description of the Experiment

The data used for describing the documentation consisted of a subset of the videotapes and design review documents described in Table 1 that dealt specifically with documentation retrieval and explanation. This information spanned eight different building projects, which are summarized in Table 4 along with the type of user who was requesting the explanation. Review sheets of four different projects (projects 2, 4, 7 and 8) represented the most powerful instrument for the analysis of explanation, because these review sheets explicitly request explanations, but other sources were also valuable, as described below.

Videotape 1 (project B from table 1 and table 4) recorded architects querying the designer to understand the HVAC system design. The videotape was also used for modeling the design process. A mechanical engineer and four architects took part in this experiment. The mechanical engineer received an initial specification for the design and developed a preliminary design for the office building being studied. The architects raised questions during the design to understand the concept being proposed and to make sure that the HVAC system design concept did not conflict with the architectural design.

Videotape 2 (project C from table 1 and table 4) recorded an HVAC system designer querying another HVAC system designer about the project during the design process. Even though only one designer was responsible for the project, they worked together for developing the design.

Videotape 3 (project D from table 1 and table 4) recorded a designer trying to understand an HVAC system design made five years earlier by another designer. The owner opted not to build the office building at that time, but wanted to restart the project. Since the original designer was no longer in the company, the new designer needed to complete and approve the final design made previously.

We also considered observations we made during the design of a small laboratory. The HVAC system designer was proposing and discussing with the owner the feasibility of the project she was proposing (project X3 from table 4).

Project	Project type	Explanation Requester
Project B	medium scale office building	architects
Project X1	university research building	owner representative
Project C	medium scale office building	HVAC designer
Project X2	university research building	owner representative
Project D	medium scale office building	HVAC designer
Project X3	small laboratory	owner
Project X4	university research building	owner representative
Project X5	university library	owner representative

Table 4: *Projects used in the second experiment.*

Participants

We studied six different types of users accessing design documents to retrieve design information: owners, owners' representatives, designers, practitioners, city inspectors, and designers of other building design trades. These users queried the designer about the project either asking the questions in a meeting or sending formal documents requesting explanations (review sheet documents). Records of design sessions with multiple participants and the review sheet documents provide the core information for defining the documentation users needs. We also examined the check list used by the Santa Clara city inspectors, but little information contained in the check list was concerned with the preliminary design stage.

3.2.2. Results

We observed that documentation users access design documents with different purposes, such as understanding, verifying and agreeing with the design. Owner representatives review the design to make sure that the project will satisfy the owner's goals. They are generally experienced mechanical engineers that were hired to supervise the project. In small projects, owners don't hire engineers to represent them. They supervise the project directly. City inspectors are responsible for approving the project. Their goal is to verify whether the design complies with the city code regulations. Designers joining the project late use the design documents as a tool for understanding the project. Typically, they are junior mechanical engineers responsible for detailing the design. They need to understand the concepts that they are detailing. Table 5 illustrates the different users and their goals when accessing design documents.

Goals Users	**Understanding**	**Verification**	**Agreement**
Owners	X		
Owner Representative	X	X	X
Designers	X	X	X
Practitioner	X		
City Inspector		X	
Other trades	X		

Table 5: *Types of documentation users and their goals in accessing design documents.*

The users' intents when manipulating design documents influence the type of questions they formulate and the types of answers they are looking for. Users trying to understand the project ("understanding" goals) asks about the device values (WHAT questions) and about the design process (HOW and WHY questions). Users verifying the project are only interested in checking the device values (WHAT questions). On the other hand, when the documentation users have to agree with the project by using their own experience, they need to understand the device, the design process, and the designer's experience (WHAT, WHY, WHAT-IF, and WHY-NOT questions).

These types of questions can be classified in three groups: clarification (WHAT and HOW), justification (WHY), and divergence (WHY-NOT). The subjects of the questions include form, function and behavior of the HVAC system artifact as well as the requirements and criteria that guide the designer. Table 6 contains classified examples of the situations observed. The classification may be unclear because the examples are presented out of their context and because communication does not always fit into neatly categorizable bites. Table 7 presents the frequency in which each combination appears in the rationale requests.

	Clarification	**Justification**	**Divergence**
Form	Steam & Chilled Water utilities are missing and connection points to these utilities need to be determined. Please define the points of utilities connection and some specific routings.	Why are we installing space fan carbon filterization?	The mechanical rooms are scattered and will be difficult to maintain. Try to consolidate if possible.
Function	The area over the XXX loading dock is covered. How do you plan to exhaust truck fumes from that area? In addition, how do you prevent fumes from entering the building?	(Not found)	(Not found)
Behavior	How will the vacuum pumps be exhausted?	Exhaust stacks are same height as building. Why?	(Not found)
Requirements	Please, clarify the type of occupancy rating of the building (B2, etc.).	Is air quality a problem?	Why are you using 15 air changes per hour? Has it been calculated? (12 is fairly normal)

Table 6: *Examples of types and subject of design questions retrieved from review sheet documents.*

	Clarification	Justification	Divergence
Form	43	5	7
Function	2	0	0
Behavior	9	2	1
Requirements	9		3

Table 7: *Frequency of each type of explanation request considering only the review sheet documents.*

3.3. Issues for Active Design Documents

In this section, we step back from the field studies to discuss implications for a partially-automated system for design and documentation. We draw on material from the interviews and more formal studies to lay out evaluation criteria crucial to the success and utility of *any* approach to automating documentation, and also present assumptions specific to the ADD approach.

3.3.1. General Issues in Automating Design and Documentation

There are several considerations about the creation and use of documentation that bear on any approach to creating documentation for preliminary design.

One of the most striking observations is that design participants not only look up design facts; they frequently access documents to obtain information about the *rationale* for design decisions. This shows up in the field studies of document use in the many HOW and WHY questions. The decisions in HVAC design, like decisions in most other areas of design, are richly interconnected. If you change the building codes, or use materials with different properties, or change the assumptions about weather -- the designs will come out differently. Furthermore, if you keep those factors constant but change otherwise arbitrary decisions in one part of the design, there are often implications for other parts of the design. Indeed, the interactions of such decisions are reflected in the interconnections of the parameter network. Links in that network specifically represent constraints between choices for values of parameters.

In preliminary design, documents are created as part of a competitive bidding situation. A designer must consider the probability of obtaining project funds (“winning the contract”) when investing time to create documentation. Thus, there is a substantial incentive to not overinvest in documentation. Thus, the amount of information that can be documented is extremely sensitive to the cost of documentation. For this reason, it is not surprising that the open-ended design rationale that document users seek (as indicated by our field studies) is often missing from documents.

Finally, the preliminary stage of design is the least stable stage of design. During this stage a major part of the designer’s task is to clarify the owner’s goals and balance his needs. The clarification is reflected in revised preferences and evaluation criteria, and takes place in the context of partial designs. Thus, design documents are often in a stage of incomplete revision. Sometimes the designer does not update all portions of the documentation when a change is made. Even beyond the preliminary design phase, design requirements change over a project life cycle so that design documents are often inconsistent and out-of-date.

3.3.2. Assumptions Specific to the ADD Approach

On the basis of the field studies, we created a computer system (described in the next section) based on active documentation. The logic of this approach was based on several assumptions about the nature of documentation and the HVAC design process. In this section, we lay out some of the most important assumptions in terms of “hypotheses.” We call these hypotheses in the sense that they could be true or false in the situation. Testing them is possible by evaluating the performance of ADD on cases. Furthermore, we consider the hypotheses central to how ADD works and to the applicability of the approach to other domains.

o "Rationale Predictability" Hypothesis: For ordinary HVAC system design, a computational predictor of design rationale (reasoning from data about the design situation and the design decision) can usually predict the design choices and their rationale. Such a predictor diminishes the documentation load of designers.

This hypothesis is about the performance and utility of a parametric model for HVAC design. The ADD system is given a parameter network, the constraints from the problem statement for the case, the evaluation criteria from the problem statement, and constraints and defaults that hold for all cases such as building codes and physical laws. The hypothesis says that in the context of working on a design problem, these givens can be used to predict design decisions and that the predictions are usually right. When the system makes the same choices as a designer, it can record the "rationale" for the decision, without requiring the designer to enter that rationale. If the system's predictions match the designer's most of the time, then the designer's work in documenting decisions is greatly reduced. This hypothesis can be tested by directly counting the number of times that the system's decisions match the designer's decisions.

- "Explanation Completeness" Hypothesis: The same model that generates design should be able to explain it.

This hypothesis says that the model used by designers is also useful for people using documentation. For example, the rationale used in making design decisions should "make sense" to people who use documentation. The application of this hypothesis is broader than just the specific design decisions of a designer on a particular case. In its fullest sense, it means that the parameter network can reliably answer "WHAT-IF" questions about options that the designer did not explicitly consider. This hypothesis can be tested by checking the efficacy of answers produced by a system by people who use design documentation. The checking can be tested both on questions considered by designers and on questions that they did not consider.

- "Explanation Diversity" Hypothesis: Although all explanations can be derived from the same model that originated the design, explanations vary according to the perspective and goals of the person requesting it. In other words, it should be possible to get different views of the same information (relevance).

This hypothesis reflects our concern that explanation needs vary with the goal and perspectives of the documentation users (understanding, verification and agreement). It says that the parameter model and the generation of explanations from it must be adequate to cover the diversity of document users. It can be tested by checking the efficacy of answer for populations of different kinds of users.

- "Explanation Kernel" Hypothesis: A small set of parameterized document types suffices to provide form and context for most document queries.

This hypothesis reflects our approach to answering queries. Rather than basing the information-finding and delivering capabilities on a query language and retrieval system, we base it on a stock of report templates that cover that main kinds of questions that people have. Thus a report bundles together a selected set of related information. Our hypothesis is that a small number of report types is adequate to cover most of the needs of document users. Again, this hypothesis is tested simply by checking the performance of the system with users, noting whether the reports are judged by the users as adequately covering their needs.

In the course of this research, we developed a software prototype that performed satisfactorily as described in the following sections. The ADD system is an existence proof of feasibility of the active design document approach, and the hypotheses correspond to necessary “assumptions” for the feasibility and utility of the method.

Roughly, our approach to active design documentation is intended for "routine, preliminary design." A more precise way of characterizing its applicability is in terms of the satisfaction of these hypotheses. However, we suspect that these hypotheses are often satisfied by routine, preliminary design tasks.

4. A Model for Building and Retrieving Active Documents

We propose an approach to design documentation based on an active document—active design document (ADD). The new document contains the design model used to develop a project by a designer. Therefore, the new document is case and designer specific.

4.1. The Architecture of ADD

We created and implemented an architecture, shown in Figure 5 (Garcia, Howard, and Stefik 1993) This architecture includes:

Figure 5: *An architecture for implementing the Active Design Documentation model. Even though documentation acquisition and retrieval are two different processes, the architecture for implementing both models shares many modules. In brief the Anticipator generates the design, the Reconciler reconcile’s ADD’s choices with the designer’s, and the Rationale Generator creates explanatory reports.*

- Reasoning Components. These are responsible for generating design decisions, comparing these decisions with the designer's decisions, preparing design reports and controlling the documentation process (the Anticipator, Reconciler, Rationale Generator and Controller respectively);
- Design Knowledge Base. This contains knowledge about the HVAC system docmain and knowledge about a specific case;
- Interfaces for creating documentation. These are active only when creating a design document. These interfaces allow designers to develop their projects and to adjust ADD's design model (the design and justification interfaces, respectively);
- Interfaces for Retrieving documentation. These allow documentation users to query and question the design (the Explanation interface).

The role of the Anticipator is to predict a value for a decision topic given by the designer considering the current state of the design and the active requirements. To make a prediction, the Anticipator uses the domain knowledge base (the parametric design and engineering decision-making models) and information about the specific design case. It is a constraint-based reasoner; i.e., given some constraints and a set of evaluation criteria, it is able to generate and analyze alternatives and propose a set of solutions. Figure 6 illustrates the Anticipator procedure to decide the value for a parameter.

The Anticipator proposes a set of parameter values that need to be compared to the value proposed by the user. The module responsible for this comparison is the Reconciler. In the best and typical cases, ADD agrees with user and thus determines that it understands the rationale for the decision. If there is a mismatch, the Reconciler diagnoses the type of match or mismatch that occurs between the designer's and ADD's proposed values.

Whenever a mismatch is diagnosed, the rationale generator and the justification interface are activated. The *Rationale Generator* is activated to prepare ADD's rationale for its expectation. The Justification Interface is activated to elicit changes to ADD's model from designers.

Figure 6: *A simplified version of the Anticipator's procedure.*

The Knowledge Elicitor is activated whenever a mismatch is diagnosed. The Knowledge Elicitor module works closely with the Justification Interface. This module interprets the information provided by the designer. The elicitation is guided by the user triggers a procedure to change requirements, change design constraints, or change design criteria.

The Controller supports the overall interaction cycle. The Controller defines ADD's sequence of actions, but not the order of designer actions. It is often in idle mode. As soon as the designer updates the design case by proposing a new parameter to be evaluated, the Controller activates the Anticipator to generate a expectation for the parameter value. Then, the Controller sends ADD's expectation and the user's value to be evaluated by the Reconciler. If the Reconciler diagnoses a match between the values, the Controller updates the Design Case and returns to its idle mode. Otherwise, it activates the Justification Interface to acquire more information for ADD's design model. The Controller also propagates the changes to any parameter influenced by those changes checking whether the changed parameters still comply with the values proposed previously proposed by the user.

4.2. Examples of Document Creation

Designers can propose values for design parameters in any order. Even when parameter dependencies require a certain order of actions, ADD does not impose this order on designers. ADD creates assumptions for the dependent parameters to avoid imposing decision ordering. Figure 7 gives an example of the justification user interface. The *Justification User Interface* starts working as soon the Reconciler detects a conflict between the designer's and the Anticipator's decisions. A designer can either accept ADD's proposed value and explanation, force his/her own solution, or go to the acquisition mode to adjust ADD's model.

Figure 7: ADD's Justification Interface: the lower right window presents the options for changing ADD's design model while the remaining windows present ADD's rationale for its expectation (evaluation of alternative values for a parameter considering active constraints and criteria, dependency network containg the parameter being discussed and history of the decisions).

Using a set of cascading menus, a designer specifies whether to change the proposed value (Change Parameter Value option), a value of a parameter previously decided (Change Previous Decision option), or the concepts influencing the problematic parameter (Make Changes option). If the designer selects the Make Changes option, another menu pops up to specify the type of change. At the end of the cascading menu, the designer selects among changes in building description data, device information, constraints definitions, or evaluation criteria.

4.3. Examples of Document Retrieval.

The appearance of ADD's document retrieval interfaces are modeled after current engineering documents. This approach to supporting the retrieval of information is in contrast with the conventional use of query languages. Facts are not found by describing appropriate names and indices to a query language. Rather, they are presented in a regular integrated way in the reports that make up the design documentation. To find information, one requests the appropriate kind of report. In this way, information is always presented in a context, and pieces of information that are generally used together are reported together.

The interface offers a menu with the options ADD is able to process. As Figure 8 illustrates, ADD enables a user to change design parameters, view the owner's design specification, print the design proposal, review the steps in the design process, and ask questions about the design.

Figure 8: Explanation Options.

The Specification, Device Summary and Proposal reports each contain a definition of the set of parameters that need to be retrieved to formulate the report.

The Design History report presents the user with the sequence of decisions related to design parameters made during design. Consequently, this report retrieves the decisions and the chronological relationships among them. The report provides a chronological list of decisions.

The parameter-related reports (WHY-value, WHY-NOT-value and IMPACT reports) require a more complex data retrieval. They require the retrieval of the parameter value, the alternative values, the set of active constraints and criteria with their evaluation, and the design model supporting the final parameter value.

Sometimes information needs to be generated instead of just retrieved. This is the case for instance of the WHY-NOT-value and Impact reports. In these cases, the Controller invokes the Anticipator to provide a value and an evaluation for the parameters. If the design specification does not change, the Anticipator's task consists of retrieving the evaluation it did during design. Otherwise, the Anticipator recalculates the parameter value. User demands for alternative comparisons cause the Rationale Generator to evaluate each alternative and select the data that make the alternatives differ from each other.

Once the data for the reports are available, the Rationale Generator's effort orients towards filtering the information to be presented to documentation users. ADD considers two types of explanation filters: breadth and design view filters. The breadth filtering is defined by user selection of the explanation emphasis. The breadth filtering determines the amount of information ADD displays to the user for a given design parameter (as illustrated in Figure 8).

The main role of design documents in preliminary design is to allow a 2-way dialog between the designer and documentation users. HVAC designers need to understand the design requirements requested by the other building design participants including the owners and the architects. Since designers do not have perfect knowledge of these requirements, they make guesses about the requirements during design. In addition, during the preliminary stage of the design requirements often change.

Figure 8: The data mask caused by the breadth filter. In this example only information on first cost and operating cost of parameter alternative values are displayed to users.

ADD's dynamic document allows documentation users to discover the designer's assumptions, as well as to support some exploration of design space even when designers are not available. ADD's available design model can simulate what would be the design under different circumstances. Consequently, the design choices move to be shared by designers and clients.

Whenever a change in the building specification or in the design requirements is proposed by a documentation user, the Rationale Generator retrieves all parameters influenced by the changes. For each of these parameters, the Controller invokes the Anticipator to obtain an expectation considering the new design conditions. The Reconciler is activated to check the match between the old and new parameter values. If the new specification produces a different parameter value, the Controller records and propagates the changes to the influenced parameters. At the end, the Rationale Generator contains a list of parameters that need to be changed to adjust to the new design specifications. The changed parameters correspond to the impact on the HVAC system design given changes in the design specification. As soon as the impacts are calculated, the Rationale Generator returns the design to its original specification.

The same process occurs when the change affects the HVAC system design. In this case, the user is probably interested in checking the impact on the other design trades (or even in other aspects of the HVAC system design) if a HVAC system design parameter changes. The Rationale Generator receives the new HVAC parameter value and evaluates it in terms of the active constraints and criteria. Consequently, it checks the local impact caused by the change in the design specification, such as a violation of an architectural constraint. In addition, it forces the new value and propagates it to the set of influenced parameters. At the end, the Rationale Generator reports the local evaluation of the change and a list of other HVAC parameters affected.

4.4. Evaluating ADD's Performance

To determine the viability of the ADD approach, we developed a pilot study using ADD to build, revise and use an active document for a realistic problem. Our studies have helped us to understand how an active document can impact the cost of creating and using documentation.

We kept statistics on ADD's performance. In addition to the statistics we interviewed designers and documentation users to understand their evaluations of it.

4.4.1. Pilot Study in Creating Documentation

We selected two experienced designers from different HVAC system companies to take part in our experiments. Both have more than 10 years of professional experience in generating HVAC system designs. They had no previous experience with ADD or with the problem case. They were asked to develop a design proposal (preliminary design document) for a given case.

The problem consisted of developing and documenting a design for a 5-story office building located near San Francisco using ADD. This realistic project was not considered for developing ADD's initial KB.

Each designer was given an initial set of specifications—including the building blue prints, design criteria, the list of design participants, and the owner's requirements for the design. They were asked to prepare a preliminary design meeting these specifications. The preliminary design of an HVAC system involves the instantiation of about 150 parameters. They explicitly decided among 15 and 20 parameters. All the other parameter instantiation were considered implicit decisions, but they also checked their values at the end of the session. There was no imposed order for their decisions nor actions. The sessions were videotaped.

The designers interacted with ADD though one of us. They asked for information about the case, selected parameters to be nstantiated as well as provided values for those parameters. We did not provide any additional verbal about the project. We wanted to verify whether the information shown by ADD was suficient.

The sessions lasted about 2 hours each. They were videotaped for further analysis. In addition to the material collected observing designers interact with ADD to develop projects, we interview them to verify the usefulness of a tool like ADD, the adequacy of the approach to support design and documentation, and the need of such tool to assist the documentation. process.

We analyzed the data in the protocols to verify the adequacy of the architecture to allow building an active document and the impact of an active document on the cost of creating documentation.

We recorded the frequency in which each module was activated in each session as well as the number of right expectations generated by ADD. The prupose was to measure the percentage of automatic documentation as a measure of the time saved.

The results of this pilot study are shown in Table 8. The numbers in the column, except for the hit ratio and informal evaluation, indicates the number of times the module was activated for a given session.

According to table 8, to document the portion of design developed during session 1, designers select values for 17 (number of parameters verified by the Reconciler) minus 4 (number of parameters that triggered the Knowledge Elicitor—elicitation cycle), while ADD decides the values of 74 parameters. Consequently, there is a gain of 61 parameters documented automatically. The Knowledge Elictor was activated whenever ADD’s expected value was not similar to designer’s proposed value; i.e., ADD creates wrong expectations for 4 times in session 1. The fraction of the parameters correctly and automatically documented is what we call the anticipation hit ratio.

The initial indications are that the architecture supported the documentation task and that the success of active documents are related with the rate of right anticipation.

		Session 1	Session 2
	User	Project Manager A	Project Manager B
Architecture Use	Design Interface	33	13
	Anticipator	74	59
	Reconciler	17	12
	Knowledge Elicitor	4	6
	Justification Interface	10	20
	Anticipation Hit Ratio	0.96	0.95
	Informal Evaluation	Excellent	Excellent

***Table 8**: Pilot study results for creating documentation.*

4.4.2 Checking the Rationale Predictability Hypothesis

We now consider our hypotheses creation of documentation.

- "Rationale Predictability" Hypothesis: For ordinary HVAC system design, a computational predictor of design rationale (reasoning from data about the design situation and the design decision) can usually predict the design choices and their rationale. Such a predictor diminishes the documentation load of designers.

If this hypothesis is correct, then ADD should predict the designer's decision in a very high percentage of times. The data from Table 8 shows that ADD predicted the correct decision roughly 95% of the time -- the anticipator's "hit ratio." Equation 1 shows us the importance of the high percentage of right expectation for the success of the approach in terms of documentation time.

$$T_e = T_h * r + T_i * (1 - r) \quad \text{(Equation 1)}$$

where:

T_e is the time to enter rationale;

T_h is the time to document a parameter automatically;

T_i is the interaction time—the time spent adjusting ADD's design model to match the user's decision; and

r is the anticipator's hit ratio.

Equation 1 tells us that the expected design time depends on the anticipator's hit ratio. If the knowledge base is tuned to the user's practice, then *r* is close to 1. In such cases, ADD's approach to documenting additional parameters imposes very little overhead on the designer. In our test cases, even though the initial design model was not tuned, the anticipation hit ratio was nearly 1.0. This suggests that the domain is mature and that the strategies used for designers do not have a big variance.

4.5. Pilot Study in Retrieving Documentation

We also set up two test cases to evaluate ADD's performance in delivering rationale. We selected two natural users of HVAC system documents: an owner's representative and a mechanical engineer. The owner's representative had more than 15 years of experience analyzing design documents and making sure that the owner's requirements were satisfied by a project. The other user was a tenured university professor that teaches the design of HVAC systems. They had no previous experience with ADD or with the problem case.

The problem consisted in understanding and approving the design proposal developed by the previous experiment (documentation acquisition).

One of us served as the interface with the system. Our role was to input the users questions to ADD. Both users easily verbalized their questions about the design and analyzed the answers presented by the system. We did not add any new information to that displyed by ADD. All information about the project was inside ADD.

There was no time restriction on the interaction, however both took 1 to 1 1/2 hours interacting with the system. The participants were videotaped. The videotapes were the basis for our analysis.

They reported that the problem was very usual, however they took longer to perform it. They mentioned that in their work they would be interrupted often. They also commented that they missed the blue prints at first till they realized that the blue prints were also available in the tool. They were surprised with the potential of the tool, especially the ability to check impacts on a design given changes in the requirements.

We analyzed the data in the videotapes produced by the two experiments. We were looking for

- the adequacy of the architecture to provide answers to users' questions, and
- the adequacy of the answers to satisfy documentation user's needs.

Table 9 presents the number of times each module was activated to generate design explanations. All the modules were activated during both sessions. This fact suggests that all modules were necessary. Even in a retrieval session, the Anticipator was activated, illustrating the fact that explanations are generated and not just retrieved from active documents. During session 1, the user retrieved design facts, such as initial requirements, and decisions history. He questioned the value of 4 parameters and verified the value of other 6 parameters. In both sessions they changed specifications to check the design response to those changes.

		Session 1	*Session 2*
	User	Project Manager A	Mechanical Engineer
Architecture Use	Explanation Interface	33	13
	Anticipator	74	59
	Reconciler	17	12
	Rationale Generator	4	6
	Answer Acceptance	0.9	1
	Informal Evaluation	Excellent	Excellent

***Table 9**: Pilot study results for retrieving documentation.*

This results of this pilot study are shown in table 9. ADD was able to answer all of their questions, except for the ones related to parameters outside its design model or related to parameters not mentioned nor affecting the ones mentioned by the designer during the development of the project.

4.6. Checking the Explanation Hypotheses

We now consider the evidence from the testing of ADD for our hypotheses related to explanation and document retrieval.

- "Explanation Completeness" Hypothesis: The same model that generates design should be able to explain it.

The very architecture of ADD uses the same knowledge structures for designers and document users. The shading in Figure 9 shows how different portions of the parameter network. Different shadings correspond to parameters explicitly considered by the designer, parameters whose values were requested in explanations by $user_1$, and parameters whose values were requested by $user_2$. This suggests the satisfaction of the hypothesis in the strong sense, in that the model was usable even for questions not considered by the designer.

Figure 9: *AMEBA diagram illustrating the portion of the design model being discussed. Nodes in this figure correspond to design parameters in the HVAC parametric model. Links represent dependencies, such as derivation paths and constraints. See the legend for the interpretation of the shaded regions.*

- "Explanation Diversity" Hypothesis: Although all explanations can be derived from the same model that originated the design, explanations vary according to the perspective and goals of the person requesting it. In other words, it should be possible to get different views of the same information (relevance).

This hypothesis was not broadly tested by the studies. Only two subjects were considered -- an owner's representative and a mechanical engineer. Under the time pressure of finishing a doctoral thesis, it was not possible to do formal studies for a building inspector or other document users. However, in addition to the formal studies, many live (but unrecorded) system tests were performed with various visitors to our research facility over a period of weeks. These visitors ranged from people in the building trades, professors in civil and mechanical engineering, and contractors. In general, the assessment of the approach was very positive.

- "Explanation Kernel" Hypothesis: A small set of parameterized document types suffices to provide form and context for most document queries.

The set of reports generated by ADD corresponded to the kinds of reports used in a standard fashion in the industry. Thus, although we did not test this hypothesis over diverse populations of users, we did model the system reports after current documentation practice.

5. Final Discussion

Many researchers in the area of design rationale have studied ways to support design documentation over the last few years. Three major approaches to design rationale have been proposed: to record the sequence of actions (history-based rationale), see (Lakin, Wambaugh, Leifer, Cannon and Sivard, 1989; Karinthi; 1992), to record the arguments and issues raised during design (argumentation-based rationale) see (Conklin and Begeman, 1988; Fischer, Lemke, McCall, and Morch, 1991; McCall, 1986; MacLean, Young, Bellotti and Moran, 1991; Lee, 1990), and to record the final product model (device model-based rationale) see (Gruber and Russell, 1992; Baudin, Sivard and Zweben, 1990). However, these approaches have not addressed the requirements that we observed for design documentation, namely, low documentation overhead, documentation completeness, documentation consistency and easy access to relevant information.

This paper has presented a broad sweep of our field studies both prior to and after the creation of the ADD system. Although the work involves analysis of many hours of video data, the goals of this work have been more akin to an in-depth "proof of concept" rather than an investigation into methodology itself. In this regard there are many ways that the studies themselves could be strengthened. Certainly, one area of improvement would be in simply increasing the number of subjects. For example, a reasonable question would be whether the people testing the final version of ADD were likely to be biased in its favor since they had no real investment in it, were exposed to it for a short period, and did not need to invest in its use in their work on a regular basis. An aside of some interest on this point is that both of the main subjects and many of the people who used it in an informal way became very excited about its potential, and wanted to obtain commercial versions for their organizations.

This excitement about ADD reflects our own overall assessment that the methodology of ADD has the potential to solve an important problem. Preliminary design is a crucial stage in design both for designers and owners. It is a stage in which the major costs of building are often determined. It is a stage in which many different design options and criterial are communicated and negotiated. It is also a stage in which there are substantial incentives for the designers to not spend too much time on documentation. An approach like ADD radically changes the cost structure of creating documentation, and more generally, of providing access to good information about design choices to the many documentation creators and users.

References

Baudin, C., Sivard, C. and Zweben, M. (1990). Using Device Models and Design Goals for Design Rational Capture. *Techinical Report, MS244-17*, NASA Ames Research Center.

Conklin, J. and Begeman, M. L. (1988). gIBIS: A hypertext tool for exploratory policy discussion. *Proceedings of the 1988 Conference on Computer Supported Cooperative Work (CSCW-88)*, Portland, Oregon.

Coyne, R. D., Rosenman, M. A., Radford, A. D., Balachandran, M. and Gero, J. S. (1990). *Knowledge-Based Design Systems.* Reading: Addison Wesley.

Ericsson, K.A. & Simon, H.A. (1984). *Protocol Analysis. Verbal reports as data.* The MIT Press: Cambridge, Massachusetts.

Fischer, G., Lemke,, A. C., McCall, R. and Morch, A. I. (1991). Making Argumentation Serve Design. *Human-Computer Interaction* 6(3/4)

Garcia, A. C. B., Howard, H. C. and Stefik, M. J. (1993). Active Design Documents: A New Approach for Supporting Documentation in Preliminary Routine Design. *Technical Report #48*, Center for Integrated Facility Engineering, Stanford University, CA.

Gruber, T. R. and Russell, D. M. (1992). Generative Design Rationale: Beyond the record and replay paradigm. *Design Rationale*. Hillsdale: Lawrence Erlbaum Associates.

Guindon, R. and Curtis, B. (1988). Control of Cognitive Processed During Software Design: What Tools are Needed? *CHI'88 Human Factors in Computing Systems,* Washington D.C.

Karinthi, R. (1992). Capturing Design Rationales for Use in a Concurrent Engineering Environment. *AAAI '92-Design Rationale Workshop*, San Jose, CA.

Lakin, F., Wambaugh, H., Leifer, L., Cannon, D. and Sivard, C. (1989). The electronic design notebook: Performing medium and processing medium. *Visual Computer: International Journal of Computer Graphics* 5(4): 214-226.

Lee, J. (1990). SIBYL: A tool for managing group decision rationale. *Proceedings of the conference on Computer Supported Cooperative Work (CSCW-90),* Los Angelos, CA.

Luth, G., Krawinkler, H., Law, K. H. (1991). Representation and Reasoning for Integrated Structural Design. *CIFE Technical Report #055*, Center for Integrated Facility Engineering, Stanford University, CA.

MacLean, A., Young, R., Bellotti, V. and Moran, T. (1991). Questions, Options, and Criteria: Elements of A Design Rationale for User Interfaces. *Human Computer Interaction* 6(3/4)

McCall, R. (1986). Issue-serve systems: A descriptive theory for design. *Design Methods and Theories* 20(8): 443-458.

Newell, A. and Simon, H. A. (1972). *Human Problem Solving*. Englewood Cliffs: Prentice-Hall, Inc.

Simon, H. A. (1981). *The Sciences of the Artificial*. Cambridge: The MIT Press.

Stefik, M. J.(1995) Introduction to Knowledge Systems. Palo Alto: Morgan-Kaufmann Publishers.