CIFE CENTER FOR INTEGRATED FACILITY ENGNEERING

The Charrette Test Method

By

Mark Clayton, John Kunz, and Martin Fischer

CIFE Technical Report #120
September, 1998

STANFORD UNIVERSITY

Copyright © 1998 by

Center for Integrated Facility Engineering

If you would like to contact the authors, please write to:

c/o CIFE, Civil and Environmental Engineering
Stanford University
Terman Engineering Center
Mail Code: 4020
Stanford, CA 95305-4020

THE CHARRETTE TEST METHOD

Mark J. Clayton, John C. Kunz, Martin A. Fischer

Introduction

This paper discusses validation of an evaluation process in building design. We claim that it both matches designers' cognitive activities and is implementable as software that partially automates the process. The working SME system provides evidence for the latter claim. An example of the Cyclotron Suite shows that the software can automatically create a product model of a building as the result of performing design activities including evaluation. This paper presents methods that we used to provide evidence that the software and the design theories that it embodies are usable by other designers. We developed and used the Charrette Test Method for testing the effectiveness of computing methods. We report on its use to develop evidence for the general suitability of SME and its effectiveness in aiding the design evaluation process. The experiment compares a manual process of solving a design evaluation problem to a partially automated process using the SME prototype.

The problem

The basic issue that we address in the Charrette Test Method is how to determine whether a process performed using one set of tools is superior to a process performed using another set of tools. We will use the term "effectiveness" to refer to the changes in speed and quality of processes that are assisted by software.

A key aspect of testing new software is to realize that the new software requires "re-engineering" of an existing process. The new software (other than games) is intended to accomplish a task using an improved process. The new process must be more effective than other processes, whether they are other computer-aided processes or conventional manual processes. When one of the processes is aided by software, one might refer to this as an issue of "software productivity." However, the term "software productivity" has come to mean the productivity of software development teams.

Judging the merits of a new process, especially when the process is only implemented to the point of a research prototype, is a difficult task. The charrette method used as a comparative empirical experiment is unusual and possibly unique in the fields of design methodology and product modeling in the AEC industry. We used the Charrette Test Method to test the SME software. It provides empirical evidence confirming some theoretical benefits of the Virtual Product Model and integrated computer-based product models in general. At a more abstract level, investigators can use the method widely to gather empirical evidence for the effectiveness of a formalization of the design process. The results of the process that are achieved using the new software can be compared with results using an alternative process, such as a conventional process.

The word "Charrette"

The word "charrette" is widely used in the architectural community to signify a short but intensive design exercise in which a practical problem is solved under time pressure. Charrette is actually the French word for "cart." Its architectural meaning originated in the traditions of the Ecole des Beaux Arts. To allow a student to work on a project until the last possible moment, the student's friends would load the

student and his drawing board into a cart and deliver him and his project to the jury. The student continued to work "en charrette" to apply the finishing touches to the project. The Charrette Test Method employs a short but intensive design problem and compares the performance of several designers in undertaking the problem using various carefully defined design processes.

Outline of the paper

Tests of Software to Support Architectural Design

Although there are reams of research reports and articles that describe experimental design software, concrete evidence for the expected benefits of the software is conspicuous by its absence in the literature. Perhaps due to the inherent confidence and faith of innovators, improvements to the design process that would result from the software are assumed rather than demonstrated. In the field of intelligent CAD research for the AEC industry, research is often published before it is tested empirically. Typical evidence that is presented consists of reference to other published research, an informal argument, an expression of opinion that is based upon professional or academic experience, or construction of a software prototype. We will refer to these research methods as "theoretical evidence." For example, a product model based upon a unifying concept of spaces and space boundaries has been presented with purely a theoretical argument (Bjork 1992).

Since theoretical evidence is ubiquitous in research projects addressing computing methods for design, we will characterize the non-theoretical methods as "test" methods that attempt to provide corroboration for a theory. We will also refer to the new process or model suggested by a research project as the "innovative process" in contrast to a "conventional process."

The methods that are most commonly used may be categorized as:

- a worked example, either at the level of "toy" problem or a more realistic problem,
- a demonstration of the software to a representative audience, and
- a trial.

What is a good test method?

From the point of view of a potential adopter of an innovative process, the paramount question in research into computing methods for design is whether the proposed process is more effective than existing processes. A good test method must address effectiveness of the new process. Effectiveness has several dimensions. First, an effective process must be general in several ways. It must be usable by not only the researchers, but also by a typical designer or software user. An effective process must also be usable for not only a test case, but for a relatively large class of problems. Beyond generality, the effective process should be relatively fast in comparison to other processes. It should also be relatively accurate or productive of higher quality solutions.

Any experimental test may be challenged on the basis of reliability and validity. Reliability is the question of whether the experiment would produce the same results if it were repeated under different circumstances, most importantly with different users and by different researchers. Reliability is partly a question of to what extent can the results be generalized. It is also a question of whether the results are objective. Validity is the question of whether the results actually reflect the dimension that the evaluator intended to measure. A good test method must be both reliable and valid.

A good test method must also be practical. It should be cost-effective and it must be within the skills of the evaluator. It must also be generally applicable to a wide variety of research projects.

Worked example

In design methods research and computing research, a worked example is very often used to demonstrate that the ideas being promulgated are applicable. A worked example describes a plausible scenario using the innovative process. In some cases, the worked example may be a description of a hypothetical session of a designer, or it may be a description of a populated data structure, or it may be a report of an actual session with a software prototype. We use the term "worked example" to refer to hypothetical sessions imagined by the researchers or actual sessions undertaken by the researchers.

Worked examples can vary in degree of complexity and closeness to real-world situations. At the simpler and less convincing end of the scale, a worked example may employ a trivial case to help illustrate the concepts in the research. A worked example may also be based upon a careful transcription of a real-world situation. In that case, the worked example provides stronger evidence for the validity of the research than one that relies upon a "toy" problem.

Due to its simplicity and small demands upon the researchers' resources, the worked example has often been used in arguments for comprehensive architectural design methods. For example, one often-cited investigation suggests that a designer can map out the constraints upon the design problem and then apply set theory to derive a solution (Alexander 1964). In addition to the theoretical argument, the innovative process was shown to be usable by inclusion of a worked example of the design of a small agricultural village. A very extensive set of worked examples has been presented in support of an innovative way of studying the integration of systems in buildings (Rush 1986). Nineteen buildings are diagrammed and discussed using the method for studying integration described in the book.

The conceptual model of integrated CAD as a general constraint management system has been suggested by El-Bibany (1992). This research produced a general architecture for CAD software systems, called CONCOORD. A worked example shows how a CAD system based on the CONCOORD architecture could be used to model a single room and to reason about the architecture, the structural system and the construction systems. The worked example was at least partially implemented as software.

Another research project suggested that an object-oriented paradigm would be an appropriate means for programming integrated CAD software (Froese 1992). It demonstrated how such a system could be implemented as two software products, GenCOM and OPIS. OPIS is a working prototype that runs on NeXT computers. Its domain is limited to the construction planning and cost estimation of concrete frame structures. The research was tested with several sample projects of small to moderate complexity. These tests are essentially worked examples.

The Building Product Model (BPM) is another research software system that addresses integration of design and construction processes through software integration (Luiten 1994). It was programmed using the Eiffel language and incorporates many concepts from the STEP initiative. The BPM was demonstrated to permit the modeling of a commercial precast concrete structural system. The worked example is of unusually high quality as it is derived from a real-world construction system.

The CAADIE project also used a worked example (Chinowsky 1991). CAADIE provides space layout generation for architectural designs, integrating the consideration of several factors. The worked example is a small building intended as a university research facility. It consists of fifteen rooms, such as faculty offices, seminar rooms, and research labs. The problem was invented for the research project.

The Engineering Data Model research has been described in several publications with worked examples. A composite structural wall is worked out in detail in one paper to demonstrate the sufficiency of EDM to provide for extensibility and modularity (Eastman 1992b). The example does not appear to be derived from a particular real-world project.

The Primitive-Composite approach is a method for constructing a database to support multiple domains of reasoning about a design (Phan and Howard 1993). The primary test of the research was a worked example of an integrated database for electric power tower design. A real-world problem was used in developing the worked example.

A variation on the worked example was used in research to develop software to assist in structural engineering of buildings (Fuyama 1992). The worked example was taken from a problem in an engineering textbook. As part of the development of a new textbook, a practicing engineer wrote the problem and provided a solution. The software produced a structural design that was very similar to that described in the textbook problem.

Researchers have depended upon the worked example as the primary supporting evidence for theoretical arguments. The worked example has advantages and faults. To its credit, it is relatively easy to create and provides a large degree of confidence compared to a purely theoretical argument. It is not surprising that worked examples are so prevalent in the literature. To its detriment, the worked example lacks objectivity since the evidence is obtained only from the researchers who are the proponents of the innovative process. Most often, only one example is worked, and thus there remain doubts of the generality of the process. An open question may also be whether other researchers or designers can use the proposed design process. Thus the worked example lacks strength in terms of reliability and validity. More rigorous test methods are certainly desirable.

Demonstration

Occasionally in a research project that addresses computing methods for design, software has been implemented to the extent of a working prototype. When this has been achieved, the researchers have the opportunity to present the software to outside experts in a plausible design scenario and elicit feedback and comments. While research software is often demonstrated informally, the formal collection of feedback is rare. One project in which feedback was collected formally employed a multimedia mock-up of software for architectural design (Vasquez and Mendívil 1996). The software mock-up was demonstrated to over 50 professional designers.

SME was demonstrated numerous times to representatives from industry and academia. However, formal surveys of audience response were not collected.

The demonstration is stronger evidence than the worked example. The use of outside experts provides greater confidence in the generality of the innovative process. A large sample of experts provides increased reliability. If carefully designed, a survey of the experts may provide some evidence for validity.

Trial

In a trial, individuals who were not involved in the research perform an assigned design task using the innovative process. An example research project that used a trial is the project on Active Design Documents (Garcia, Howard and Stefik 1993). A "field test" was conducted in which two professional HVAC designers used the research software to prepare a preliminary mechanical system design for a multi-story office building. Data was collected on how they used the software including which modules they used and which commands they invoked. An informal interview elicited complimentary responses from the participants. In addition, theoretical arguments for the worth of the techniques were presented.

Some of the most extraordinary evidence for a design theory was offered for the pattern language theory of architectural design (Alexander et al. 1975). The evidence is a description of the master planning of the University of Oregon campus in which an innovative user-centered design process was used. Although this evidence demonstrates clearly that Alexander's process can be used to create a satisfying design, it does not meet high standards for an experiment. There are no controls for reliability; having used the pattern language once, there is no indication that it can be used on another project or by other practitioners. There are also no controls for validity. Was the satisfaction of the project due to use of the pattern language or to some other factor? The Oregon Experiment is essentially a very large and intensive trial that lacks controls for reliability and validity.

SME has been tested in numerous trials. Over 40 representatives from the AEC industry have used SME during the CIFE summer workshops in 1994 and 1995. These participants were not surveyed

for their responses. However, their successful use of the software in small design evaluation problems demonstrates that SME and the concepts embodied in it are widely understandable and usable.

As a test method, the trial provides strong evidence that the innovative technique is usable by people other than the researchers and thus is an improved test method in terms of generality. However, a single trial or small number of trials still falls short in demonstrating reliability. A trial also does not provide evidence for validity.

Other research methods

Other research methods for design processes and computing methods for design are notable. Protocol studies have been used to support design theories in which researchers have observed designers at work. Protocols of design educators interacting with students have been used to support the reflection-in-action theory of professional expertise (Schön 1983). Another research effort set up a synthetic experiment of a design problem and then collected information about the processes used by the participants to test hypotheses regarding "fixation" in design problem solving (Purcell et al. 1994). Other protocol studies have analyzed observations of designers to produce a coding of the design activities in terms of consideration of structure, function and behavior (Gero 1995). Rigorous application of formal logic has been used to support a theory of communication and negotiation among designers (Khedro, Teicholz and Genesereth 1993).

Motivation for new methods

All of these research test methods fall short of providing definitive evidence that the innovative process or technique will be effective. As generally practiced, the worked example is produced by the primary researcher and thus lacks objectivity, reliability and validity. The worked example does not include specific measures by which success may be judged. The demonstration method, while introducing an outside audience to provide some objectivity, does not show that the innovative technique is usable by people other than the researchers. It may also suffer from a bias that is introduced into the survey results as the respondents attempt to please the surveyor. Thus, although it may be reliable, it is often not strongly valid. The trial method achieves a much higher degree of objectivity and demonstrates generality across users. However, lacking a recognized metric for defining the effectiveness of the software, the trial does not show that claimed benefits will ensue from adoption of the software and thus is also weak in terms of validity. Without testing the software on several design problems, these methods do not demonstrate generality of the innovative process across a range of design problems.

From this survey of research in computing methods for design, it is clear that the usual methods of testing design research are certainly weak and arguably inadequate. Improved test methods could provide:

- increased reliability through use of multiple examples or trials;
- increased reliability due to repetition of the test by other researchers and testers;
- greater validity due to reduced bias and increased reliance upon objective measurements; and
- greater validity for arguments of effectiveness due to clear comparisons to current practice.

The Charrette Test Method can address the limitations of other methods and can offer these improvements. The Cyclotron Suite Charrette that was conducted as part of this research demonstrates that the Charrette Test Method is practical for research projects in computing methods for design. This new test method is grounded in techniques developed for software usability testing.

Tests of Software

The field of computer science (especially its sub-discipline of human-computer interaction research) has produced several ideas that are relevant to testing computer-aided design processes. The productivity of software developers has been studied extensively to attempt to identify advances in

productivity due to new programming paradigms and techniques. Less frequently, software effectiveness in comparison to conventional, manual methods has been examined. Finally, software usability is a strong computer science specialty that has a collection of accepted practices and rigorous methods. The Charrette Test Method draws heavily upon software usability methods.

Software development productivity testing

As mentioned earlier, software productivity is generally accepted to mean the productivity of the programmers and software engineering team in producing the software. Rarely do published studies of software productivity address the potential increases in productivity that can be gained by users of the software. Nevertheless, the literature in software productivity is relevant.

The study of software development productivity is occasionally a study of the process by which the software is produced with an intention of choosing an effective process. It has been pointed out that a new process for use by a development team can be seen as a product that is in itself designed (Gelman 1994). A new process may be produced by following a fairly typical iterative design approach including stages of Define, Deliver, Apply, Assess, and Refine. However, as admitted by Gelman, the "Assess" stage in which the new process is evaluated typically has inadequate metrics. In the research described by Gelman, assessment is entirely by submittal of change requests by users of the process. The process designers rank the requests by importance.

Numerous measures of software development productivity have been suggested and used to assist in the management of software projects (Grady 1994). Some of these are: counting lines of code, counting "found-and-fixed" defects, and various measures of program complexity. These measures are then typically compared to time expended in development to arrive at an efficiency or effectiveness rating for the development method that was used. Although Grady argues that the metrics have rarely if ever been validated empirically, they are a precedent that suggests metrics for comparing architectural design processes. Similarly, building design processes may be evaluated by isolating quantifiable measurements and then relating them to the time expended.

Software effectiveness testing

Effectiveness testing of software must somehow compare a process using one software product to either a recognized baseline or to a process using another set of tools. As baselines are generally non-existent, effectiveness testing should compare at least two processes. In most cases of software effectiveness testing, both sets of tools have been software packages.

One common way to compare software packages is by inspection by an expert. Informal reviews that use this method are commonplace in the popular press (Sullivan 1997, Peppers 1994). These reviews are often structured as feature comparisons or comparative trials by an expert evaluator. They may also involve measurements of times necessary to complete a benchmark task. Such reviews are questionable as to either reliability (they may be mostly unsupported opinion) and validity (the reviewer may not be comparable to a prospective user).

More formal comparisons are occasionally conducted. Some reviews compare various software packages in speed of performance of some set task. For example, several CAD software systems were compared in terms of time expended in performance of some common tasks (Olson 1993).

A test conducted by Autodesk to compare AutoCAD R12 to AutoCAD R13 is notable in its rigor and as a model for design process research (Autodesk 1995). Autodesk argues that the benchmarks that have been used to measure CAD performance are inadequate for obtaining an understanding of gains in productivity. The testers observe that "Speed or performance is used to evaluate the absolute time the software takes to perform an operation, whereas productivity refers to the amount of work that can be accomplished in a given timeframe." The specific objective of this test was to compare the difference in overall user productivity between the two versions. The test was not intended to compare the speed or performance of the software in any particular task or operation.

In the study, the test conductors attempted to factor into the experiment the typical work environment, human work habits and project requirements. The format of the experiment required the test users to modify a set of drawings to conform to a set of marked up drawings. Much effort was undertaken to insure that the test problem was realistic in scope and complexity. The test users were expert operators hired from a CAD service provider. The test included studying the given drawings, developing a plan of attack, organizing the information, drawing new elements and correcting old elements, and finally scaling and plotting. The testers repeated the problem several times to overcome learning curves and tester variability. The final conclusions were drawn from the best times recorded on each of the platforms and versions.

Although the study cannot be considered definitive, it nevertheless provides convincing empirical evidence regarding relative productivity of AutoCAD versions. The reliability of the experiment is low due to a small sample size and has not yet been confirmed by additional tests. The experiment also involves a particular kind of problem (drafting) in a particular domain (architecture) and thus does not establish the effectiveness of AutoCAD more broadly. Additionally, the range of problem solving processes is small, involving only versions of AutoCAD. No attempt was made to say that AutoCAD was better than other CAD systems or manual drafting processes.

Another researcher has examined whether the use of 3D solid modeling is a more productive way of teaching spatial visualization than manual methods and wireframe modeling (Devon et al. 1994). The experiment employed test users who were first year students in an engineering program. Some sections of the class were taught using solid modeling software while other students used wireframe modeling software and manual drawing methods. Spatial visualization was measured at the start of the course and at the end of the course using a standard and accepted test of spatial abilities. The researchers analyzed the scores of the students using statistical methods. They concluded that the data supported the statement that the use of solid modeling software enhanced the students' ability to obtain spatial visualization skills. The researchers point out some inadequacies of the experiment. The methods were somewhat unstructured, and the standard test of spatial abilities was arguably too easy for the engineering students. Nevertheless, the study is very interesting in that it demonstrates that the effectiveness of a software-assisted method of accomplishing a task can be measured against conventional methods.

One comparison of software-assisted innovative processes to their manual predecessors focuses upon the category of expert systems (Feigenbaum, McCorduck and Nii 1988). The book is unabashedly enthusiastic toward expert systems and focuses upon major demonstrable successes. The bias of the book is indicated by its lack of discussion of failures of expert systems. As the example systems had been in place for many months, it was possible to compare measurements of productivity before the introduction of the computer software to productivity afterward. The efficiencies of the software-assisted processes are due to increased throughput of a process, decreased error rates or increased institutional memory. Of course, the nature of these studies as follow-up studies to the fielding of commercial systems precludes them as models for experimental projects.

These various efforts establish a clear precedent of using comparisons to study software effectiveness. They emphasize metrics that are related to the practical task being accomplished rather than aspects of the software itself.

Empirical artificial intelligence

Empirical methods similar to those used in psychology or biology have been devised for application to artificial intelligence software (Cohen 1995). Experiments that involve repeated trials using software under different inputs and environments are analyzed using statistical methods. Development and adoption of empirical methods can enable artificial intelligence researchers to broaden their kit of research methods and possibly obtain new insights and validation of results obtained with other methods. The focus of the empirical methods is upon developing a better understanding of the factors that affect performance of a software agent or artificial intelligence system. Such an understanding can lead to development of more efficient or more accurate software. It may also lead to insights into human cognition as one compares the successes and failures of the software to human performance on similar problems. However, the thrust of

the methods described by Cohen are less toward determining effectiveness and more toward understanding the operation of software under non-deterministic execution.

Software usability testing

The field of software usability engineering has become an established field with a clear set of interests and methods. The field is generally concerned with the comparison, design and selection of alternative user interfaces. Usability has been defined to include issues of:

- learnability,
- efficiency for expert users,
- memorability to allow casual users to return to the software,
- error rates, focusing upon the avoidance of "catastrophic" errors, and
- user satisfaction (Nielsen 1993).

These issues are somewhat different from effectiveness of a software-assisted process. Nevertheless, the methods of usability testing appear to be very relevant.

Nielsen has provided a characterization of the normal methods of the field and various recommended guidelines. Although usability testing frequently includes surveys and expert heuristic evaluation, "User testing with real users is the most fundamental usability method and is in some sense irreplaceable..." (Nielsen 1993, 165). User testing involves having representative test users perform a set of pre-selected tasks. Times are measured and errors are counted to provide measurements of efficiency and error rates. An overall score for each dimension is typically computed as a mean value of the test users' scores, although statistical analysis is common. Subjective satisfaction is generally measured by survey. Rarely used are psychophysiological studies that measure physiological changes in people due to changes in stress, comfort or pleasure. Measurements using such an approach might include pulse rate or presence of moisture on skin surfaces.

In user testing, reliability is assured through use of accepted practices, statistical analysis, comparison to alternative measures of the same dimensions, and repetition of experiments. Expert opinion and theory generally support validity. Since only one version of the software will ultimately be widely fielded, it is difficult or impossible to assure validity by direct evaluation. Pilot tests and follow-up tests can help establish both reliability and validity.

Nielsen discusses some of the techniques for controlling usability tests. A key issue is the variability among users. Care must be taken to arrive at a representative sample of users, yet in small samples one must not include outliers. User variability may be controlled by a "within-subject" test design in which each user tests each of the interfaces under consideration. Within-subject testing introduces another problem, that of learning by users who must repeat the same tasks, albeit on different systems. Learning is reduced by introducing a delay between the trials with each system and by altering which system is tested first. In "between-subject" testing, the users each test only one system, so learning is avoided. However, a much larger sample is required to control the experiment for user variability.

The purpose of "formative testing" is to guide the development of a new software product while "summative testing" is intended to provide statistical assurance of a fundamental user interaction method once a software product has been fielded (Hix and Hartson 1993). To accommodate the rapid cycles of iterative software design, formative testing employs small samples of participants and simplified techniques. Hix points out that the first test often provides the most information and that a sample size of only three to five participants is often the most cost-efficient. In formative testing, elaborate statistical studies are unnecessary. The results may be characterized simply as averages.

An example of a published usability test can further clarify the typical methods. In one study, the subject was the comparative usability of alternative interfaces for manipulating 3D models (Grissom and Perlman 1995). The research also produced a rigorous "standardized evaluation plan (StEP)" for

comparing alternative user interfaces for 3D manipulation. The evaluation was conducted by having several participants use a particular 3D modeling tool to perform manipulations that were described by illustrations. The sessions were videotaped and then times were measured with a stopwatch. To test the reliability of the StEP, several different tests were conducted, including a "between-subjects" design and a "within-subjects" design. The 3D StEP was found to be reliable and valid.

The Design of the Cyclotron Suite Charrette

This section elaborates upon the design of the Cyclotron Suite Charrette.

Definitions

This discussion uses the following terms and definitions:

Process: a set of activities and tools that are used to accomplish the task. The processes are the subjects of the test. In the Charrette Test Method, at least two processes must be studied.

Innovative process: the new, experimental process that will be studied in the test.

Conventional process: a baseline process for performing the task similar to processes used in practice.

Proposition: a statement regarding the effectiveness of the innovative process that can be tested for truth.

Participant: a person whose sessions with the processes are measured to provide the experimental data.

Trial: a session in which one participant performs the operations that provide the test data.

Task: the activity or activities that produce a measurable result. The task conforms to a real-world unit of effort on a project. In the Charrette Test Method, the task should remain constant between the two processes.

Overview

The Charrette was set up to compare the innovative process using SME to a relatively conventional "manual" design evaluation process. The users' task was to evaluate three design alternatives. The participants performed the task with both processes. The comparison of the two processes was based upon measurements of time expended, and accuracy of the evaluation results. To provide some statistical confidence in the measurements, several participants undertook the Charrette and measurements were made for each trial.

Propositions to be tested

The Charrette was intended to test three propositions regarding the design evaluation process modeled in SME:

Usability: Representatives from the AEC industry can use SME, its interpretation functions, and its Virtual Product Model to perform non-trivial evaluations of designs.

Speed: SME and its Virtual Product Model can provide a benefit of increased speed in performing the design evaluation tasks.

Accuracy: SME and its Virtual Product Model can provide a benefit of increased accuracy in evaluating the designs.

In addition, the test of these propositions provides indirect evidence for the underlying model of the design evaluation process. The successful and beneficial use of SME suggests that the underlying

model of interpretation, prediction and assessment is a reasonable way of characterizing the design evaluation process.

Measurements

The propositions require specific measurements to be made of the performance of participants using the two processes.

The first proposition depends upon establishing that the task to be accomplished is non-trivial and that participants can accomplish the task using SME. If the task requires a significant amount of time using manual methods, such as two hours, and is accomplished with a clear amount of variation in results by the participants, it can be concluded that it is a non-trivial task. The problem was calibrated to provide a noticeable degree of challenge to the participants, indicated by the degree of variation in performance by the participants. The number of participants who completed the task with SME can indicate whether the software is usable.

The second proposition can be evaluated by measuring the time taken by participants on each trial. That time can be averaged for all manual trials and averaged for all SME-based trials. The two averages may be compared. The fastest times achieved also suggest which process is potentially faster when used by skilled users.

The third proposition depends upon establishing a measurement of accuracy. One measure of accuracy is to establish what is the correct answer and measure the variation from that correct answer. The variation of all trials can be averaged. However, this method can introduce a poor rating when a small error results in large variation from the correct answer. Another measure of accuracy is the amount of variation displayed by several trials. A process that produces little variation is arguably tunable to produce a correct answer independent of the user or the particular trial. Quantitative accuracy may be measured simply by applying arithmetic calculations. Accuracy for qualitative issues must be measured by finding a way to quantify them. Clear mistakes in a qualitative issue may be identified and counted for each trial. The number of mistakes becomes the error rating for that trial. The precise measurements made for studying accuracy of the two processes are discussed in a section to follow on results.

Limits to scope

Numerous other questions are of interest but were not addressed in the Charrette.

The Charrette only compared two design evaluation processes: a particular manual process and a process using the SME software. There are many other ways to solve the problem, and thus the Charrette results do not provide evidence in support of a best process in an absolute sense.

The Charrette focused upon conceptual building design. It did not provide evidence regarding design evaluation process in other design stages, such as detailed design, or other design domains, such as mechanical engineering.

Furthermore, the Charrette addressed only the evaluation activity in conceptual design. The use of defined and documented design alternatives eliminated the synthesis activity from the processes studied in the Charrette. To eliminate design analysis activities, the criteria for evaluating the alternatives were explicit, documented and complete for the purpose of the Charrette. The participants did not invent new criteria or modify the given criteria. Documentation of the products of design synthesis was also not in the scope. The participants were not required to draw using paper and pencil or the CAD software.

The three design alternatives can not be considered representative of all possible designs in the "design space" of the problem. The Charrette merely measured design activities in evaluating three design alternatives that might arise sequentially in response to the design problem.

In addition, the Charrette has not measured the effectiveness of the processes at multiple project scales. The design evaluation problem solved in the Charrette is simple, although it is not trivial. The Charrette has not demonstrated that either method is usable or superior for more complex design problems.

The sophistication of the evaluation used in the Charrette is also very low compared to a real-world design problem. Incorporation of more complex and sophisticated evaluations, such as a more complete cost estimate, a construction scheduling issue, a lighting analysis or a more complete building code critique, could produce dramatically different results. A computationally intensive evaluation could favor a software-based process while an evaluation that is complex, subtle and intuitive may favor a manual process.

Finally, the Charrette did not attempt to demonstrate generality of the processes across many design evaluation problems. All participants solved the same design problem.

The participants' task

The design evaluation task of the Charrette was the evaluation of three alternative designs for a cyclotron suite addition to a hospital in Palo Alto. The three alternative designs are the same as those portrayed in the scenario. Each design solution satisfies some of the programmatic requirements in the evaluation but fails in others. The evaluations address the four issues described in the scenario: energy consumption, construction cost, spatial layout, and egress provision. The complexity and accuracy of the evaluation was intended to conform roughly to evaluation conducted during the conceptual stage of a project to assist in choosing among conceptual alternatives.

Participants were given a two hour period to respond to this problem. The expectation was that no participant would evaluate all three alternatives in the given time. The two-hour limit imposed a time pressure upon the participants. They were told to work quickly but accurately.

The spatial requirements for the project are based upon specifications for a suite of rooms to accommodate a cyclotron. The requirements were distilled and simplified from a specification document obtained from Dillingham Construction Company. The spatial program consists of requirements for a cyclotron room and additional support rooms, such as a control room, a water recirculation system room, a radiology laboratory, and a storeroom. Each room has rough or specific size requirements, such as minimum length and width, or floor area. Other spatial requirements are included, such as adjacencies between spaces and connectivity of spaces through doors. In addition, the spatial program places requirements upon the size of doors and hallways to support the delivery and maintenance of equipment.

The design evaluation problem includes a limited consideration of construction cost. The costs are to be estimated at the assembly level of detail, using data from the Means Assembly Costs reference (Means 1995). SME includes work packages for the construction cost of the building envelope, foundation, structural system and interior. Electrical, mechanical and furnishing costs are not part of the problem. Management costs and financing costs are also excluded from consideration.

A small excerpt from the Uniform Building Code specifies requirements regarding egress (ICBO 1985). This excerpt requires participants to determine the number of exits for spaces based upon occupancy loads of rooms. It also places restrictions upon the width and height of doors.

The operations cost due to energy consumption is the fourth issue in the design evaluation problem. The problem requires an estimate of heating and cooling loads. Consideration of energy is different in the manual trials from the consideration of energy in the computer-based trials. In the manual trials, a very simplified calculation of energy flows through the walls, doors, windows and roof has been specified. Although simple, the calculations are based upon those specified in the Prescriptive Approach, using the Overall Envelope option of the California Energy Efficiency Standards (CEC 1992). For the computer-based trials, the DOE-2 energy simulation software was used to provide a detailed and extensive computation of energy usage.

The use of two different calculation methods for energy is an asymmetry in the Charrette. The DOE-2 software is widely recognized as a powerful and effective technique for simulating energy use. A comparable manual method of simulating energy usage would be so difficult as to be impractical. The use of DOE-2 in the computer-based trials is thus a practical advantage of the computer-based method over the manual method.

Preliminary trials with the problem suggest that a trial requires between two and three hours to complete using either the manual methods or the computer-based methods. Although it is not a particularly difficult problem, it is certainly not a trivial problem.

The alternative processes

The process used by the participants was the independent variable of the experiment. The process could either be a "manual" process or a computer-aided process.

The manual process employed tools of paper, pencil, and pocket calculator. It was nevertheless a relatively standardized process to assure that measurements of the design evaluation activities could be recorded. The participants addressed each critique issue sequentially although they chose their own sequence for the issues. The participants used a set of forms and carefully designed procedures for each issue. The forms are included in Appendix 1.

The forms structure the process as follows:

1) On each form, the first step is to record the starting time.
2) The next step is to fill in the left side of the form, listing all of the items to be checked or included in the calculations. At completion of this task, the time must again be recorded.
3) The third step is to fill in the right side of the forms, which are the calculations and comparisons. When the calculations are complete, the time is again recorded.
4) When finished with all of the forms for one alternative, proceed to the next alternative.

Step 2 corresponds roughly to the interpret step in design evaluation. Step 3 corresponds roughly to prediction and assessment.

Although the forms guide the participant through the process of collecting and formatting data for use in analyzing the results, they are not exhaustive checklists. Some vagueness and necessity for judgment was incorporated into the standard procedures. An over-formalization of the process would not be representative of contemporary practice.

The computer-aided process employed the SME prototype. It clearly incorporates the three steps of interpret, predict and assess. The software was run on Sun SPARCstation computers.

Measurements

The forms described above are the measurement instruments for the manual trials. They record not only the evaluation results and the process, but they also collect time-stamps to allow reconstruction of the times spent on various activities. The forms provide times for the interpret activity but do not differentiate between the predict and assess activities. Consequently, the time measurements lump the predict and assess activities together as a critique activity.

The SME software collects the measurements for the computer-based trials. A log capability was implemented in SME to record the major operations performed by the participants and how long the operations required. The time spent in fixing technical computer problems has been eliminated from the comparisons. The software also saves the product model produced by the user by the interpreting, predicting and assessing activities. This product model can be reloaded into the Kappa environment and inspected to determine the accuracy of the participant's work.

Participants

Five individuals participated in the Charrette. Four participants were students in the Civil Engineering Department at Stanford University. They all have had some involvement in the AEC industry as architects, structural engineers, contractors or construction managers. The fifth participant was a student in Computer Science who has contributed to research in civil engineering at the Center for Integrated Facility Engineering. All participants had some degree of proficiency with computers, although not necessarily with CAD systems or AutoCAD. Some participants had used other versions of SME for coursework or research.

To provide a control for the variety of knowledge and skill level among the participants, the Charrette was designed as a "within-subjects" experiment. The participants performed the Charrette once with the manual process and once with the computer-aided process. Two participants used the manual process first, and two participants used the computer-aided process first. An interval of about one week between trials by the same person was intended to reduce the effects of learning. One participant performed the Charrette only once, with the computer-aided process.

The participants were trained in both processes. The training consisted of the use of the process to evaluate a simple example building. During training, the participants received personal instruction and were allowed as much time as they desired. Training consisted of between 45 minutes and 1-1/2 hours for each method. Participants should not be considered experts at either method.

Results of the Charrette

The Charrette provided clear comparisons of the time required for each process and the accuracy and reliability of the results. Reduced time required for evaluation is clearly valuable in reducing the cost of design. Accuracy is also of obvious value. The Charrette results suggest that the innovative process is in some ways and for some tasks more effective than the manual process. However, the manual process may also have some advantages.

Time measurements

Table 1 shows the time measurement data. The chart summarizes the times expended by participants in performing the tasks and sub-tasks. No participant successfully evaluated all three alternatives in the allocated two-hour period. Although the objective was that participants should halt after two hours, several participants chose to work longer. The results include their results as well. Each trial shows an incomplete sub-task as the last entry in the chart. All participants completed at least an evaluation of the first alternative and some participants completed an evaluation of the second alternative.

Interpreting the time measurements

Figure 1 shows the times for the various trials as a bar chart.. The average elapsed time for the manual trials at each step is less than the average time for the computer-based trials, suggesting that the manual process is faster. The average elapsed time at the completion of the evaluation of the second alternative was 106 minutes using the conventional process and 129 minutes using the innovative process. Furthermore, interpreting manually clearly took less time than interpreting with the computer. However, prediction and assessment took less time with the computer than with the conventional process. As prediction and assessment are fully automated in SME, the user interaction with the software during interpretation appears to be a bottleneck in the process.

Error! Not a valid link.

Figure 1, Comparison of times between conventional process and innovative process. The chart shows the two measurements for the first alternative in black. The two measurements for the second alternative are shown in white. One trial includes a measurement for the interpret activity for the third alternative, shown in gray. On average, the conventional process was faster.

		First alternative		Second alternative		Third alternative	
	Trial ID	Finish interpretation	Finish critique	Finish interpretation	Finish critique	Finish interpretation	Finish critique
Conventional	C-A	*Not performed*					
Process	C-B	0:40	1:20	*Incomplete*			
	C-C	0:42	1:05	1:27	1:50	*Incomplete*	
	C-D	0:12	0:41	1:05	1:27	1:41	*Incomplete*
	C-E	0:24	1:02	1:15	2:00	*Incomplete*	
	Average	**0:29**	**1:02**	**1:15**	**1:45**		
Innovative	I-A	0:18	0:26	1:23	1:30	*Incomplete*	
Process	I-B	1:31	1:42	*Incomplete*			
	I-C	1:01	1:14	2:09	*Incomplete*		
	I-D	1:20	1:40	2:39	2:51		
	I-E	1:01	1:09	2:00	2:07	*Incomplete*	
	Average	**1:02**	**1:14**	**2:02**	**2:09**		

Table 1: Time measurements. The measurements are provided as elapsed time in hours and minutes. The completion times for the nine trials show that the conventional method was faster than the innovative method.

Several explanations for the faster performance with the conventional process can be put forward. The most obvious conclusion is that the conventional process is simply faster than the innovative process. However, other factors may also have affected the results. The conventional process may have achieved greater speed because of its greater familiarity, or at a sacrifice in accuracy and completeness, or purely due to inadequacies of the user interface of the research software. There is also a problem with the implementation of SME using AutoCAD Release 12 that reduced the reuse of information from one alternative to another. AutoCAD Release 12 does not retain identifier numbers for all entities during operation. Thus, many entities had to be manually re-interpreted when conceptually the interpretation should be reused from one alternative to the next. Reprogramming SME using AutoCAD Release 13 or Release 14 could lead to greater reuse of interpretations from one alternative to another and thus decreased time spent interpreting.

The trials provide some evidence that the innovative process can be faster. The fastest trial for critiquing the first alternative was the Innovative I-A trial. This participant had extensive experience with the software and thus the Innovative I-A trial may reflect the performance of an expert user.

Figure 2 shows expert performance as a basis for comparison. The innovative process was considerably faster, both in interpreting the alternatives and in critiquing them. Of course, these results may reflect a bias.

Accuracy in quantitative evaluation issues

In the two design evaluations that produce quantitative results, energy and cost, the measurements of the trials are very simple and are shown in Table 2. Energy A is the heat gain factor that was calculated using the manual method, Energy B is a total heat loss factor calculated using the manual method, and Energy C is the total cooling load taken from the DOE-2 output produced by SME. Only one measurement is shown from DOE-2; if this measurement is similar between two runs then it is likely that all values are similar.

The numeric results, shown in Table 2, suggest that the innovative process using SME is more accurate than the conventional process. The SME results were relatively consistent among all trials. For example, among the trials with SME the cost of the first alternative ranged between $63,940 and $77,005, or a range of $13,065. Among the trials using the conventional process, the cost of the first alternative ranged between $55,725 and $85,024, or a range of $29,299. The energy measurements are also more consistent with the innovative process than with the conventional process. Clearly, the innovative process was more accurate than the conventional process for these evaluation issues.

		First alternative			Second alternative		
	Trial	Cost, US $	Energy A, unitless	Energy B, unitless	Cost, US $	Energy A, unitless	Energy B, unitless
Conventional	C-A	*Not performed*					
Process	C-B	$77,287.05	364.86	113.02	$94,706.05	*Incomplete*	*Incomplete*
	C-C	$85,024.00	347.38	119.46	$107,830.00	408.99	119.46
	C-D	$84,016.36	386.70	35.36	$122,271.64	396.89	35.36
	C-E	$55,725.03	291.29	80.20	$105,615.00	366.26	80.20
			Energy C, mbtu/year			Energy C, mbtu/year	
Innovative	I-A	$77,004.60	*Incomplete*		*Incomplete*	*Incomplete*	
Process	I-B	$74,487.70	106.75		*Incomplete*	*Incomplete*	
	I-C	$74,487.70	105.96		$97,066.20	142.01	
	I-D	$63,940.00	106.75		$97,066.20	142.01	
	I-E	$68,311.80	106.75		$84,504.30	142.02	

Table 2, Numeric evaluation results. **The numeric results for the evaluations for the nine trials show the range produced for costs and energy measurements.**

Accuracy measurements in qualitative evaluation issues

The results of the spatial and building code critiques are more difficult to measure. For the quantified critiques of cost and energy, the forms used in the conventional process are fairly complete in describing the components of the calculations. However, for the qualitative critiques of spatial layout and egress, the forms do not provide a good record. Very few items were listed, suggesting that the participants either did not check many items in these issues or checked them visually, leaving no record. Thus, the Charrette did not produce adequate measurements of the accuracy of qualitative evaluations in the conventional process trials.

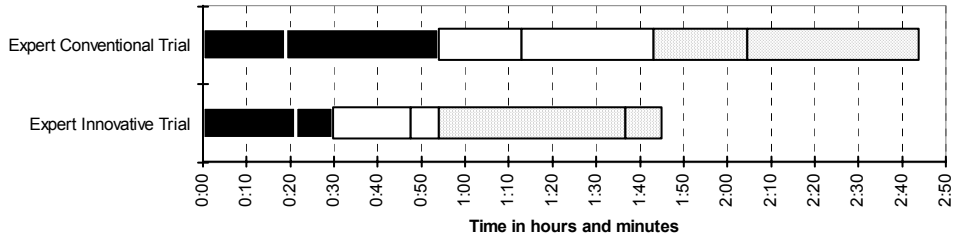

Figure 2: Expert trial times. One of us (MJC) worked the trials several times to achieve times that may be considered expert performance. The chart shows the two measurements for the first alternative in black. The two measurements for the second alternative are shown in white. The measurements for the third alternative are shown in gray. The innovative method was faster.

In contrast, the critique results from an SME trial can be examined as text output and compared line by line to other trials. Records produced with the innovative process enumerate all of the rules that were checked. The results produced by SME were very similar among the trials, exhibiting only one or two items that appeared in one trial that did not appear in another trial.

Although it is not possible to directly compare the two processes, analysis of the innovative process results may give an indication of accuracy of that process. One way to characterize accuracy in the non-numeric evaluations is to compare the number of rules checked in a particular trial to the number checked in a correct reference trial. Table 3 lists the trials and the number of rules checked for the spatial issue and the egress issue. In all cases in which a different number of rules were checked than in the reference trial, there were more rules checked. This may be due to duplicated features or incorrectly interpreted objects. In all cases in which the same number of rules was checked, the trial results were identical to the reference trial results.

The discussion of accuracy has focused upon the results of prediction and assessment. The accuracy of the interpret activity could also be examined. However, the records of the manual trials are insufficient to determine the actual interpretation of the alternative designs. The accuracy of the predict and assess activities in the innovative process implies a similar accuracy in the interpret activity. An incorrect interpretation of the design will probably lead to an incorrect prediction and assessment. A correct interpretation using SME will produce a correct prediction and assessment.

		First alternative		Second alternative	
	Trial	Space	Egress	Space	Egress
Innovative	Ref	23	21	24	26
Process	I-A	23	23	*Incomplete*	*Incomplete*
	I-B	25	21	*Incomplete*	*Incomplete*
	I-C	24	23	24	29
	I-D	23	25	*Incomplete*	*Incomplete*
	I-E	23	29	24	34

Table 3, Number of rules checked in space and egress critiques. **The trials with the innovative process produced fairly accurate reports.**

Reliability

Reliability is important to a process in that the process can be improved and refined only if it is reliable. A process that produces different results depending upon the user or the instance inspires little confidence and provides little opportunity for calibration. A process that produces consistent results that vary only slightly or not at all among trials is arguably a desirable process.

In both cost and energy calculations, some of the trials using SME produced identical results, while none of the manual trials produced identical results. Figure 3 compares the range of results for cost and energy issues between the conventional process and the innovative process. The values have been normalized by expressing them as a percentage of the reference value. For the issue of cost, the range of results among the innovative process trials is obviously smaller than for the conventional process. For the issue of energy, the difference is striking as the innovative process produced identical results in all trials while the conventional process produced results that vary wildly.

The trials also suggest that SME is reliable in its spatial and egress critiques. For Table 1, an error value was calculated by expressing as a percentage the difference between the number of rules checked in a trial and the number checked in the reference trial. The average errors for issue and alternative are also shown. It is apparent that in the spatial critique, there was a high degree of accuracy. The egress critique was in general less accurate, although one trial achieved a perfect score.

One should not conclude too much from the comparisons of accuracy and reliability. A small error in the energy and cost calculations can lead to a large deviation from the correct answer. For example, omission of the roof is conceptually a single error but would drastically alter the estimated cost. These errors in "interpretation" are thus potentially costly. An error in interpretation can be introduced into a manual process or a process with SME. Errors in results with either the conventional process or the innovative process could also reflect insufficient training. Nevertheless, the trials suggest that the process using SME is more effective in achieving accurate results than a conventional process, especially for quantitative evaluations. Furthermore, the process with SME appears to be more reliable than the conventional process.

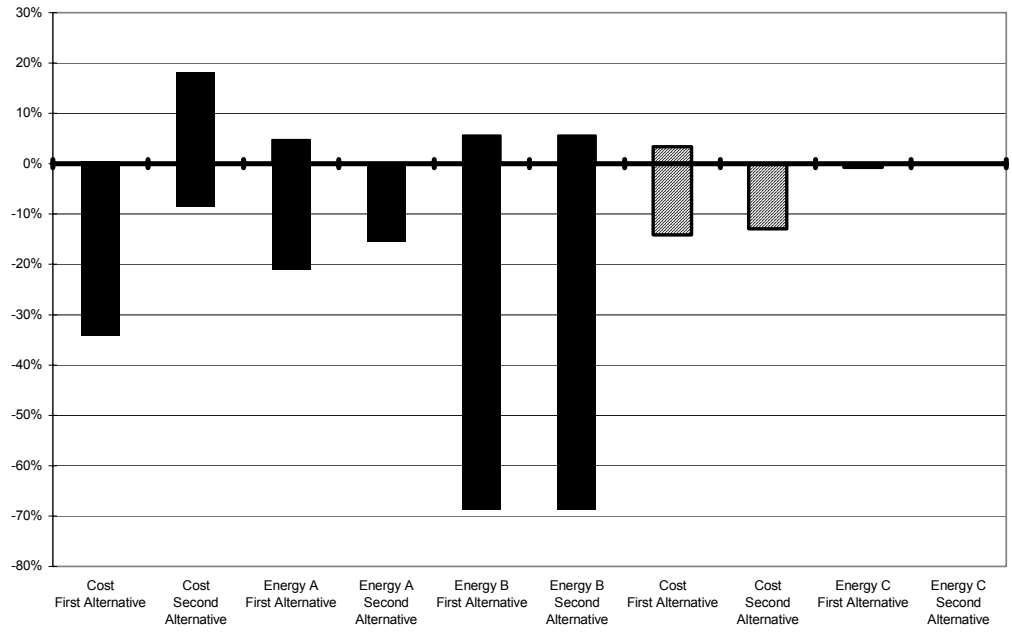

Figure 3, Variation in results in numeric critiques. Expressed as percentages from a correct answer, the black bars show the range of results with conventional methods and the hatched bars show the range of results with the innovative methods. The innovative method produced a much smaller range of results.

Some of the reliability of SME may be due to its provision of commands that help a user to study the state of a partially complete interpretation. For example, a user can hide all of the interpreted entities in the CAD model to reveal which entities have not been interpreted.

Effects of learning

The reason for using a within-subjects test format was to reduce the variability among skill level of participants. However, this test design introduces a danger that a significant level of learning will take place during a first trial that can affect the results in the second trial. If learning takes place, then the second trial should be faster than the first trial regardless of the method used in the second trial. Figure 4 charts time measurements paired by participant with the first trial for each participant shown first. Since all four participants were faster with the conventional method regardless of whether they had already performed the innovative method, the data do not suggest that learning was a significant factor in the results.

		First alternative		Second alternative	
	Trial	Space	Egress	Space	Egress
Innovative	I-A	0.00%	9.52%	*Incomplete*	*Incomplete*
Process	I-B	8.70%	0.00%	*Incomplete*	*Incomplete*
	I-C	4.35%	9.52%	0.00%	11.54%
	I-D	0.00%	19.05%	*Incomplete*	*Incomplete*
	I-E	0.00%	38.10%	0.00%	30.77%
Average error		2.61%	15.24%	0.00%	21.15%

Table 4: Error in non-numeric critiques. The space critique exhibits a very high accuracy in the five trials with the innovative process. The egress critique was less accurate, although one trial achieved a perfect score.

Conclusions Regarding the Effectiveness of SME

Although the Cyclotron Suite Charrette is a small experiment, it has provided empirical evidence regarding the three propositions to be tested. They suggest that software that implements the Interpret-Predict-Assess model of design evaluation is usable and may be effective.

SME is usable

Clearly, SME is usable by designers in performance of non-trivial evaluation tasks. All participants completed an evaluation of at least one design alternative. The interpret, predict and assess capabilities were easy for participants to learn and use. Consequently, they appear to be a reasonable formulation of some of the activities in design.

The conventional process may be faster

The Charrette results suggest that the conventional process is faster than the process with SME. However, SME appears to be competitive with the conventional process in terms of speed, particularly for expert users. While the predict and assess activities are actually performed faster with SME, the bottleneck for SME is the interpret activity.

Further development of SME should consequently focus upon speeding the interpret activity. The fast trials with SME suggest that increased training may overcome the apparent speed advantage of the

conventional process. Improvements to the user interface of SME could also reduce, eliminate or reverse the speed difference between the two processes.

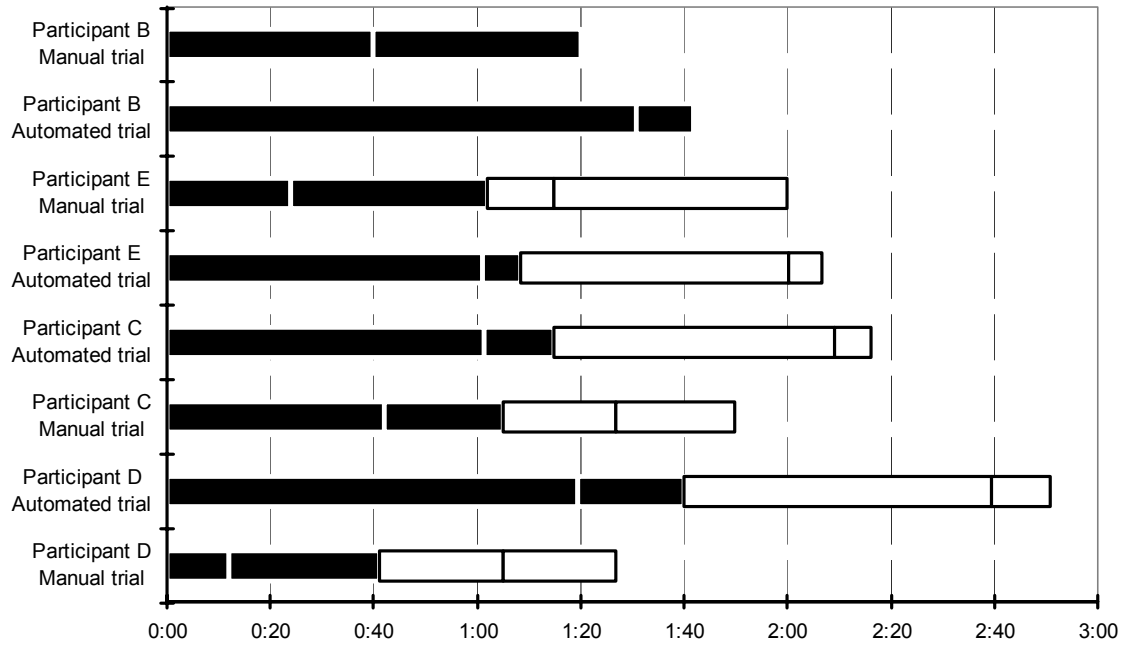

Figure 4: Comparison of first trials to second trials. The chart shows the two measurements for the first alternative in black. The two measurements for the second alternative are shown in white. One trial includes a measurement for the interpret activity for the third alternative, shown in gray. Regardless of whether participants used the conventional method first or the innovative method first, they were faster with the conventional method. This suggests that learning did not introduce a bias between the two trials for each participant.

SME may be more accurate

SME appears to produce evaluation results that are more accurate than the results produced with the conventional process. Even with the small amount of training that participants received, the participants were able to produce evaluations with a high degree of accuracy and reliability.

Other observations and conclusions

Other informal observations of the participants in the Charrette are of interest. Some participants spent extra time "playing" with the software, such as initiating critiques before the interpretations were complete. Some participants expressed a desire to critique a single issue, rather than being constrained to initiate all four critiques. Some participants seemed eager to start the official trials with SME and thus did not complete the training exercise. This probably had a deleterious effect upon their performance with SME as they faced numerous problems with the software operation. Users who had previous experience with SME performed better with it than did other participants.

SME may have produced better documentation than did the conventional process, but the Charrette was not designed to test this hypothesis. The forms that supported the manual process did not record enough information to allow a determination of accuracy and identification of errors, especially for the qualitative critiques. SME, on the other hand, provides complete traces of rules that were used in the critique and user actions. These traces were used in characterizing the accuracy of the trials. The traces serve as an audit trail that could be used in training, determination of responsibility, or process improvement. Improved documentation may be a consequence of a more rigorous interpret activity that enforces thorough documentation. There may be a trade-off between speed and quality of documentation.

The speed achieved with SME in trial I-A and the expert trial suggest that a highly skilled user could have a significant advantage over a person using conventional methods.

The reliability of SME in achieving similar or identical results from many people is also of value. A software-assisted process may be better than a manual process when responsibilities for tasks change frequently. Using a manual process, each person may produce different results and document the work in a different way. With a software-assisted process, the results and documentation from one person to another may always be consistent.

Implications to Further Research with SME

Beyond the evidence for the propositions, the Charrette produced additional observations and feedback that could be used for improving SME.

Correct the tracking of entity identity

Some straightforward refinements to the software could lead to significant improvement in the times required by participants. In the five SME Trials of this Charrette, interpreting the second and third alternatives was a time-consuming activity in spite of the potential for re-use of the interpretations from the first alternative. The software as implemented with AutoCAD Release 12 and AME does not effectively keep track of the identity of AutoCAD graphic entities due to an inconsistency in the internal AutoCAD data structures. This problem has been corrected in Release 13 and Release 14 of AutoCAD. This characteristic of AutoCAD results in the loss of some interpreted values from one alternative to another. A new implementation of SME using the latest version of AutoCAD would allow more reliable tracking of interpreted features. This could lead to much faster times for interpreting the second and third alternatives.

Faster hardware and software

Newer processors, operating systems and application software could also significantly speed the operation of SME. In particular, faster redraw times, such as those provided by AutoCAD Release 14, could also result in faster times for interpreting the alternatives. Faster hardware will also speed the prediction and assessment steps.

Test expert performance

The results produced in trial I-A and the expert trial suggest that, when used by experts, the SME process could be very effective. Additional trials using only expert participants could be conducted to test this hypothesis.

Accelerate the process of interactive interpretation

The bottleneck of the interpret activity could be addressed by automating some of the interpret process. For example, typing in a name for each feature is slow compared to writing the name onto the forms. A more sophisticated capability of the software to generate default names or allow pointing at names with the mouse could overcome this impediment to speed.

Another strategy for speeding the interpret activity is to develop an interpretation module that spans across issues. For example, rather than an interpretation module that addresses construction cost and another module that addresses construction scheduling, a single module could address both. A single interpretation module could address an entire suite of issues so that on routine design problems the single module would be the only one needed. Such an interpretation module would probably be more difficult to use than a simple module, but it would probably result in a noticeable reduction in time in the interpret activity. If such a module were represented as a Virtual Product Model, it would preserve an ability to accommodate new issues and new interpretation modules if the need arose.

Each of these changes to the software could be studied with a charrette to help to determine whether they are actually improvements and to what degree.

Accelerate the process using automated and preset interpretation

Incorporation of some semi-automated interpretation capabilities could also be useful. An example capability would be to find and interpret all AutoCAD blocks with a particular name or select and interpret all red objects on a particular layer as being a particular class of feature. However, with these more sophisticated commands the software not only would be more difficult to program but would be more difficult to operate. More sophisticated and complex software would probably require more training. A charrette could help to establish how much acceleration to the evaluation process could be achieved through automated interpretation or how much training is needed for a more complex application.

A preset interpretation approach would likely improve the speed of a computer-based process by reducing the need to interpret a digital model of the building. Entities in AutoCAD would be interpreted at the time of their placement within the design. It is not yet proven that such an approach will be fast. It might involve time-consuming navigation through very complex class hierarchies for selecting each object. A charrette to test the speed of software using preset interpretation would be a very interesting experiment. A pure preset approach may also reduce the flexibility and adaptability of the software. However, if a Virtual Product Model were used as the underlying data structures for the preset interpretation, the software could also allow a user to over-ride the preset interpretation or add interpretation modules.

A hybrid approach to the interpret activity

A hybrid approach to the interpret activity may be an attractive solution for fast, flexible design-support software. A future software system may allow some elements to be interpreted at the moment of their insertion into the model, some elements to be automatically interpreted once the necessary contextual information is in place, and some elements to be interpreted directly by the user based upon design intents.

The software architecture of SME could accommodate such a hybrid approach. A user interface could be constructed for AutoCAD that presents a catalogue of commonly used building components. When a user selects and inserts a component, the software could transparently interpret the entity as one or more features in interpretation modules. At a later point in the design process, the user could invoke a command to infer the features for an interpretation. For example, the software could search for large steel section components and compute connectivity to automatically produce a structural engineering interpretation. At any time, the user could edit the features that were created in the background or could create features of new interpretation modules. Such a hybrid system could strike an optimum balance between speed and flexibility.

A large number of charrettes could be devised and conducted to explore a hybrid approach. Charrettes could help to determine efficient user interfaces. Other charrettes could examine to what extent participants use the different kinds of interpretation. Tests could begin to provide some understanding of the trade-off between flexibility and efficiency.

Limitations of the Charrette

The experience of having conducted an initial test using the Charrette Test Method indicates that this research method can provide insight into the potential effectiveness of experimental design tools. The Cyclotron Suite Charrette produced intriguing results that are strong evidence in support of a computer-aided design process like that provided by SME. It provides some indication of the real benefits of such a system and the limitations. Nevertheless, the Charrette Test Method by its very nature raises as many questions as it answers. Within the method, there is a potential to answer some of the questions by conducting further trials. This section discusses implications of the Cyclotron Suite Charrette as they pertain to the SME research.

Impact of familiarity and training

The results produced by the Cyclotron Suite Charrette show considerable variability that may be due to differences in ability using either the conventional process or the innovative process. The consistently poor performance using the conventional process on the energy critique and the innovative process on the egress critique may be due to lack of familiarity among the participants with the reasoning used in those fields. The poor performance using the conventional process on the cost critique is less excusable as all of the participants had some experience conducting cost estimates. Additional trials could help to determine whether the poor reliability of the conventional process could be overcome by more training. The innovative process may simply be more reliable.

Although the Charrette was not intended to test expert performance with either process, the variation in ability should be more carefully controlled. The simplest way to control ability and thus focus the results more upon the process would be to provide more extensive training. As was done in the Autodesk experiment, participants could repeat the trials until no more improvement is detected.

Measurement instruments

Although the collection of time, cost and energy data proceeded relatively smoothly, collection of egress results and space results was insufficient using the conventional process. In part, this is evidence suggesting that SME provides more capable documentation capabilities in comparison to the informal techniques used in practice. However, a more thorough record of the rules that were considered using the conventional process may have been obtainable by videotaping or audio recording the participants. Using such a protocol, the accuracy comparison may have been more successful.

Arbitrariness of the processes

It is important to point out that the manual procedures are quite arbitrary. Professionals from the AEC industry reviewed the manual procedures to provide some validation of the procedures. Nevertheless, the procedures could be made more difficult by providing participants with less systematic forms and procedures, or could be made easier by providing more detailed checklists. In particular the energy calculations are arguably over-simplified in comparison to the computer-based energy calculations. The Charrette was certainly a synthetic experiment that does not carry weight equivalent to a natural experiment.

Use of more sophisticated statistical methods

A larger number of trials would allow the application of more sophisticated statistical methods. The participant population might be validated to be representative of a target population of prospective users, such as practicing architects. Appropriate statistical methods may be similar to those used in empirical artificial intelligence research (Cohen 1995).

A New Test Method

The Charrette Test Method is summarized by a statement of its purpose, a set of definitions, and a set of guidelines.

Purpose

The Charrette Test Method is intended to provide increased reliability and validity in comparison to other commonly used research methods, while remaining simple enough for use in exploratory research. It is a formative method in the sense that it should be used to guide research rather than establish incontrovertible fact. The method was devised to provide empirical evidence for effectiveness of a design process to complement evidence derived from theory and from working models. It is intended to have several advantages over other methods, such as the worked example, the demonstration, and the trial, by:

- employing multiple trials to achieve high reliability;
- providing a clear experimental protocol that can be repeated at different times and by different researchers to achieve high reliability;
- employing objective measurements to avoid a bias that results from the initial researchers being the sole determinants of whether the design methods can be applied effectively; and
- employing a comparison between two processes to achieve greater validity for arguments for improvements over conventional practice.

Guidelines

When conducting a charrette, the following steps should be followed:

1) Prepare clear propositions that will be examined in the charrette.
2) Devise two or more processes for performing the same task, one to be designated the innovative process and one to be designated the conventional process. Formalize the two processes to an extent such that clear measurements of performance may be taken. The innovative process may combine several sub-tasks in a way that the conventional process does not normally combine them. Care must be taken to assure that the formalized conventional practice approximates actual practice.
3) Develop clear, quantifiable measurements of participants' performance, such as time expended, variation from correct values, and variation in results among trials.
4) Define the task so that it can be accomplished by either process in a predictable and limited amount of time.
5) Refine the software and its user interfaces to such an extent that they do not unduly bias the results through software crashes, loss of data and user frustration with the hardware and software.
6) Design the test to control for variation in users by using within-subject testing and at least three participants. Allow time delays between trials by the same participant to lessen a bias in the measurements toward the second trial due to learning by the participant.
7) Select participants who are representative of a target class of users. Provide training sufficient to enable participants to handle the most difficult problem in the test task.
8) Conduct the test. Collect and record measurements.
9) Analyze the test results to reveal evidence for the propositions. Simple averages of performance may be used to compare speed of the two processes. Accuracy may be studied by computing variability of results.

10) Refine the software so that it and the test protocol may be distributed to other researchers.

The Contribution of the Charrette Test Method

Although much study remains to establish reliability and validity, the Cyclotron Suite Charrette clearly establishes that the Charrette Test Method can be employed in research in computing methods for design. The Charrette Test Method is a practical and convincing way of gathering evidence for propositions regarding design processes and new computing methods for design. It can provide evidence at an early stage in research, address a range of propositions, and complement other research methods. It confirms the proposition that:

3) A method can be devised to provide empirical evidence for the effectiveness of software prototypes for computing methods for design. The method must be justified by theory and it must be usable in academic research.

Strength of evidence

In comparison with worked examples, demonstrations and trials, the Charrette Test Method provides stronger and more convincing evidence. It more closely approximates field tests than do theoretical arguments. It obtains evidence from outside sources instead of relying heavily upon evaluations by the primary researchers. Rather than providing an isolated and ungrounded rating like the demonstration method or a trial, it is inherently comparative and provides a relative rating that may eventually be usable in comparing many different research projects. The Charrette Test Method can provide improved validity and reliability over other commonly used research methods.

A formative method

The Cyclotron Suite Charrette has provided evidence regarding the effectiveness of SME in its current state of an early research prototype. Although it would seem absurd to extrapolate the results produced by the Charrette to professional practice, the body of experience with software usability testing suggests that the results can be extremely useful in guiding further research. In addition, the use of several charrettes throughout a research effort may allow researchers to compare results from one version of the research prototypes to another. An initial benchmark may allow the researchers to gain insight into the trend of their research and the potential for success. Thus, the Charrette Test Method appears to be a very useful formative test method for research in computing methods for design.

Raise the bar

The development of the Charrette Test Method "raises the bar" for research in computing methods for design. By providing a means to test additional propositions and by increasing the strength of evidence, the Charrette Test Method may improve the quality of research and accelerate its progress. The results obtained in the Cyclotron Suite Charrette, although not definitive, demonstrate a way to make research projects more rigorous and more objective. The increase both reliability and validity of research conclusions.

References

Alexander, C. 1964. *Notes on the synthesis of form*. Cambridge: Harvard University Press.

Alexander, C., M. Silverstein, S. Angel, S. Ishikawa, and D. Abrams. 1975. *The Oregon experiment*. New York: Oxford University Press.

Autodesk. 1993. *AutoCAD release 13 user's guide.* Sausalito, CA: Autodesk, Inc.

Autodesk. 1995. *AutoCAD release 13 productivity study,* http://www.autodesk.com/products/autocad/acadr13/whitepap/prdstudy.htm. Sausalito, CA: Autodesk, Inc.

Björk, B. C. 1992. A conceptual model of spaces, space boundaries and enclosing structures. *Automation in Construction* 1: 193 - 214. Amsterdam: Elsevier Science Publishers B. V.

California Energy Commission (CEC). 1992. *Energy efficiency standards for residential and nonresidential buildings*. Sacramento, CA: California Energy Commission.

Chinowsky, P. 1991. *The CAADIE project: Applying knowledge-based paradigms to architectural layout generation.* Technical Report No. 54. Stanford, CA: Center for Integrated Facility Engineering.

Cohen, P. R. 1995. *Empirical methods for artificial intelligence*. Cambridge: The MIT Press.

Devon, R., R. S. Engel, R. J. Foster, D. Sathianathan, G. F. W. Turner. 1994. The effect of solid modeling software on 3-D visualization skills. *Engineering design graphics journal* 58(2): 4-11. Washington: American Society for Engineering Education.

Eastman, C. M. 1992b. A data model analysis of modularity and extensibility in building databases. *Building and Environment* 27(2): 135-148. New York: Pergamon Press.

El-Bibany, H. 1992. *Architecture for human-computer design, management and coordination in a collaborative AEC environment*. Technical Report No. 70. Stanford, CA: Center for Integrated Facility Engineering.

Feigenbaum, E., P. McCorduck and H. P. Nii. 1988. *The rise of the expert company.* New York: Random House, Inc.

Fuyama, H. 1992. *Computer assisted conceptual structural design of steel buildings.* Ph. D. dissertation, Department of Civil Engineering, Stanford University.

Garcia, A. C. B., H. C. Howard and M. J. Stefik. 1993. *Active design documents: A new approach for supporting documentation in preliminary routine design.* Technical Report No. 82. Stanford, CA: Center for Integrated Facility Engineering.

Gelman, S. 1994. Silver Bullet: an iterative model for process definition and improvement. *ATandT Technical Journal* 73: 35-45. New York: ATandT.

Grady, R. B. 1994. Successfully applying software metrics. *Computer*, 27(9): 18-25. New York: IEEE Computer Society.

Gero, J. S. 1995. The role of function-behavior-structure models in design. In *Computing in civil engineering*, vol. 1, 294 - 301. New York: American Society of Civil Engineers.

Grissom, S. B., and G. Perlman. 1995. StEP(3D): a standardized evaluation plan for three-dimensional interaction techniques. *International Journal of Human-Computer Studies* 43 (July): 15-41. London: Academic Press Limited.

Hix, D and H. R. Hartson. 1993. *Developing user interfaces: Ensuring usability through product and process*. New York: John Wiley and Sons, Inc.

International Conference of Building Officials (ICBO). 1985. *Uniform building code*. 1985 edition. Whittier, CA: International Conference of Building Officials.

Khedro, T., P. M. Teicholz, M. R. Genesereth. 1993. Agent-based technology for facility design software integration. In *Computing in civil and building engineering,* vol. 1, 385-392. New York: American Society of Civil Engineers.

Luiten, G. T. 1994. *Computer aided design for construction in the building industry*. The Hague: Gijsbertus Theodorus Luiten.

Luiten, G. T., and M. A. Fischer. 1995. *SPACE CAKE: System for project management in civil engineering using computer-aided knowledge engineering.* Working Paper No. 40. Stanford, CA: Center for Integrated Facility Engineering.

Means, R. S. 1995. *R. S. Means assemblies cost data.* Kingston, Mass: R. S. Means Company.

Nielsen, J. 1993. *Usability engineering*. Boston: Academic Press, Inc.

Phan, D. H., and H. C. Howard. 1993. *The primitive-composite (P-C) approach: A methodology for developing sharable object-oriented data representations for facility engineering integration***.** Technical Report No. 85A and 85B. Stanford, CA: Center for Integrated Facility Engineering..

Purcell, A. T., J. S. Gero, H. M. Edwards, E. Matka. 1994. Design fixation and intelligent design aids. In *Artificial Intelligence in Design, '94*, ed. J. S. Gero and F. Sudweeks. Dordrecht, the Netherlands: Kluwer Academic Publishers.

Rush, R. D. ed. 1986. *The building systems integration handbook*. New York: John Wiley and Sons, Inc.

Schön, D. 1983. *The reflective practitioner*. New York: Basic Books, Inc.

Olson, C. J. S. 1993. CAD performance: the issue of speed. *Progressive Architecture*. 74 (5):44-48. Cleveland: Reinhold Publishing.

Sullivan, A C. 1996a. Digitizing acoustic designs. *Architecture* 85(11): 175-179. New York: BPI Communications, Inc.

Vasquez de Velasco, G. and A. Angulo. 1996. Professional decision support clients and instructional knowledge based servers. In *Proceedings of the Third International Conference on Design and Decision Support Systems in Architecture and Urban Planning*, ed. H. J. P. Timmermans. Spa, Belgium.