# The Utility of Simple Prototype Tasks in Understanding and Augmenting Real-World Design Behavior

**John C. Thomas**
IBM T. J. Watson Research Center
PO Box 704 Yorktown Heights New York 10598 USA
jcthomas@us.ibm.com
(01)-914-784-7561

**ABSTRACT**
In this paper I argue that progress in describing, understanding and supporting complex, real-world design may be aided by the adoption of a small set of constrained benchmark tasks that capture the essence of generic, unsolved prototypical problem types that recur in real design problems in a variety of domains. There is, of course, a danger that the study of such problems may become divorced from the real world contexts that they are meant to inform. To avoid this, I suggest some methods to insure an ongoing dialectic between efforts to improve performance in these prototype tasks and the study of and participation in fully contextualized real-world design activities.

**Author Keywords**
Design, design studies, problem solving

**ACM Classification Keywords**
H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

**INTRODUCTION**
Design is a quintessentially human activity. In the space of problem solving activities, it is not only one of the most difficult, it is also one of the most ill-defined in that the initial conditions, the allowable transformations and the desired end-stand are all typically ill-defined. To further complicate matters, real world design today often involves very large teams consisting of participants from many different academic and cultural backgrounds. Design activities may span many weeks, months or even years. The "goodness" of a design in terms of real impact may not even be known when the design is finished or even implemented and situated in the real world setting. For example, a bridge may function quite adequately for years and then twist apart in a storm. Only then is a weakness in the original design made manifest.

Given the complex nature of design and its contextually embedded nature, it may seem absurd to suggest that any value can be gained through the study of small, controlled prototypical problems as being suggested here. Yet, the history of science shows that the study of simplified models *can* be useful provided several pitfalls are avoided. In this paper, I first define what I mean by a prototypical design task, give some examples and then argue how the study of such tasks may inform "real design." I then outline some ways to ensure that the study of such tasks stays relevant to real-world design.

**PROTOTYPICAL DESIGN TASKS**
What I mean by a "prototypical design task" is a task which embodies one or more of the most salient, characteristic, and difficult aspects of real design tasks and yet is stripped away from any kind of complex political and cultural context such as commonly occurs in real design. Further, rather than representing a tangled weave of inter-related issues, such tasks are focused on a single difficulty that repeatedly occurs (though in different guises) in real design tasks. Excellent performance on real design tasks often takes specialized knowledge built up over years of study and experience. Such knowledge is often largely tacit and difficult to extract. It is nearly impossible to "control" in any experimental sense. Prototypical tasks do not require this kind of extensive real world knowledge. They can be explained so that most adults in our culture can understand the problem and have the background knowledge to solve it. Furthermore, prototypical tasks have the advantage of being *cheap.* It might be nice to study the relative effectiveness of three or four proposed design methodologies by having comparable groups work on them. However, a real world design project is an expensive undertaking. Even if we could persuade an organization to pay for four different groups to use four different design methods, what would we learn from the outcome? How could we know whether any differences found were due to chance, due to differences in the skills of the participants in the groups, an interaction between the skills and the particular methods or really reflective of an actual effect of

the method itself? Quite clearly, we could not. Only be studying a fair number of groups in each condition could we be reasonably assured of our inferences.

## SOME EXAMPLES OF PROTOTYPICAL TASKS

In order to explain what is meant by a prototypical task, a few examples should help. First, consider one of the chief difficulties in real design problems; viz., coming to terms with a host of requirements that seem initially to be somewhat contradictory. Often, finding a design that meets a subset of the requirements is fairly easy while finding a design that meets *all* of the requirements at first seems impossible. The following is offered as a prototype tasks that captures most of the aspects of this real phenomenon but in highly simplified form.

"I am thinking of a real world three-dimensional shape. I can turn this shape so that its projection (or shadow) is a square. In another orientation, the projection is a circle. In still another orientation, the projection is an equilateral triangle. What is the shape?"

I would argue that this problem does not require the solver to know a huge amount about a particular domain such as architecture, computer science or electrical engineering. There is an answer to this problem and yet it is not obvious. If we have some technique, or tool, or method that we think may help people deal with the integration of multiple constraints, we can test such a technique, tool or method on this prototypical problem. In doing so, we can test "ordinary people" in large enough numbers and over a short enough period of time to have a reasonable confidence in a found result. Needless to say, there may be useful tools for dealing with thousands of requirements whose benefits may not be obvious with this prototypical problem. And, conversely, once we find something useful for this prototypical problem, we will want to examine its utility in more ecologically valid situations.

Here is another example. "There are two locked boxes, each containing the other's one and only unique key. The only way to open both boxes is by use of these keys. And, yet I am able to open both boxes. How is this possible?"

In this case, the process that the prototypical example ties into is the blockage caused by an unstated assumption. In real world situations, possible solutions are often precluded by just such unstated assumptions. The thesis is that if we can find a reliable way to make people aware of and to challenge their unstated assumptions in this prototypical problem, we have a reasonable chance that this same way will work at least sometimes in more complex, real-world situations.

## SOME BENEFITS AND DANGERS

Some benefits of using prototypical tasks have already been hinted at. Since they are easily explained, small, and self-contained, they can be solved relatively quickly. They lend themselves to simple objective outcome measures although one may also, with a little more work and time, also make interesting qualitative process observations. Since they do not require specialized extensive training, a large number of people can be given the problems meaning that a wide variety of proposed methods, tools and techniques can be tested. Furthermore, because they are relatively divorced from cultural and political entanglements, it is possible for investigators around the world to work collaboratively to determine effective treatments. This would problematic, for example, if one were to compare, say, architectural design firms in three different countries because the training, legal strictures, customs, and so on might differ so much as to overwhelm effects that are due to proposed methods or tools.

The main danger in using prototypical tasks is to imagine that such tasks are *equal to* complex, real-world design tasks and therefore to argue that a positive result found on such a task *implies* that the tool, method, or technique used is good for real world design. Just as drugs are tested first with mice and only if found effective and harmless later tested in clinical trials with humans, so to, we cannot presume that methods which impact prototypical tasks will necessarily be effective in the context of large-scale design problems. Even more likely, *a priori,* there may well be tools dealing with the interactions of design processes that may actually be useful in complex real world design tasks that are not especially valuable in improving performance on any single prototypical task. However, we need not be blind to such implications. If a tool is designed with cross-task or cross-phase coordination in mind, or if its purpose is to deal with the political and social contexts of design, then obviously we would not use simple prototypical tasks sa a proving ground (unless we could be clever enough to design a prototypical tasks that addresses such concerns). On the other hand, if the design rationale for a design method is, say, to help people become aware of their unstated assumptions and we cannot show it to be effective even on the "Two Boxes Problem" it may reasonably increase our skepticism with respect to such a claim.

## GUIDELINES FOR USING PROTOTYPICAL TASKS

Many of the guidelines have already been implicitly mentioned above but basically, prototypical tasks must first of all be grounded in the problems found in real world design. Second, the inferences made on the basis of using such tasks must be limited to those aspects of design the task is meant to reflect and even then, the conclusions are tentative. Nonetheless, through a dialectic of study between prototypical tasks and the in-depth study of real world design efforts, more progress can be made more quickly than by only studying one or the other.

## REFERENCES


Malhotra, A., Thomas, J.C. and Miller, L. (1980). Cognitive processes in design. *International Journal of Man-Machine Studies*, **12**, pp. 119-140.

Carroll, J., Thomas, J.C. and Malhotra, A. (1980). Presentation and representation in design problem solving. *British Journal of Psychology/*,**71 (1)**, pp. 143-155.

Carroll, J., Thomas, J.C. and Malhotra, A. (1979). A clinical-experimental analysis of design problem solving. *Design Studies*, **1 (2)**, pp. 84-92.

Thomas, J.C. (1978). A design-interpretation analysis of natural English. *International Journal of Man-Machine Studies*, **10**, pp. 651-668.

Thomas, J.C. and Carroll, J. (1978). The psychological study of design. *Design Studies,* **1 (1)**, pp. 5-11.

Thomas, J.C., Lee, A. and Danis, C. Enhancing creative design via software tools. *CACM,* **45** (10), 112-115.


**The columns on the last page should be of approximately equal length.**