

The Designing Mind

P. Edwards.

Abstract: Imagination is the key ingredient that makes systems designers successful. A designer must juggle the ability to visualize pictures, changing focal points until the image is suitable for the designer and the situation. It is this balancing process, in which choices are evaluated, that will produce a distinctive design. Two qualities that are important in a designer are the characteristics to simplify a problem and to comprehend a problem from a user's point of view. The interview process is one method to discover whether a designer has the necessary characteristics. Rating an individual by past accomplishments can provide an unclear picture since it is not evident what part the person played in the creative process. The interviewee's perception of past work is valuable and can be obtained by asking what was the most grave mishap and what was the most enjoyable project for which they were responsible.

In my capacity as designer of large-scale business applications I was asked recently, "What makes someone a good designer--as opposed to a good analyst--and how can an interviewer recognize this quality?"

I offered some off-the-cuff ideas, but promised to think about it in more detail.

Well, what is it, what quirk or special gift gives one person the ability to rattle off design ideas that are devastating in their elegance, while colleagues, who are apparently equally bright, labor to produce leaden structures, unwieldy and clumsy?

I knew it was going to be tough to figure out--for as fast as new tests and techniques are developed to measure inherent skills or personality traits, there is reason to dispute them or at least render their significance minor in relation to more traditional processes.

But when wrestling with the difficulties of selecting personnel so vital to an organization, some guidelines are better than none. And surely there must be some ideas that can help inexperienced or unsuccessful designers make better decisions or that can help the successful ones be even more so. With this in mind, I kept looking for those elusive attributes germane to system design that are externally manifested.

Of primary importance, it seemed, was characterizing the thought processes and perspectives of successful designers in the hope of detecting the same in others. Eventually some fragments of the design process began to crystallize.

Imagination is the key. The ability to visualize, invent, concoct, conceptualize, design. The words are good ones, but don't in themselves provide insight as to how these capacities participate in the process.

This ability to conjure up images, shifting focus and emphasis until the result is pleasing or appropriate is surely a powerful and indispensable attribute of the designer. The process of designing is one where options are juggled mentally; expert juggling eventually yields a design that stands out from the others. Alan Polk, product manager for pc-based systems at Applied Data Research, Princeton, N.J., sees this juggling as multidimensional. The good designer is able to modify a number of alternatives simultaneously, fully aware of how altering one will affect the bounds of the others. This process is done mentally until a balance or harmony is reached. "It's more an art than a science," he adds, but the designer must keep an eye on technology and use all the alternatives offered by it. Polk also feels that the good designer has a high "ambiguity tolerance," which permits him to leave noncritical details for later resolution, when appropriate.

Ron Weeks of CINCOM Systems, Cincinnati, (a member of the ANSI/X3/SPARC Committee that developed the three-schema architecture on which CINCOM's TIS relational database is based), sees lateral thinking and brainstorming, popularized by Edward de Bono's series of books on creative thinking, as methods essential to designers, permitting simultaneous consideration of a wide

range of alternatives. The criteria of acceptance and rejection of these alternatives are key to the success of the design selected. Stopping the process too soon may exclude better alternatives not yet considered. Sunday school taught that "the greatest enemy of the best is the second best" and this is especially true in choosing system alternatives.

Herbert Spencer of Western Management Consultants in Vancouver, B.C., a noted designer and developer of a sophisticated query language analyzer, gave me further insight. He says that the task of designing business information systems can be likened to the job of a movie director. Both are charged with selecting and capturing only that which epitomizes the "reality" of the user environment. The systems designer must map this model of reality into a computer system. The good designer does this mapping intuitively--and unfortunately intuition cannot be measured. When mapping, it is important to take the medium of implementation into account. All the manual systems of old are constructed completely differently from the on-line decision support systems of today, but both satisfy the same basic needs, each within its own physical limitations. Good designers respond to the basic need, not a user's view of the proposed system.

SEEING USER'S PROBLEM

This is a point shared strongly by Art Benjamin, president of Online People Inc., Toronto, and father of the on-line prototyping and development tool ACT/1. He speaks of "perspective transfer," which is the ability to perceive a problem through the eyes of the user, or the person who will benefit from the system. The good designer is able to absorb the problem and react to its solution firsthand. In many cases this is in direct contradiction to the user's stated requirements, which are often amateurish second guesses at what he perceives to be a valid solution to his need.

It is not easy to shed the burden of responding to these requirements, especially when they have been emphatically stated. Who hasn't heard a user say, "I need a system that will give me a file of X, Y, and Z." Such a requirement is often a

direct reflection of previous environments, and is not a design to satisfy the real need, which could be to determine the deviations of Z experienced for various subsets of X and Y.

The good designer builds the system in response to the real need, perhaps without recourse to the stated requirement. This eminently human attribute of being able to transcend the technicalities of implementation and to respond to the fundamental needs of the user is what drives the initial juggling of options and it is the user perspective (in part, at least) that acts as arbiter of alternatives formulated in the process.

The ideas being juggled are all drawn from past experience, perhaps modified or extended, but never brand-new. The reservoirs from which the ideas spring are boundless: any magazine, book, or article ever read, any conversation ever held, and any fanciful thought ever dallied over. Anything and everything can play a part--each awaiting its turn for consideration, improvement, or overhaul. Most are summarily dismissed, a few get juggled, and the best survive.

The space shuttle was hardly a new invention despite the impact it has had on space technology. Its success lies in the unique combination of its components, each of which were at some stage a successful evolution of earlier breakthroughs. At a macro level, the shuttle is a product of aircraft design, rocket ships, and sci-fi--Buck Rogers has been shuttling back and forth through space for years.

Shakespeare also knew how to handle original ideas. Most of his plots were based on themes borrowed or stolen. But oh, what manner of work are these, wrought from second-hand tales using well-worn words! Designers who use ideas from the past in successful and skillful new combinations are fulfilling their function without compromise.

Beyond a healthy imagination and the ability to be selective, many attributes have been cited as desirable to the makeup of a good designer. Some, however, are simply attributes of a successful individual, regardless of profession. Almost everyone I spoke to either suggested or concurred with the idea that the good

designer is a well-rounded individual. Some put it in terms of a broad range of interests, varied career history, and so forth. A variety of activities, hobbies, or affiliations with charitable or professional organizations were considered important, in that they add the balance and stimulation so necessary for creativity.

In the same vein, CINCOM's Ron Weeks feels good designers "listen, probe, and above all have patience," while Lib Gibson, director of software development at I.P. Sharp Associates, Toronto, says "the best designers have good personal skills," enabling them to elicit the most useful comments from users. Gibson has also noted a high correlation between design skills, writing skills, and articulation, saying that those who can organize their thoughts to write or speak well are probably also in possession of design ability.

BREAK PROBLEMS DOWN

Joe Williams, manager of technical research at CINCOM, feels an essential attribute of the successful designer is the ability to simplify a problem. This reductive process is again intuitive, he adds, saying it is dangerous to proceed with designers who are intimidated by the complexity of a problem. Every problem can be reduced to various simpler levels, each obeying principles similar to the axioms of algebra. This view is shared by many others.

Winning designers also concur that the good designer is egoless during the creative process. This means that no homegrown idea is sacrosanct in the face of better ideas received, regardless of the source. The president of a micro-based software development company in California rates pride and optimism as the key traits he looks for when interviewing. This sounds contradictory, but pride and confidence in the product are essential to defend it against critics.

For the less secure or improperly motivated designer, the acceptance of good ideas from others is particularly difficult. A complicating factor for managers is that the stubbornness of the less competent designer may appear the same as the superior designer's unwillingness to accept inferior ideas from others.

"When problems begin to form a pattern, it's time to do something about it," says Ken Sloan, a vp at Applied Data Research, and manager of its R&D group in Dallas. It may be that an individual obstinately refutes good ideas in an attempt to maintain independence and a claim to authorship. Or, it may be that a person of vision is surrounded by people incapable of recognizing talent. Either situation is insupportable.

ADR has taken deliberate steps to build a team of people with complementary skills. Even graphics designers are invited to participate in design efforts in an attempt to round out the experience and perspectives of the group. "Ten years ago, I would have said it was possible for someone to successfully design a system singlehandedly," Sloan continues, reflecting on the importance of teamwork. The wisdom taught by experience has dramatically modified his view. He now feels it is rare and even dangerous to have both practicality and global vision embodied in one individual, citing the fact that the most successful implementations have relied upon the combined skills of a small group of dynamic individuals. This willingness to work with and through others is another vital success factor that obviously is not limited to systems designers.

A software designer rated as "brilliant" by her president feels that a "natural curiosity" is an important attribute, among others. Jeffrey Ross, an independent developer of financial software and more recently a financial planner at the Meisenbach Co. in Seattle, sees natural curiosity as a critical factor in determining the way the **designing mind** works. The questions posed reflect the thought processes taking place, and these in turn indicate the way the interrogator is conceptualizing the subject under discussion. No questions asked would seem to indicate no concept visualized. This bears witness to the truism that one can usually tell more about a person from the questions they ask than from the answers they give.

When asked for success criteria, one designer rushes to give two attributes that are often thought to be critical success factors, but in her experience were not: "brilliance" and "hard work" do not a systems designer make, she laments,

although many successful designers have these qualities. She feels, however, that being intelligently lazy is a success factor.

Effort Reduces Workload

The intelligently lazy person is theoretically willing to expend effort to reduce long-term work loads. Marty Goetz, president at ADR, has been hiring and influencing hiring decisions for many years, and sees things from a different, but not necessarily conflicting, perspective. He seeks people with high energy levels who are dedicated to their work. Goetz looks for a competitive spirit and a strong passion for computers as evidenced through education and career choices.

But how does one tell which designer possesses these attributes, even if there is some agreement on what they are? First, there is no surefire test that will even come close to guaranteeing success. But that does not mean there is no help available. The interview process has been around since time immemorial and for a good reason--it can work. It is during the interview that prospective designers can show their true colors, often in subtle ways.

Art Benjamin says it is almost impossible to rate an individual from references to past work. In many cases it is never clear just how much the individual contributed to the creativity of the final product. "But what is important is the way they perceive their past work." He adds, "If they describe their past systems in terms of satisfying user requirements, the chances are you have a designer on your hands; if they start describing inputs and outputs, you are dealing with a programmer or at best an analyst."

This reflection on past work is perhaps the single most unifying factor in all the responses received. It has been expressed again and again, in different forms by different people, both designers and those responsible for hiring them. Jeffrey Ross looks for "enthusiasm" with describing a previous achievement, but not necessarily the "passion for computing" on Marty Goetz's list.

Ken Sloan uses this introspection as a key factor in his interviewing techniques, which he was happy to share. He asks only two questions. The first is, "What

was the most serious disaster you have ever been personally responsible for?"

Respondents are grouped into three categories:

- * Those who haven't experienced any disaster. These are dismissed for lack of experience.
- * Those who describe a disaster but claim no personal responsibility. These are viewed as being not insightful enough to have evaluated the situation properly, or not forceful enough to have persuaded others to take on their ideas for corrective action, or not honest in their assessment of their responsibility.
- * The group who do best on this question are those who can emote and tell a tale of mistakes made and lessons learned.

The second question is the exact inverse: "What was the most enjoyable project for which you have ever been personally responsible?" Nobody should say they have never had responsibility for anything. Even if it was a small part of a larger project, responsibility should be claimed.

Those who claim no responsibility are soon weeded out. The rest are taken to an adjoining room and asked to explain their system to a group of bored-looking, antagonistic technicians who set about picking holes in the design being described (it's a setup).

Respondents are again classified into three groups:

- * Those who fold under the intensity of the probing. They effectively demonstrate that the part they played was minor or affiliatory and are forced to concede that they do not know the system as well as they claimed.
- * Those who respond to the challenge and revel in the opportunity to describe their brainchild to one more audience. Since the audience is deliberately antipathetic, the session may get a little heated, but ultimately the role played by the person being interviewed and the quality of the product become very clear.
- * The third group does not operate well under the process. They were perhaps responsible for great things but are unable to operate in such a hostile environment. Fortunately, there are safeguards in the system to recognize and accommodate this category, if it is warranted.

Admittedly, this approach is tough and demanding, but the environment and conditions under which these software designers will be operating is similar. Perhaps it is unnecessary to go to such extremes in a calm application environment where the competitive edge of the design area is not a matter of life and death to the organization.

Words Vs. Pictures

Ron Weeks uses the interview process as an opportunity to test the ability to communicate with pictures and not just words. While the discussion centers on past work, the interviewer should keep an eye out for doodles and diagrams that expressively encapsulate the core of the design without being unnecessarily complicated or clumsy.

So much for the designer's perception of past work. Herber Spencer feels that programming, especially COBOL, corrupts the mind and limits its opportunities for expansion.

There is a more lenient reading, however. There are "programmers" it seems, but there is no given evolutionary path from programmer to systems designer.

Programmer in this context means a third (or lower) generation language application career programmer. Having encountered many of these, I can agree that few of them should be promoted into the loftier plains of systems analysis, let alone systems design. But the intuitive spirit inherent in the systems designer is not, I venture, crushed by an encounter with procedural languages.

Many interviewers heed personal chemistry in seeking an indicator for success. Lib Gibson says, "Winners have always generated a reaction with me," although she does most of her recruiting internally, allowing a greater exposure to past performance and access to an internal aptitude test.

It is interesting to see the degree to which nontechnical language pervades the vocabulary of these designers. The phrases "work of art," "organic," "Mozartian," "elegant," or one which "sends chills down one's spine," or "brings tears to one's eyes," belie the myth that this is a technical or mechanical process. Some designers go to pains to emphasize the artistic nature of their work in contrast to

the more popular perception. Others work on the assumption that it is purely creative and seek ways to link it with technology in the hope that this will add credence to their art.

The world of systems designers, like any other discipline, is inhabited by geniuses and frauds, discoverers and followers. Winners in the field see designers as interpreters of users' requirements who have an aesthetic sense of "what is right" and a firm grasp of technology. Many see these traits as inborn, cultivated by experience, motivation, and exposure.

A good design is a neat thing. It speeds up development, enlivens the review process, and brings cheer to its implementers. A bad one causes delays, friction, tension, long-lasting dissatisfaction, and usually costs a great deal more.

Systems designers continue to hold a strategic position in every organization that employs them, for the very simple reason that their work has far-reaching effects: from the productivity of programmers who bring the design to life, to (in some cases) the image and success of the entire organization. It is hoped these insights into the **designing mind** prove helpful to those who aspire to this talent, as well as to those who need to recognize it.