

Human-Centered Design Considered Harmful

By Donald A. Norman > Nielsen Norman Group > norman@nngroup.com

Human-centered design has become such a dominant theme in design that it is now accepted by interface and application designers automatically, without thought, let alone criticism. That's a dangerous state—when things are treated as accepted wisdom. The purpose of this essay is to provoke thought, discussion, and reconsideration of some of the fundamental principles of human-centered design. These principles, I suggest, can be helpful, misleading, or wrong. At times, they might even be harmful. Activity-centered design might be superior.

Know Your User

If there is any principle that is sacred to those in the field of user-interface design and human-computer interaction, it is “know your user.” After all, how can one design something for people without a deep, detailed knowledge of those people? The plethora of bad designs in the world would seem to be excellent demonstrations of the perils of ignoring the people for whom the design is intended. Human-centered design was developed to overcome the poor design of software products. By emphasizing the needs and abilities of those who were to use the software, usability and understandability of products has indeed been improved. But despite these improvements, software complexity is still with us. Even companies that pride themselves on following human-centered principles still have complex, confusing products.

If it is so critical to understand the particular users of a product, then what happens when a product is designed to be used by almost anyone in the world? There are many designs that do work well for everyone. This is paradoxical, and it is this very paradox that led me to reexamine common dogma.

Most items in the world have been designed without the benefit of user studies and the methods of human-centered design. Yet they do quite well. Moreover, these include some of the most successful objects of our modern, technological worlds. Consider two representative examples:

The Automobile. People all over the world learn to drive quite successfully with roughly the same configuration of controls. There were no systematic studies of users. Rather, early automobiles tried a variety of configurations, initially copying the seating and steering arrangements of horse-drawn carriages, going through tillers and rods, and then various hand and foot controls until the current scheme evolved.

Everyday Objects. Just look around: kitchen utensils, garden tools, woodworking tools, typewriters, cameras, and sporting equipment vary somewhat from culture to culture, but on the whole, they are more similar than not. People all over the world manage to learn them—and manage quite well.

Activity-Centered Design. Why do these devices work so well? The basic reason is that they were all developed with a deep understanding of the activities that were to be performed: Call this activity-centered design. Many were not even designed in the common sense of the term; rather, they evolved with time. Each new generation of builders slowly improved the product upon the previous generation, based on feedback from their own experiences as well as from their customers. Slow, evolutionary folk design. But even for those devices created by formal design teams, populated with people whose job title was “designer,” these designers used their own understanding of the activities to be performed to determine how the device would be operated. The

users were supposed to understand the task and to understand the designers' intentions.

Activities Are Not the Same as Tasks

Do note the emphasis on the word "activity" as opposed to "task." There is a subtle difference. I use the terms in a hierarchical fashion. At the highest levels are activities, which are composed of tasks, which themselves are composed of actions, and actions are made up of operations. The hierarchical structure comes from my own brand of "activity theory," heavily motivated by early Russian and Scandinavian research. To me, an activity is a coordinated, integrated set of tasks. For example, mobile phones that combine appointment books, diaries and calendars, note-taking facilities, text messaging, and cameras can do a good job of supporting communication activities. This one single device integrates several tasks: looking up numbers, dialing, talking, note taking, checking one's diary or calendar, and exchanging photographs, text messages, and emails. One activity, many tasks.

What Adapts? Technology or People?

The historical record contains numerous examples of successful devices that required people to adapt to and learn the devices. People were expected to acquire a good understanding of the activities to be performed and of the operation of the technology. None of this "tools adapt to the people" nonsense—people adapt to the tools.

Think about that last point. A fundamental corollary to the principle of human-centered design has always been that technology should adapt to people, not people to the technology. Is this really true? Consider the history of the following successful technologies.

The Clock (and Watch). An arbitrary division of the year and day into months, weeks, days, hours, minutes, and seconds, all according to physical principles that differ from psychological or biological ones, now rules our lives. We eat when our watches tell us it is meal time, not when we are hungry. We awake according to the harsh call of the alarm, not

when we are rested. University classes are taught in one-hour periods, three times a week, in ten- to 15-week sessions, not because this is good for education, but because it makes for easier scheduling. The extreme reliance on time is an accidental outgrowth of the rise of the factory and the resulting technological society.

Writing Systems. Consider printing, handwriting, and typing. All are artificial and unnatural. It takes people weeks, months, or even years to learn and become skilled. One successful stylus-based text input device for the Roman alphabet is graffiti—yet another unnatural way of writing.

Musical Instruments. Musical instruments are complex and difficult to manipulate and can cause severe medical problems. Musical notation is modal, so the same representation on a treble clef has a different interpretation on the bass clef. The usability profession has long known of the problems with modes, yet multiple staves have been with us for approximately 1,000 years. It takes considerable instruction and practice to become skilled at reading and playing. The medical problems faced by musicians are so severe that there are books, physicians, Web pages and discussion groups devoted to them. For example, repetitive stress injuries among violinists and pianists are common. Neither the instruments nor the notation would pass any human-centered design review.

Human-Centered versus Activity-Centered: What's the Difference?

What is going on? Why are such non-human-centered designs so successful? I believe there are two reasons, one the activity-centered nature, and two the communication of intention from the builders and designers. Successful devices are those that fit gracefully into the requirements of the underlying activity, supporting them in a manner understandable by people. Understand the activity, and the device is understandable. Builders and designers often have good reasons for the way they constructed the system. If these reasons can be explained, then the task of learning the system is both eased

and made plausible. Yes, it takes years to learn to play the violin, but people accept this because the instrument itself communicates rather nicely the relationship between strings and the resulting sounds. Both the activity and the design are understandable, even if the body must be contorted to hold, finger, and bow the instrument.

Activity-centered design (ACD) is actually very much like human-centered design (HCD). Many of the best attributes of HCD carry over. But there are several differences, first and foremost that of attitude. Attitude? Yes, the mindset of the designer.

The activities, after all, are human activities, so they reflect the possible range of actions, of conditions under which people are able to function, and the constraints of real people. A deep understanding of people is still a part of ACD. But ACD is more: It also requires a deep understanding of the technology, of the tools, and of the reasons for the activities.

Tools Define the Activity: People Really Do Adapt to Technology

HCD asserts as a basic tenet that technology adapts to the person. In ACD, we admit that much of human behavior can be thought of as an adaptation to the powers and limitations of technology. Everything, from the hours we sleep to the way we dress, eat, interact with one another, travel, learn, communicate, play, and relax. Not just the way we do these things, but with whom, when, and the way we are supposed to act, variously called mores, customs, and conventions.

People do adapt to technology. It changes social and family structure. It changes our lives. Activity-centered design not only understands this, but might very well exploit it.

Learn the activity, and the tools are understood. That's the mantra of the human-centered design community. But this is actually a misleading statement, because for many activities, the tools define the activity. Maybe the reality is just the converse: Learn the tools, and the activity is understood.

Consider art, where much time is spent learning

the vagaries of the media. If you want to do oil painting, then you need to understand oil, and brushes, and painting surfaces—even how and when to clean your brush. Is this the tool wagging the dog? Yes, and that is how it always is, how it always shall be. The truly excellent artists have a deep and thorough understanding of their tools and technologies. It isn't enough to have an artistic sense. So too with sports, with cooking, with music, and with all other major activities that use tools.

To the human-centered design community, the tool should be invisible; it should not get in the way. With activity-centered design, the tool is the way.

Why Might HCD Be Harmful?

Why might a human-centered design approach ever be harmful? After all, it has evolved as a direct result of the many problems people have with existing designs, problems that lead to frustration, grief, lost time and effort, and, in safety-critical applications, errors, accidents, and death. Moreover, HCD has demonstrated clear benefits: improved usability, fewer errors during usage, and faster learning times. What, then, are the concerns?

One concern is that the focus upon individual people (or groups) might improve things for them at the cost of making it worse for others. The more something is tailored for the particular likes, dislikes, skills, and needs of a particular target population, the less likely it will be appropriate for others.

The individual is a moving target. Design for the individual of today, and the design will be wrong tomorrow. Indeed, the more successful the product, the more that it will no longer be appropriate. This is because as individuals gain proficiency in usage, they need different interfaces than were required when they were beginners. In addition, the successful product often leads to unanticipated new uses that are very apt not to be well supported by the original design.

But there are more-serious concerns: First, the focus upon humans detracts from support for the activities themselves; second, too much attention to

the needs of the users can lead to a lack of cohesion and added complexity in the design. Consider the dynamic nature of applications, where any task requires a sequence of operations, and activities can comprise multiple, overlapping tasks. Here is where the difference in focus becomes evident, and where the weakness of the focus on the users shows up.

Static Screens versus Dynamic Sequences

We find that work in the kitchen does not consist of independent, separate acts, but of a series of inter-related processes. (Christine Frederick, *The Labor-Saving Kitchen*. 1919.)

The methods of HCD seem centered around static understanding of each set of controls, each screen on an electronic display. But as a result, the sequential operations of activities are often ill-supported. The importance of support for sequences has been known ever since the time-and-motion studies of the early 1900s, as the quotation from Frederick, above, illustrates. Simply delete the phrase “in the kitchen” and her words are still a powerful prescription for design. She was writing in 1919: What has happened in the past 100 years to make us forget this? Note that the importance of support for sequences is still deeply understood within industrial engineering and human factors and ergonomics communities. Somehow, it seems less prevalent within the human-computer interaction community.

Many of the systems that have passed through HCD design phases and usability reviews are superb at the level of the static, individual display, but fail to support the sequential requirements of the underlying tasks and activities. The HCD methods tend to miss this aspect of behavior: Activity-centered methods focus upon it.

Too Much Listening to Users

One basic philosophy of HCD is to listen to users, to take their complaints and critiques seriously. Yes, listening to customers is always wise, but acceding to their requests can lead to overly complex

designs. Several major software companies, proud of their human-centered philosophy, suffer from this problem. Their software gets more complex and less understandable with each revision. Activity-centered philosophy tends to guard against this error because the focus is upon the activity, not the human. As a result, there is a cohesive, well-articulated design model. If a user suggestion fails to fit within this design model, it should be discarded. Alas, all too many companies, proud of listening to their users, would put it in.

Here, what is needed is a strong, authoritative designer who can examine the suggestions and evaluate them in terms of the requirements of the activity. When necessary, it is essential to be able to ignore the requests. This is the goal to cohesion and understandability. Paradoxically, the best way to satisfy users is sometimes to ignore them.

Note that this philosophy applies in the service domain as well. Thus, Southwest Airlines has been successful despite the fact that it ignores the two most popular complaints of its passengers: provide reserved seating and inter-airline baggage transfer. Southwest decided that its major strategic advantage was inexpensive, reliable transportation, and this required a speedy turn-around time at each destination. Passengers complain, but they still prefer the airline.

Sometimes what is needed is a design dictator who says, “Ignore what users say: I know what’s best for them.” The case of Apple Computer is illustrative. Apple’s products have long been admired for ease of use. Nonetheless, Apple replaced its well known, well-respected human interface design team with a single, authoritative (dictatorial) leader. Did usability suffer? On the contrary: Its new products are considered prototypes of great design.

The “listen to your users” produces incoherent designs. The “ignore your users” can produce horror stories, unless the person in charge has a clear vision for the product, what I have called the “conceptual model.” The person in charge must follow that vision and not be afraid to ignore findings. Yes, listen to

customers, but don't always do what they say.

Now consider the method employed by the human-centered design community. The emphasis is often upon the person, not the activity. Look at those detailed scenarios and personas: Honestly, now, did they really inform your design? Did knowing that the persona is that of a 37-year-old, single mother, studying for the MBA at night, really help lay out the control panel or determine the screen layout and, more importantly, to design the appropriate action sequence? Did user modeling, formal or informal, help determine just what technology should be employed?

Show me an instance of a major technology that was developed according to principles of human-centered design, or rapid prototype and test, or user modeling, or the technology adapting to the user. Note the word "major." I have no doubt that many projects were improved, perhaps even dramatically, by the use of these techniques. But name one fundamental, major enhancement to our technologies that came about this way.

Human-centered design does guarantee good products. It can lead to clear improvements of bad ones. Moreover, good human-centered design will avoid failures. It will ensure that products do work, that people can use them. But is good design the goal? Many of us wish for great design. Great design, I contend, comes from breaking the rules, by ignoring the generally accepted practices, by pushing forward with a clear concept of the end result, no matter what. This ego-centric, vision-directed design results in both great successes and great failures. If you want great rather than good, this is what you must do.

There is a lot more to say on this topic. My precepts here are themselves dangerous. We dare not let the entire world of designers follow their instincts and ignore conventional wisdom: Most lack the deep understanding of the activity coupled with a clear conceptual model. Moreover, there certainly are sufficient examples of poor design out in the world to argue against my position. But note, many

of those bad designs are profitable products. Hmm. What does that suggest? Would they be even more profitable had human-centered design principles been followed? Perhaps. But perhaps they might not have existed at all. Think about that.

Yes, we all know of disastrous attempts to introduce computer systems into organizations where the failure was a direct result of a lack of understanding of the people and system. Or was it a result of not understanding the activities? Maybe what is needed is more activity-centered design; Maybe failures come from a shallow understanding of the needs of the activities that are to be supported. Note too that in safety-critical applications, a deep knowledge of the activity is fundamental. Safety is usually a complex system issue, and without deep understanding of all that is involved, the design is apt to be faulty.

Still, I think it's time to rethink some of our fundamental suppositions. The focus upon the human may be misguided. A focus on the activities rather than the people might bring benefits. Moreover, substituting activity-centered for human-centered design does not mean discarding all that we have learned. Activities involve people, and so any system that supports the activities must of necessity support the people who perform them. We can build upon our prior knowledge and experience, both from within the field of HCD, but also from industrial engineering and ergonomics.

All fields have fundamental presuppositions. Sometimes it is worthwhile to reexamine them, to consider the pros and cons and see whether they might be modified or even replaced. Is this the case for those of us interested in human-centered design? We will never know unless we do the exercise.



ABOUT THE AUTHOR *Don Norman wears many hats, including co-founder of the Nielsen Norman Group, professor at Northwestern University, and author; his latest book is Emotional Design. He lives at www.jnd.org.*