HCI Standards: A Mixed Blessing

ELIZABETH BUIE

So you're on a new software project and it's your job to design the user interface. Ah well,

you'll just get the Windows Style Guide and take it from there.

Not so fast.

There's a lot more than that to using standards and guidelines for human–computer interaction (HCI) design. Such guides are a mixed blessing and could even be a mixed curse. Here are a few things it might help you to know.

Why do we want to use HCI standards in the first place?

The obvious answer (surprise!) is to standardize the look and feel of a user interface. We certainly want to standardize the various windows and dialog boxes of a single product, and we may also want to standardize the interfaces of multiple products or systems that people may be using. In addition, we may want to standardize to some extent across all products for a single platform such as Macintosh, Windows, or Motif. Standardization facilitates learning and reduces errors by taking advantage of knowledge the users have gained from other products or from other parts of your product.

There are a few less obvious answers as well to this question of the benefits of HCI standards.

✘ **Incorporate human factors research and "best practice" in HCI.** A large body of empirical research exists on the usability of specific HCI design features, and many recommendations have emerged from these findings and been incorporated into standards documents. Some of these are based on human physical characteristics, especially vision. Most are based on cognitive characteristics—how people process information: perceiving, thinking, learning, understanding, decision making, and so on. A few recommendations are based on affective characteristics—how people feel and react: preferences, excitement/entertainment, æsthetics, and so on. See the sidebar to this article for examples of recommendations based on these three types of factors.

✘ **Smooth the HCI design process.** Reduce the number of look-and-feel decisions that have to be

made during design. The more guidance you can get, the less work it will be for you to do the design. And truly, you don't feel like arguing over whether your menus should be arranged logically or alphabetically, do you?

✗ **Achieve mandated compliance.** Some countries have passed legislation stipulating that software products comply with certain HCI standards. Many US government contracts include requirements that the delivered system or products comply with specific standards (usually US government standards or commercial standards). In these cases, you must use the specified standards and make sure your product complies.

Who writes these rules, anyhow, and what kinds of standards are there?

Lots of people. (Lots of organizations, mostly.) Standards and guidelines come in many flavors (the following with examples).

✶ **International:** standards developed by organizations to reflect agreements among national member organizations, for example,

– **International Organization for Standardization** (ISO) [http://www.iso.ch/], in particular **ISO 9241**, *Ergonomic Requirements for Office Work with Visual Display Terminals* [http://www.iso.ch/cate/cat.html—search on ISO number 9241]

✶ **National:** standards developed by organizations to reflect agreements among companies and other entities within a country, for example,

– **American National Standards Institute** (ANSI) [http://www.ansi.org/]
– **HFES-200, Ergonomics of Software User Interfaces** (in process), developed by the Human Factors and Ergonomics Society [http://hfes.org/]
– **British Standards Institution** (BSI) [http://www.bsi.org.uk/]
– **Ente Nazionale Italiano di Unificazione** (UNI) [http://www.unicei.it/uni]
– **Deutsches Institut für Normung** (DIN) [http://www.din.de/frames/Welcome.html]

✶ **Military and other government agencies**, such as the Department of Defense (DoD), US Military (MIL), National Aeronautics and Space Administration (NASA), for example,

– **MIL-STD-1472D**, Human Engineering Design Criteria for Military Systems, Equipment and Facilities [http://tecnet0.jcte.jcs.mil:9000/htdocs/teinfo/directives/soft/humeng.html] **MIL-STK-1472D**, a HyperCard stack version of this standard [ftp://ftp.cis.ohio-state.edu/pub/hci/1472]
– **NASA-STD-3000**, Man-Systems Integration Standards Handbook [http://cseriac.flight.wpafb.af.mil/products/man_sys.htm]
– **ESD-TR-86-278**, Guidelines for Designing User Interface Software (developed in 1986 for the US Air Force by The MITRE Corporation) [ftp://ftp.cis.ohio-state.edu/pub/hci/Guidelines/]

✶ **Commerce and industry:** style guides for the look and feel of products to run on a specific platform, for example,

– **Macintosh Human Interface Guidelines** (Apple Computer) [http://developer.apple.com/techpubs/mac/HIGuidelines/HIGuidelines-2.html]
– **Common User Access** (CUA) (IBM) [http://www.ibm.com/ibm/hci/designer/docs/cua.html] IBM HCI Guidelines [http://www.ibm.com/IBM/HCI/guidelines/guidelines.html]
– **OSF/Motif Style Guide** (Open Software Foundation) [http://www.premier.sco.com/guide/MotifStyleGuide/en_US/TOC.html]
– **The Windows Interface Guidelines** for Software Design (Microsoft) [http://www.microsoft.com/win32dev/

Elizabeth Buie
Computer Sciences
Corporation
15245 Shady Grove Road
Rockville, MD 20850

uiguide/]

- ✹ **Independent:** guidelines developed by a company or person, either to use along with their own consulting services or to sell in book or electronic form, for example,
  - – **ISO and ANSI Standards for Computer Products:** A Guide to Implementation and Compliance, by Wanda Smith
  - – **Principles and Guidelines in Software User Interface Design,** by Deborah Mayhew
  - – **Human–Computer Interface Design Guidelines,** by C. Marlin Brown
- ✹ **Project:** a standard for a specific project, tailored from other sources.

What's in a standard?

With one exception (which we'll discuss at the end of this section), HCI standards contain statements about the features of the product's HCI design. These statements may be requirements (the HCI shall have some feature if it is to comply with the standard) or recommendations (the HCI should have some feature). In most cases, most or all of the statements are recommendations, because the standard aims to be general enough to cover a wide variety of applications, and there are very few design features that are always needed. HCI standardization is not like, say, telecommunications standardization, because the appropriate features of a UI often depend on who the users are, what they're doing, and in what environment they're doing it.

From now on, I'm going to talk about "standards." When I do, you should think "standards, style guides, and guidelines"—whichever kind of document you're using. I'm also going to talk about "recommendations" because that's what most of the statements in the standards are. Just think "requirements, recommendations, guidelines, and design rules"—whatever applies to the standard with which you're working.

In addition to the recommendations, a standard typically includes one or more of the following kinds of information (usually for each recommendation):

- ✘ **Rationale and principles.** Why this recommendation is good design practice. Sometimes this includes the empirical human factors research on which the recommendation is based.
- ✘ **Examples.** How the recommendation might be implemented, sometimes in more than one way.
- ✘ **Exceptions.** Situations and conditions in which the recommendation might not apply.
- ✘ **References.** Sources of additional information about individual recommendations or about the topic of the standard as a whole.

Finally, some standards (particularly those of international, national, or military or government organizations) include a Compliance section. This section prescribes the method to be used to establish that the product complies with the standard.

There is one exception to the declaration that standards always include statements about product features: standards for the HCI engineering process. But these are few and far between (I know of only two, and they're not in final form yet), and they're not really the topic of this discourse. They aren't the problem.

Why are HCI standards a mixed curse?

Two reasons. First, too many projects rely on them to cover too many of their HCI design decisions. This doesn't work. According to Jared Spool, founding principal of the usability consulting firm User Interface Engineering, standards address only a small percentage of the questions that need to be answered during user interface design.

Second (and more dangerous), test houses are starting to use standards as a means of certifying usability. They'll evaluate your product, and if it complies with a selected standard, poof! it's usable. You get a certificate, which you can use to assure potential customers that your product will meet their usability needs. Or if you're an employer buying or building software, you can use it to get the government's occupational safety folks off your back.

Sound good? Think again.

➤ *What if* you needed a Web browser,

Test houses are starting to use standards as a means of certifying usability.

and your vendor had used a database application standard that called for every action to be confirmed by the user before being carried out?

➤ *What if* you contracted for an air-traffic control center (a safety-critical application if ever there was one) and your contractor used a standard aimed at office applications (which assumes that most errors are recoverable)?

➤ *What if* you had commissioned a custom application for your business and your vendor used the right standard but didn't have a usability engineering process, so the strong teamwork among your employees was never discovered and the product doesn't have any interaction to support that—and now you've got workflow bottlenecks where you had none before?

You see, it turns out there's one tiny little thing that HCI standards and guidelines cannot do: ensure a usable product.

In particular, standards cannot address

✳ Design considerations for any specific user population and the work they need to perform (unless they were written for those users or tasks or environment, as are a few highly specific military and government standards)

✳ Issues and constraints imposed by the context of work

✳ The content and structure of the information exchange that the HCI must support.

In short, standards just can't cover the hard part of HCI design. Why not? There are two reasons why not—which just happen to be closely related:

✳ Standards tend to prescribe features of the product, for instance,
   – What the interface should look like
   – How menu items should be organized
   – How objects should be aligned on the screen
   – What colors should not appear adjacent to each other.

✳ You need something else to cover the hard part—an HCI process.

Now wait just a minute, you object—there are standards for the process. You know there are.

You're right, there are. In particular, ISO/DIS 13407, *Human-centred design processes for interactive systems* [http://www.iso.ch/cate/d21197.html], describes the generic HCI design process; and ISO 9241-11, *Ergonomic requirements for office work with visual display terminals - Part 11: Guidance on usability* [http://www.iso.ch/cate/d16883.html], gives some important advice on specifying and measuring usability. These process standards, however, are extremely general and must be tailored to each organization's needs. Without a specific HCI engineering process for your project, even a good process standard cannot guarantee you a good interface.

So, should I be using standards at all?

Yes, absolutely. (See the first topic we discussed, "Why do we need HCI standards?") Just keep them in perspective. Don't rely on them for all (or even most) of your design decisions, and don't let your standards compliance lull you into complacency, thinking that you have thus assured yourself of a product that meets the needs of your users, their tasks, and their work environment.

OK, then how do I use standards?

I'm not going to give a detailed description of the HCI engineering process on this page; in any case, many of the details depend on the project. I'll just give a brief outline here.

1 **Select** which of the available standards you are going to use as your starting point. This is probably your easiest task.

2 **Develop** a project standard by tailoring the standards you selected to use. Identify the specific recommendations that apply to your project and determine how you are going to apply them.

3 **Apply** the recommendations. Refer to them when making HCI design decisions. (Caution: This is not as easy as it might sound.)

4 **Revise and refine** your project standard as necessary to accommodate new information and considerations that may arise during product development. (Note: Revision may occur at any stage after the original development of the

Without a specific HCI engineering process, even a good process standard cannot guarantee you a good interface.

project standard.)

5 **Inspect** the design and the completed product to verify that your HCI actually complies with the project standard.

Clearly, the preceding steps do not cover the entire HCI engineering process. They address only the use of HCI standards.

Where can I find more information on HCI standards?

Several places:

✘ Offices of ISO
[http://www.iso.ch/]
and the various national bodies (for example, the ones mentioned earlier)

✘ The standards column in the *SIGCHI Bulletin*
[http://www.acm.org/sigchi/bulletin/]

of the Association for Computing Machinery (ACM)
[http://www.acm.org/]

✘ More general standards information on the Web. The best I've found is Loughborough University of Technology's guidance on usability standards
[http://info.lut.ac.uk/research/husat/inuse/usabilitystandards.html]

✘ The Usenet newsgroup comp.human-factors
[news:comp.human-factors]

✘ All the Webspaces I listed in the preceding discussion of who writes HCI standards.

I recommend that you get to know the characteristics and limitations of the various standards (at least the ones you're likely to

Bases and Examples of HCI Design Recommendations

Following are some examples of HCI design recommendations that often appear in standards. These examples include recommendations based on human physical, cognitive, and affective characteristics.

Physical Factors

Some of the recommendations in standards for software HCI are based on the physical attributes of human beings. (Hardware standards, conversely—screen reflectivity, for example, or the force required to press a key or mouse button—have a much stronger physical basis. This article addresses only software HCI.)

Here are two examples of physically based HCI recommendations:

❖ **Avoid displaying saturated red text on a saturated blue background (or vice versa).** A condition called "chromostereopsis" makes the text virtually unreadable for most people (example at right). The eye, you see, focuses different wavelengths of light (i.e., colors, whence "chromo") differently. (So does a camera's lens, for that matter, and in fact a number of "apochromatic" lenses have been designed to counteract this.) When red is in focus, blue appears ever-so-slightly fuzzy, and vice versa. This makes red and blue seem to be at different distances (whence "stereo") from the viewer's eye (whence "opsis"). (**Note:** Green is almost as bad as blue.)

Saturated Red Text on a Saturated Blue Background

❖ **Use motion only for getting and *keeping* the user's attention.** Peripheral vision is more sensitive to motion (e.g., animation and blinking) than is foveal vision; that is, movement that you see out of the corner of your eye will tend to distract you from anything else on the screen. (Just look at any Web page with an animated ad banner, and notice how hard it is to keep your eyes on the page content. And you know *they* want your attention!)

Physically based recommendations tend to be stronger advice than cognitively or affectively based ones because the relevant characteristics vary less from person to person. You and I may have rather different likes or learning styles, but virtually everyone with normal color vision experiences chromostereopsis in looking at red on blue.

use). This will give you confidence in which ones to use, how to use them, and what is needed to develop a project standards in your environment.

Standards can be a useful tool for HCI design and evaluation. As with any other tool, the most important thing is to use them properly. ✐

Elizabeth Buie is a Senior Principal Engineer with Computer Sciences Corporation's (CSC) Federal Sector, Civil Group, in Rockville, Maryland. She has worked in the area of user interface design, development, and evaluation since shortly after joining CSC in 1977. Elizabeth has supported contracts with a variety of civilian government agencies in the United States and Europe. She is currently working on projects for the US National Aeronautics and Space Administration, the US Food and Drug Administration, and the Blue Cross and Blue Shield Association. She chairs CSC's internal Special Interest Group on User Interface Design and Technology and provides internal consulting on HCI and Web HCI topics to numerous projects within CSC. Elizabeth has master's degrees in mathematics and in human development.

Elizabeth has been a member since 1992 of the Human Factors and Ergonomics Society's Human-Computer Interaction Standards Committee. This committee has contributed to ISO standards for software ergonomics and usability engineering and is currently developing HFES-200, Human Factors Engineering of Software User Interfaces, as an ANSI-

Cognitive Factors

Cognitive factors (factors in how people process information) drive so many recommendations that it was difficult to decide which ones to choose. Here are two:

❖ **When listing options for user selection (e.g., in a menu or list box), present them in an order that makes sense to the user's task, grouping them if there are more than just a few. If there is no logical order, list the options alphabetically.** For example, a File menu has New and Open together, Page Setup and Print together, and Quit at the bottom; and a font list shows the available fonts in alphabetical order by name. This recommendation takes advantage of the user's understanding of the task, or of some other organizing principle, and facilitates finding items in the list.

❖ **Provide keyboard "shortcuts" or "accelerators" for commonly used functions and menu items.** This allows users who become familiar with the product to use the keyboard for what may be faster access to those functions. For example, Command-S on the Macintosh (Ctrl-S on the PC) activates the "Save" function without bringing up the File menu. (Note: This is different from keyboard navigation via what are often called "mnemonics" in that it provides for immediate activation of the functions without displaying the menus.)

Make sure you tailor cognitively-based recommendations to the specific knowledge, tasks, and work environments of your product's users.

Affective Factors

Affect is, essentially, subjective reaction. It includes emotions, values, preferences, satisfaction—all the stuff it's so hard for many of us to get a handle on. But if we want our users to be satisfied with our products, we have to pay attention to it. Here are two examples of recommendations based on considerations of user affect:

❖ **Design to put users in control of the interaction.** For example, avoid giving the impression that the computer is telling them what to do, and do not use loaded words such as "illegal" in error messages. (Nobody is going to jail for spelling a command name wrong.)

❖ **Provide for some user customization of the æsthetics of the interface.** Examples include Macintosh Appearance controls, Windows Schemes, and Motif Palettes.

Pay attention to affective factors. The effects of neglecting them may be subtle and discovered too late—for example, when you notice that users continue to prefer the old product to the newer, more functional one.

accredited standards developing organization. Elizabeth spoke on HCI standards at the first two National Institute for Standards and Technology (NIST) symposia on usability engineering in government systems.

Elizabeth has been an instigator in the efforts to raise awareness within ACM/SIGCHI of the special issues and challenges that confront usability engineering in government systems. She co-chaired a formal workshop at CHI'95 and special interest group sessions at CHI'94 and CHI'96, and she wrote the workshop report for the *SIGCHI Bulletin.*

Elizabeth has also coauthored papers and given presentations on integrating usability engineering with software engineering, taking the position that integration must have these two disciplines collaborating as equals under system engineering and should not place either discipline within or under the other.