UNIT-1

* Usability of interactive systems: introduction
* Usability goals and measures
* Motivations
* Universal usability
* Goals for our profession managing
* Guidelines, principles and theories

**DEFINING THE USER INTERFACE**

• User interface, design is a subset of a field of study called human-computer interaction (HCI).

• Human-computer interaction is the study, planning, and design of how people and computers work together so that a person's needs are satisfied in the most effective way.

• HCI designers must consider a variety of factors:

1. What people want and expect, physical limitations and abilities people possess,

2. How information processing systems work,

3. What people find enjoyable and attractive.

4. Technical characteristics and limitations of the computer hardware and software must also be considered.

• The user interface is to – the part of a computer and its software that people can see, hear, touch, talk to, or otherwise understand or direct.

• The user interface has essentially two components: input and output.

1. Input is how a person communicates his / her needs to the computer. Some common input components are the keyboard, mouse, trackball, one's finger, and one's voice.
2. Output is how the computer conveys the results of its computations and requirements to the user. Today, the most common computer output mechanism is the display screen, followed by mechanisms that take advantage of a person's auditory capabilities: voice and sound.

• The use of the human senses of smell and touch output in interface design still remain largely unexplored.

• Proper interface design will provide a mix of well-designed input and output mechanisms that satisfy the user's needs, capabilities, and limitations in the most effective way possible.

• The best interface is one that it not noticed, one that permits the user to focus on the information and task at hand, not the mechanisms used to present the information and perform the task

**THE IMPORTANCE OF THE USER INTERFACE**

* A well-designed interface and screen is terribly important to our users. It is their window to view the capabilities of the system.
* It is also the vehicle through which many critical tasks are presented. These tasks often have a direct impact on an organization's relations with its customers, and its profitability.
* A screen's layout and appearance affect a person in a variety of ways. If they are confusing and inefficient, people will have greater difficulty in doing their jobs and will make more mistakes.
* Poor design may even chase some people away from a system permanently. It can also lead to aggravation, frustration, and increased stress

**THE IMPORTANCE OF GOOD DESIGN**

With today's technology and tools, and our motivation to create really effective and usable interfaces and screens, why do we continue to produce systems that are inefficient and confusing or, at worst, just plain unusable? Is it because:

* We don't care?
* We don't possess common sense?
* We don't have the time?
* We still don't know what really makes good design?

But we never seem to have time to find out what makes good design, nor to properly apply it. After all, many of us have other things to do in addition to designing interfaces and screens.

So we take our best shot given the workload and time constraints imposed upon us. The result, too often, is woefully inadequate.

Interface and screen design were really a matter of common sense, we developers would have been producing almost identical screens for representing the real world.

**The Benefits of Good Design**

Poor clarity forced screen users to spend one extra second per screen.

--Almost one additional year would be required to process all screens.

--Twenty extra seconds in screen usage time adds an additional 14 person years.

The benefits of a well-designed screen have also been under experimental scrutiny for many years.

• One researcher, for example, attempted to improve screen clarity and readability by making screens less crowded.

• Separate items, which had been combined on the same display line to conserve space, were placed on separate line sin stead.

• The result screen users were about 20 percent more productive with the less crowded version.

Proper formatting of information on screens does have a significant positive effect on performance.

• In recent years, the productivity benefits of well-designed Web pages have also been scrutinized.

Training costs are lowered because training time is reduced.

Support line costs are lowered because fewer assist calls are necessary.

Employee satisfaction is increased because aggravation and frustration are reduced.

Ultimately, that an organization's customers benefit because of the improved service they receive. Identifying and resolving problems during the design and development process also has significant economic benefits

How many screens are used each day in our technological world?

How many screens are used each day in your organization? Thousands? Millions

Imagine the possible savings. Proper screen design might also, of course, lower the¬ costs of replacing "broken" PCs

**PRINCIPLES OF USER INTERFACE DESIGN**

• An interface must really be just an extension of a person. This means that the system and its software must reflect a person's capabilities and respond to his or her specific needs.

It should be useful, accomplishing some business objectives faster and more efficiently than the previously used method or tool did.

• It must also be easy to learn, for people want to do, not learn to do.

• Finally, the system must be easy and fun to use, evoking a sense of pleasure and accomplishment not tedium and frustration.

• The interface itself should serve as both a connector and a separator

• A connector in that it ties the user to the power of the computer, and a separator in that it minimizes the possibility of the participants damaging one another.

• While the damage the user inflicts on the computer tends to be physical (a frustrated pounding of the keyboard), the damage caused by the computer is more psychological.

• Throughout the history of the human-computer interface, various researchers and writers have attempted to define a set of general principles of interface design.

• What follows is a compilation of these principles. They reflect not only what we know today, but also what we think we know today.

• Many are based on research, others on the collective thinking of behaviorists working with user interfaces.

• These principles will continue to evolve, expand, and be refined as our experience with Gills and the Web increases.

**Usability goals and measures**

Usability goals in user interface design are the desired outcomes or objectives that designers aim to achieve to create an effective and user-friendly interface. These goals help guide the design process and ensure that the final product meets the needs of the users. Here are some common usability goals in user interface design, along with associated measures:

1. **Efficiency:** The goal is to make interactions with the user interface efficient and minimize the time and effort required to accomplish tasks.

Measure: Task completion time, number of steps or clicks required to complete tasks, time to locate and access specific features or information.

1. **Effectiveness:** The goal is to ensure that users can successfully achieve their goals and complete tasks accurately and with confidence.

Measure: Task success rate, error rate, accuracy of task completion, user satisfaction with task outcomes.

1. **Learnability:** The goal is to make the interface intuitive and easy for users to understand and learn how to use, especially for new or first-time users.

Measure: Time to learn (time taken for users to become proficient), success rate of completing basic tasks without prior instruction or assistance, user feedback on ease of learning.

1. **Satisfaction:** The goal is to create a positive and satisfying user experience that meets the users' expectations and preferences.

Measure: User satisfaction surveys, user feedback on satisfaction and perceived ease of use, user ratings or reviews.

1. **Error Prevention and Recovery**: The goal is to minimize errors and provide effective error prevention and recovery mechanisms to help users avoid and overcome mistakes.

Measure: Error rate, error types, user success rate in recovering from errors, user feedback on error prevention and recovery features.

1. **Consistency:** The goal is to ensure consistency in design elements, behavior, and terminology throughout the interface, promoting a sense of familiarity and reducing cognitive load.

Measure: Consistency checklist (evaluating consistency in visual design, interaction patterns, and terminology), user feedback on consistency.

1. **Accessibility:** The goal is to make the interface inclusive and accessible to users with disabilities or impairments, ensuring equal access to information and functionalities.

Measure: Compliance with accessibility standards (e.g., WCAG), user feedback on accessibility features and usability for users with disabilities.

1. **User Engagement:** The goal is to create an interface that is engaging and holds the user's attention, promoting active participation and a positive emotional experience.

Measure: User engagement metrics (e.g., time spent, interaction frequency), user feedback on enjoyment, perceived usefulness, and motivational aspects.

These usability goals and measures work together to create a user-centered interface that is efficient, effective, and enjoyable to use. Evaluating these measures during usability testing and gathering feedback from users can help identify areas for improvement and guide iterative design processes.

**Usability Motivations**

The enormous interest in interface usability arises from the demonstration of the benefits that come from well-designed user interfaces. This increased motivation emanates from designers and managers of consumer electronics who produce mobile devices, e-commerce websites, and social media where excellent user experiences are necessary to succeed in large, highly competitive markets. Usability has gone from desirable to necessary for survival

Similarly, the huge interest in games and entertainment has raised the performance of devices, networks, and user interfaces. The goals are to ensure that game playing is fluid and vivid; that photo, music, and video streaming is fast; and that sharing is graceful and simple. Strong motivations for usability quality come from high functioning professionals who demand excellence in environments such as life critical systems, industrial plants, legal offices, and police agencies.

The spirit of usability excellence is also expected by users of exploratory, creative, and collaborative interfaces as well as diverse socio-technical systems.

* Life-critical systems (Air traffic control, nuclear reactors, power utilities, police & fire dispatch systems, medical equipments)
* Industrial and commercial systems (Banking, insurance, order entry, inventory management, reservation, billing, and point-of-sales systems)
* Office, Home and Entertainment Applications (Word processing, electronic mail, computer conferencing, and video game systems, educational packages, search engines, mobile device,)
* Exploratory, Creative, and Cooperative System (Web browsing, search engines, artist toolkits, architectural design, software development, music composition, and scientific modeling)
* Social – technical System (Voting, health support, identity verification, crime reporting)

**Universal Usability**

Diversity of human abilities, backgrounds, motivations, personalities, cultures, and work styles is a challenge for interface designers.

Understanding of differences between users is vital for participation by broadest set of users. Mobile device use has begun to require for designs that are universal usable. Rethinking interface designs for different situations often results in a better product for all users.

The rethinking covers considerations for users with disabilities, older adults, young users, etc and discussion for hardware and software diversity

1. Variations in physical abilities and physical workplaces
2. Diverse cognitive and perceptual abilities
3. Personality differences
4. Cultural and international diversity
5. Users with disabilities
6. Older adult users
7. Designing for and with children
8. Accommodating Hardware and Software Diversity

1. Variations in physical abilities and physical workplaces:

Accommodating diverse human perceptual, cognitive, and motor abilities is a challenge to every designer. Fortunately, ergonomics researchers and practitioners have gained substantial experience from design projects with automobiles, aircraft, cellphones, and so on. This experience can be applied to the design of user interfaces and mobile devices.

Basic data about human dimensions comes from research in anthropometry (Preedy, 2012). Thousands of measures of hundreds of features of people—male and female, young and adult, European and Asian, underweight and overweight, tall and short—provide data to construct 5- to 95-percentile design ranges. Head, mouth, nose, neck, shoulder, chest, arm, hand, finger, leg, and foot sizes have been carefully cataloged for a variety of populations

Cellphone keypad design parameters—placement, size, distance between keys, and so forth (Section 10.2)—evolved to accommodate differences in users’ physical abilities. People with especially large or small hands may have difficulty using standard cell phones or keyboards, but a substantial fraction of the population is well served by one design. On the other hand, since screen brightness preferences vary substantially, designers often enable users to control this parameter.

The Human Factors Engineering of Computer Workstations standard (HFES, 2007) lists these concerns:

• Worktable and display-support height

• Clearance under work surface for legs

• Work-surface width and depth

• Adjustability of heights and angles for chairs and work surfaces

• Posture—seating depth and angle, backrest height, and lumbar support

• Availability of armrests, footrests, and palm rests

• Use of chair casters

Mobile devices are increasingly being used while walking or driving and in  public spaces, such as restaurants or trains where lighting, noise, movement, and vibration are part of the user experience. Designing for these more fluid environments presents opportunities for design researchers and entrepreneurs

2. Diverse Cognitive and Perceptual Abilities

A vital foundation for interactive-system designers is an understanding of the cognitive and perceptual abilities of the users (Radvansky and Ashcraft, 2013). The journal Ergonomics Abstracts offers this classification of human cognitive processes:

• Short-term and working memory

• Long-term and semantic memory

• Problem solving and reasoning

• Decision making and risk assessment

• Language communication and comprehension

• Search, imagery, and sensory memory

• Learning, skill development, knowledge acquisition, and concept attainment

It also suggests this set of factors affecting perceptual and motor performance:

• Arousal and vigilance

• Fatigue and sleep deprivation

• Perceptual (mental) load

• Knowledge of results and feedback

• Monotony and boredom

• Sensory deprivation

• Nutrition and diet

• Fear, anxiety, mood, and emotion

• Drugs, smoking, and alcohol

• Physiological rhythms

The term intelligence is not included in this list because its nature is controversial and measuring different forms of intelligence is difficult. In any application, background experience and knowledge in the task and interface domains play key roles in learning and performance. Task- or computer skill inventories can be helpful in predicting performance

3.Personality Differences

Some people are eager to use computers and mobile devices, while others find them frustrating. Even people who enjoy using these technologies may have very different preferences for interaction styles, pace of interaction, graphics versus tabular presentations, dense versus sparse data presentation, and so on. A clear understanding of personality and cognitive styles can be helpful in designing interfaces for diverse communities of users.

One evident difference is between men and women, but no clear pattern of gender-related preferences in interfaces has been documented. While the majority of video-game players and designers are young males, some games (such as The Sims™, Candy Crush Saga, and Farmville) draw ample numbers of female players.

Turning from games to productivity tools, there is also a range of reactions to violent terms such as KILL a process or ABORT a program. These and other potentially unfortunate mismatches between the user interface and the users might be avoided by more thoughtful attention to individual differences among users.

For example, some users file thousands of e-mails in a well-organized hierarchy of folders, while others keep them all in the inbox, using search strategies to find what they want later. These distinct approaches may well relate to personality variables, giving designers the clear message that multiple requirements must be satisfied by their designs.

4.Cultural and International Diversity

Another perspective on individual differences has to do with cultural, ethnic, racial, or linguistic background. Users who were raised learning to read Japanese or Chinese will scan a screen differently from users who were raised learning to read English or French. Users from reflective or traditional cultures may prefer interfaces with stable displays from which they select a single item, while users from action-oriented or novelty-based cultures may prefer animated screens and multiple clicks.

More and more is being learned about computer users from different cultures, but user experience designers are still struggling to establish guidelines that are appropriate across multiple languages and cultures (Sun, 2012; Pereira and Baranauskas, 2015).

User-interface design concerns for internationalization include the following:

• Characters, numerals, special characters, and diacriticals

• Left-to-right versus right-to-left versus vertical input and reading

• Date and time formats

• Numeric and currency formats

• Weights and measures

• Telephone numbers and addresses

• Names and titles (Mr., Ms., Mme., M., Dr.)

• Social Security, national identification, and passport numbers

• Capitalization and punctuation

• Sorting sequences

• Icons, buttons, and colors

• Pluralization, grammar, and spelling

• Etiquette, policies, tone, formality, and metaphors

To develop effective designs, companies run usability studies with users from different countries, cultures, and language communities with vastly different language skills and technology access. To promote international efforts to foster successful implementation of information technologies, representatives from around the world meet regularly for the United Nations World Summit on the Information Society.

5.Users with Disabilities

When digital content and services can be flexibly presented in different formats, all users benefit (Horton and Quesenbery, 2014). However, flexibility is most appreciated by users with disabilities who now can access content and services using diverse input and output devices. Blind users may utilize screen readers (speech output such as JAWS or Apple’s VoiceOver) or refreshable braille displays, while low-vision users may use magnification. Users with hearing impairments may need captioning on videos and transcripts of audio, and people with limited dexterity or other motor impairments may utilize speech recognition, eye-tracking, or alternative keyboards or pointing devices. Increasingly, especially on Apple products, these alternate forms of input or output are integrated into technology out of the box (other laptops, tablets, and smart phones have add-on screen reader and magnification capability, and a small number of laptops have built-in eye tracking).

There is a long history of research on how users with perceptual or motor impairments (such as those described above) interact with technology, and research on intellectual or cognitive impairments is now also increasing (Blanck, 2014; Chourasia et al., 2014). In some cases, people with intellectual impairments need transformation of content, but in other cases, no modifications or assistive technologies are needed. Designing for accessibility helps everyone. The same captioning on video that is utilized by users with hearing impairments is also used by users watching video in noisy locations, such as gyms, bars, and airports.

Many accessibility features help with graceful presentation of content in multiple formats, allowing for flexibility in presentation on small screens of mobile devices or with audio output instead of visual output.

For interfaces to be accessible for people with disabilities, they generally need to follow a set of design guidelines for accessibility. The international standards for accessibility come from the Web Accessibility Initiative, a project of the World Wide Web Consortium. The best-known standards are the Web Content Accessibility Guidelines (WCAG); the current version is WCAG 2.0 (since 2008, http://www.w3.org/TR/WCAG20/). There are also other guidelines such as the Authoring Tool Accessibility Guidelines (ATAG) for developer tools and the User Agent Accessibility Guidelines (UAAG) for browsers.

The Americans with Disabilities Act, as interpreted by federal courts and the U.S. Department of Justice, also requires accessibility of state and local government websites as well as those of private companies and organizations that are considered “public accommodations” (stores, museums, hotels, video rental, etc.).

The European Union Mandate 376 (http://www.mandate376.eu/) will require procurement and development of accessible technologies by EU governments and will coordinate with U.S. Section 508, utilizing WCAG 2.0 and enabling developers to easily satisfy both U.S. and EU legal requirements. Prior to EU Mandate 376, many European countries, such as the UK, Italy, and Germany, and other countries around the world, including Australia and Canada, also had information technology accessibility requirements

The United Nations Convention on the Rights of Persons with Disabilities (CRPD, http://www.un.org/disabilities/convention/conventionfull.shtml), an international human rights agreement, also addresses accessible technology. Article 9 of the CRPD calls upon countries to “Promote access for persons with disabilities to new information and communications technologies and systems, including the Internet,” and article 21 encourages countries to “[provide] information intended for the general public to persons with disabilities in accessible formats and technologies appropriate to different kinds of disabilities.”

6.Older Adult Users

Seniority offers many pleasures and all the benefits of experience, but aging can also have negative physical, cognitive, and social consequences. Understanding the human factors of aging can help designers to create user interfaces that facilitate access by older adult users. The benefits include improved chances for productive employment and opportunities to use writing, e-mail, and other computer tools plus the satisfactions of education, entertainment, social interaction, and challenge (Newell, 2011; Czaja and Lee, 2012). Older adults are particularly active participants in health support groups. The benefits to society include increased access to older adults, which is valuable for their experience and the emotional support they can provide to others.

The further good news is that interface designers can do much to accommodate older adult users (Chisnell et al., 2006). Improved user experiences give older adults access to the beneficial aspects of computing and network communication, thus bringing many societal advantages.

As the world’s population ages, designers in many fields are adapting their work to serve older adults, which can benefit all users. Baby boomers have already begun to push for larger street signs, brighter traffic lights, and better nighttime lighting to make driving safer for drivers and pedestrians.

Researchers and designers are actively working on improving interfaces for older adults (Czaja and Lee, 2012). In the United States, the AARP’s Older Wiser Wired initiatives provide education for older adults and guidance for designers. The European Union also has multiple initiatives and research support for computing for older adults.

In summary, making computing more attractive and accessible to older adults enables them to take advantage of technology, enables others to benefit from their participation, and can make technology easier for everyone. For more information on this topic, check out the Human Factors & Ergonomics Society (http://www.hfes.org), which has an Aging Technical Group that publishes a newsletter and organizes sessions at conferences.

7.Children

Another lively community of users is children, whose uses emphasize entertainment and education (Hourcade, 2015). Even pre-readers can use computer controlled toys, music generators, and art tools. As they mature, begin reading, and gain limited keyboard skills, they can use a wider array of desktop applications, web services, and mobile devices (Foss and Druin, 2014). When they become teenagers, they may become highly proficient users who often help their parents or other adults.

This idealized growth path is followed by many children who have easy access to technology and supportive parents and peers. However, many children without financial resources or supportive learning environments struggle to gain access to technology. They are often frustrated with its use and are endangered by threats surrounding privacy, alienation, pornography, unhelpful peers, and malevolent strangers.

For teenagers, the opportunities for empowerment are substantial. They often take the lead in employing new modes of communication, such as text messaging on cellphones, and in creating cultural or fashion trends that surprise even the designers (for example, playing with simulations and fantasy games and participating in web-based virtual worlds). Appropriate design principles for children’s software recognize young people’s intense desire for the kind of interactive engagement that gives them control with appropriate feedback and supports their social engagement with peers (Bruckman et al., 2012; Fails et al., 2014). Designers also have to find the balance between children’s desire for challenge and parents’ requirements for safety.

Designing for younger children requires attention to their limitations. Their evolving dexterity means that mouse dragging, double-clicking, and small targets cannot always be used; their emerging literacy means that written instructions and error messages are not effective; and their low capacity for abstraction means that complex sequences must be avoided unless an adult is involved. Other concerns are short attention spans and limited capacity to work with multiple concepts simultaneously. Designers of children’s software also have a responsibility to attend to dangers, especially in web-based environments, where parental control over access to violent, racist, or pornographic materials is unfortunately necessary. Appropriate information for the education of children about privacy issues and threats from strangers is also a requirement.

8.Accommodating Hardware and Software Diversity

In addition to accommodating different classes of users and skill levels, designers need to support a wide range of hardware and software platforms. The rapid progress of technology means that newer systems may have a hundred or a thousand times greater storage capacity, faster processors, and higher bandwidth networks. However, designers need to accommodate older devices and deal with newer mobile devices that may have low-bandwidth connections and small screens

The challenge of accommodating diverse hardware is coupled with the need to ensure access through many generations of software. New operating systems, web browsers, e-mail clients, and application programs should provide backward compatibility in terms of their user-interface design and file structures. Skeptics will say that this requirement can slow innovation, but designers who plan ahead carefully to support flexible interfaces and self-defining files will be rewarded with larger market shares. For at least the next decade, three of the main technical challenges will be:

* Producing satisfying and effective Internet interaction on high-speed (broadband) and slower (dial-up and some wireless) connections: Some technological breakthroughs have already been made in compression algorithms to reduce file sizes for images, music, animations, and even video, but more are needed. New technologies are needed to enable pre-fetching or scheduled downloads. User control of the amount of material downloaded for each request could also prove beneficial (for example, allowing users to specify that a large image should be reduced to a smaller size, sent with fewer colors, converted to a simplified line drawing, replaced with just a text description, or downloaded at night when Internet charges are perhaps lower).
* Responsive design enabling access to web services from large displays (3200 × 2400 pixels or larger) and smaller mobile devices (1024 × 768 pixels and smaller):Rewriting each webpage for different display sizes may produce the best quality, but this approach is probably too costly and time-consuming for most web providers. Software tools such as Cascading Style Sheets (CSS) allow designers to specify their content in a way that enables automatic conversions for an increasing range of display sizes.
* Supporting easy maintenance of or automatic conversion to multiple languages: Commercial operators recognize that they can expand their markets if they can provide access in multiple languages and across multiple countries. This means isolating text to allow easy substitution, choosing appropriate metaphors and colors, and addressing the needs of diverse cultures.

**Goals for our profession**

Clear goals are useful not only for interface development but also for educational and professional enterprises. Three broad goals seem attainable:

(1) Influencing academic and industrial researchers;

(2) Providing tools, techniques, and knowledge for commercial developers; and

(3) Raising the computer consciousness of the general public.

1.influencing academic and industrial researchers

Early research in human-computer interaction was done largely by introspection and intuition, but this approach suffered from lack of validity, generality, and precision.

The scientific method for interface research, which is based on controlled experimentation, has this basic outline:

• Understanding of a practical problem and related theory

• Lucid statement of a testable hypothesis

• Manipulation of a small number of independent variables

• Measurement of specific dependent variables

• Careful selection and assignment of subjects

• Control for bias in subjects, procedures, and materials

• Application of statistical tests

• Interpretation of results, refinement of theory, and guidance for experimenters

Materials and methods must be tested by pilot experiments, and results must be validated by replication in variant situations.

**Potential research topics**

* Reducing anxiety and fear of computer usage: Although computers are widely used, they still serve only a fraction of the population. Many otherwise competent people resist use of computers. Some older adults avoid helpful computer-based devices, such as bank terminals or word processors, because they are anxious about-or even fearful of-breaking the computer or making an embarrassing mistake.
* Graceful evolution: Although novices may begin their interactions with a computer by using menu selection, they may wish to evolve to faster or more powerful facilities. Methods are needed to smooth the transition from novice to knowledgeable user to expert
* Specification and implementation of interaction: User-interface building tools reduce implementation times by an order of magnitude when they match the task. There are still many situations in which extensive coding in procedural languages must be added. Advanced research on tools to aid interactive-systems designers and implementers might have substantial payoffs in reducing costs and improving quality
* Direct manipulation: Visual interfaces in which users operate on a representation of the objects of interest are extremely attractive
* Input devices: The plethora of input devices presents opportunities and challenges to interface designers. There are heated discussions about the relative merits of the high-precision touch screen; stylus, voice, eyegaze, and gestural input; the mouse; and haptic devices
* Online assistance: Although many interfaces offer some help or tutorial information online, we have only limited understanding of what constitutes effective design for novices, knowledgeable users, and -experts. The role of these aids and of online user communities could be studied to assess effects on user success and satisfaction
* Information exploration: As navigation, browsing, and searching of multimedia digital libraries and the World Wide Web become more common, the pressure for more effective strategies and tools will increase

2.Providing tools, techniques, and knowledge for commercial developers

* Rapid prototyping is easy when using contemporary tools
* Use general or self-determined guideline documents written for specific audiences
* To refine systems, use feedback from individual or groups of users
* User-interface building tools provide support for rapid prototyping and interface development while aiding design consistency, supporting universal usability, and simplifying evolutionary refinement
* Online electronic-mail facilities allow users to send comments directly to the designers. Online user consultants and fellow users can provide prompt assistance and supportive encouragement.

3.Raising the computer consciousness of the general public

* Many novice users are fearful due to experience with poor product design
* Good designs help novices through these fears by being clear, competent, and nonthreatening
* Usability ultimately becomes a question of national priorities. Advocates of electronic voting and other services, promoters of e-healthcare, and visionaries of e-Iearning increasingly recognize the need to influence allocation of government resources and commercial research **agendas**

Guidelines , Principles and Theories

Introduction

User-interface designers have accumulated a wealth of experience and researchers have produced a growing body of empirical evidence and theories, all of which can be organized into:

1.Guidelines: Low-level focused advice about good practices and cautions against dangers.

2. Principles: Middle-level strategies or rules to analyze and compare design alternatives.

3.Theories: High-level widely applicable frameworks to draw on during design and evaluation as well as to support communication and teaching. Theories can also be predictive, such as those for pointing times by individuals or posting rates for community discussions.

Guidelines, principles, and theories, which offer preventive medicine and remedies for these problems, have matured in recent years (Grudin, 2012). Reliable methods for predicting pointing and input times, better social persuasion principles, and helpful cognitive or perceptual theories now shape research and guide design.

**Guidelines**

A guidelines document helps by developing a shared language and then promoting consistency among multiple designers in terminology usage, appearance, and action sequences. It records best practices derived from practical experience or empirical studies, with appropriate examples and counterexamples. The creation of a guidelines document engages the design community in lively discussions about input and output formats, action sequences, terminology, and hardware devices

1. Navigating the interface
2. Organizing the display
3. Getting the user's attention
4. Facilitating data entry

1.Navigating the interface:

Since navigation can be difficult for many users, providing clear rules is helpful. The sample guidelines presented here come from the U.S. government’s efforts to promote the design of informative webpages (National Cancer Institute, 2006), but these guidelines have widespread application. Most are stated positively (“reduce the user’s workload”), but some are negative (“do not display unsolicited windows or graphics”). The 388 guidelines, which offer cogent examples and impressive research support, cover the design process, general principles, and specific rules.

This sample of the guidelines gives useful advice and a taste of their style:

* Standardize task sequences. Allow users to perform tasks in the same sequence and manner across similar conditions.
* Ensure that links are descriptive. When using links, the link text should accurately describe the link’s destination.
* Use unique and descriptive headings.
* Use headings that are distinct from one another and conceptually related to the content they describe.
* Use radio buttons for mutually exclusive choices. Provide a radio button control when users need to choose one response from a list of mutually exclusive options.
* Develop pages that will print properly. If users are likely to print one or more pages, develop pages with widths that print properly.
* Use thumbnail images to preview larger images. When viewing full-size images is not critical, first provide a thumbnail of the image.

The World Wide Web Consortium (W3C) adapted these guidelines (http://www. w3.org/TR/WCAG20/) and organized them into three priority levels, for which it has provided automated checking tools. A few of the accessibility guidelines are:

* Text alternatives: Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols, or simpler language. Time-based media: Provide alternatives for time-based media (e.g., movies or animations). Synchronize equivalent alternatives (such as captions or auditory descriptions of the visual track) with the presentation.
* Distinguishable: Make it easier for users to see and hear content, including separating foreground from background. Color is not used as the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element. Predictable: Make Web pages appear and operate in predictable ways

2.Organizing the display

Display design is a large topic with many special cases. An early influential guidelines document (Smith and Mosier, 1986) offers five high-level goals for data display:

* Consistency of data display. During the design process, the terminology, abbreviations, formats, colors, capitalization, and so on should all be standardized and controlled by use of a dictionary of these items.
* Efficient information assimilation by the user. The format should be familiar to the operator and should be related to the tasks required to be performed with the data. This objective is served by rules for neat columns of data, left justification for alphanumeric data, right justification of integers, lining up of decimal points, proper spacing, use of comprehensible labels, and appropriate measurement units and numbers of decimal digits.
* Minimal memory load on the user. Users should not be required to remember information from one screen for use on another screen. Tasks should be arranged such that completion occurs with few actions, minimizing the chance of forgetting to perform a step. Labels and common formats should be provided for novice or intermittent users.
* Compatibility of data display with data entry. The format of displayed information should be linked clearly to the format of the data entry. Where possible and appropriate, the output fields should also act as editable input fields.
* Flexibility for user control of data display. Users should be able to get the information from the display in the form most convenient for the task on which they are working. For example, the order of columns and sorting of rows should be easily changeable by the users.

3.Getting the user’s attention

Since substantial information may be presented to users, exceptional conditions or time-dependent information must be presented so as to attract attention (Wickens et al., 2012). These guidelines detail several techniques for getting the user’s attention:

---- Intensity. Use two levels only, with limited use of high intensity to draw attention.

---- Marking. Underline the item, enclose it in a box, point to it with an arrow, or use an indicator such as an asterisk, bullet, dash, plus sign, or X.

----Size. Use up to four sizes, with larger sizes attracting more attention.

----Choice of fonts. Use up to three fonts

----Blinking. Use blinking displays (2–4 Hz) or blinking color changes with great care and in limited areas, as it is distracting and can trigger seizures.

----Color. Use up to four standard colors, with additional colors reserved for occasional use.

----Audio. Use soft tones for regular positive feedback and harsh sounds for rare emergency conditions

Novices need simple, logically organized, and well-labeled displays that guide their actions. Expert users prefer limited labels on fields so data values are easier to extract; subtle highlighting of changed values or positional presentation is sufficient. Display formats must be tested with users for comprehensibility.

Audio tones, like the clicks in keyboards or cellphone ring tones, can provide informative feedback about progress. Alarms for emergency conditions do alert users rapidly, but a mechanism to suppress alarms must be provided. If several types of alarms are used, testing is necessary to ensure that users can distinguish between the alarm levels.

4.Facilitating data entry :

Data-entry tasks can occupy a substantial fraction of users’ time and can be the source of frustrating and potentially dangerous errors. Smith and Mosier (1986) offer five high-level objectives as part of their guidelines for data entry (Courtesy of MITRE Corporate Archives: Bedford, MA):

* Consistency of data-entry transactions: Similar sequences of actions speed learning.
* Minimal input actions by user: Fewer input actions mean greater operator productivity and—usually—fewer chances for error. Making a choice by a single mouse selection or finger press, is preferred over typing in a lengthy string of characters. Selecting from a list of choices eliminates the need for memorization, structures the decision-making task, and eliminates the possibility of typographic errors. A second aspect of this guideline is that redundant data entry should be avoided. It is annoying for users to enter the same information in two locations, such as entering the billing and shipping addresses when they are the same. Duplicate entry is perceived as a waste of effort and an opportunity for error.
* Minimal memory load on users: When doing data entry, users should not be required to remember lengthy lists of codes.
* Compatibility of data entry with data display: The format of data-entry information should be linked closely to the format of displayed information, such as dashes in telephone numbers.
* Flexibility for user control of data entry: Experienced users prefer to enter information in a sequence that they can control, such as selecting the color first or size first, when clothes shopping.

**Principles**

While guidelines are low-level and narrowly focused, principles are more fundamental, widely applicable, and enduring. However, they also tend to need more clarification. For example, the principle of recognizing user diversity makes sense to every designer, but it must be thoughtfully interpreted. A preschooler playing a computer game is a long way from a legal librarian searching for precedents for anxious and hurried lawyers. Similarly, a grandmother sending a text message is a long way from a highly trained and experienced air-traffic controller. These examples highlight the differences in users’ background knowledge, frequency of use, and goals as well as in the impact of user errors.

* Determine users' skill levels
* Identify the tasks
* Choose an interaction style
* 8 golden rules of interface design
* Prevent errors
* Ensuring human control while increasing automation

1.Determine users’ skill levels:

All design should begin with an understanding of the intended users, including population profiles that reflect their age, gender, physical and cognitive abilities, education, cultural or ethnic backgrounds, training, motivation, goals, and personality. There are often several communities of users for an interface, especially for web applications and mobile devices, so the design effort is multiplied. Typical user personas—such as nurses, doctors, storekeepers, high school students, or children—can be expected to have various combinations of knowledge and usage patterns. User groups from different countries may each deserve special attention, and regional differences often exist within countries. Other variables that characterize user personas include location (for example, urban versus rural), economic profile, disabilities, and attitudes toward usingtechnology. Users with poor reading skills, limited education, and low motivation require special attention.

For example, a generic separation into novice or first-time, knowledgeable intermittent and expert frequent users might lead to these differing design goals:

1. **Novice or first-time users**. True novice users—for example, bank customers making their first cell phone check deposit—are assumed to know little of the task or interface concepts. By contrast, first-time users are often professionals who know the task concepts well but have shallow knowledge of the interface concepts (for example, a business traveler using a new rental car’s navigation system). Overcoming their uncertainties, via instructions or dialog boxes, is a serious challenge to interface designers. Restricting vocabulary to a small number of familiar, consistently used concept terms is essential. The number of actions should also be small so that novice and first-time users can carry out simple tasks successfully, which reduces anxiety, builds confidence, and provides positive reinforcement. Informative feedback about the accomplishment of each task is helpful, and constructive, specific error messages should be provided when users make mistakes. Carefully designed video demonstrations and online tutorials may be effective.

2. **Knowledgeable intermittent users**. Many people are knowledgeable but intermittent users of a variety of systems (for example, business travelers filing for travel reimbursements). They have stable task concepts and broad knowledge of interface concepts, but they may have difficulty retaining the structure of menus or the location of features. The burden on their memories will be lightened by orderly structure in the menus, consistent terminology, and interfaces that emphasize recognition rather than recall. These features will also help novices and some experts, but the major beneficiaries are knowledgeable intermittent users. Protection from danger is necessary to support relaxed exploration of features and usage of partially forgotten action sequences. These users will benefit from context-dependent help to fill in missing pieces of task or interface knowledge.

**3. Expert frequent users.** Expert “power” users are thoroughly familiar with the task and interface concepts and seek to get their work done quickly. They demand rapid response times, brief and non-distracting feedback, and the shortcuts to carry out actions with just a few clicks or gestures.

Designing for one class of users is relatively easy; designing for several is much more difficult. When multiple user classes must be accommodated, the basic strategy is to permit a multi-layer (sometimes called level-structured or spiral) approach to learning. Novices can be taught a minimal subset of objects and actions with which to get started.

For example, novice users of a cell phone can quickly learn to make/receive calls first, then to use the menus to change ring tones, and later to reset the privacy protections. The multi-layer approach helps users with different skill levels and promotes universal usability (Shneiderman, 2003).

2.Identify the tasks

After carefully drawing the user profile, the designers identify the tasks to be carried out. Every designer would agree that the tasks must be identified first, but too often, the task analysis is done informally or incompletely. Task analysis has a long history (Hackos and Redish, 1998; Wickens et al., 2012), but successful strategies usually involve long hours of observing and interviewing users. This helps designers to understand task frequencies and sequences and make the tough decisions about what tasks to support. Some implementers prefer to include all possible actions in the hope that some users will find them helpful, but this causes clutter. However, mobile app designers are successful because they ruthlessly limited functionality (for example, calendar, contacts, and to-do list) to guarantee simplicity.

High-level task actions can be decomposed into multiple middle-level task actions, which can be further refined into atomic actions that users execute with a single menu selection or other action. Choosing the most appropriate set of atomic actions is a difficult task. If the atomic actions are too small, the users will become frustrated by the large number of actions necessary to accomplish a higher-level task. If the atomic actions are too large and elaborate, the users will need special options to get what they want from the user interface. The relative task frequencies are important in shaping a menu tree. Frequent tasks should be near the top and therefore quick to carry out, while rare tasks are deeper down. Relative frequency of use is one of the bases for making architectural design decisions. For example, in a text editor:

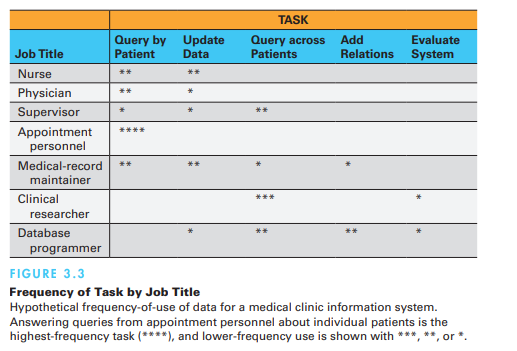
* Frequent actions might be performed by pressing special keys, such as the four arrow keys, Insert, and Delete.
* Less frequent actions might be performed by pressing a single letter plus the Ctrl key or by a selection from a pull-down menu—examples include underscore, bold, and save.
* Infrequent actions or complex actions might require going through a sequence of menu selections or form fill-ins—for example, to change the margins or to revise default printers.

Similarly, cellphone users can assign their close friends and family members to speed-dial numbers so that frequent calls can be made easily by pressing a single key.

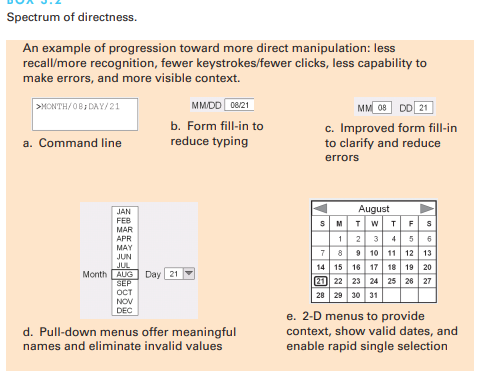
3.Choose an interaction style

When the task analysis is complete and the task objects and actions have been identified, the designer can choose from these five primary interaction styles: direct manipulation, menu selection, form fill-in, command language, and natural language.

**Direct manipulation**: When designers can create a visual representation of the world of action, the users’ tasks can be greatly simplified because direct manipulation of familiar objects is possible. Examples of such visual and tangible user interfaces include the desktop metaphor, drawing tools, photo editing, and games. By pointing at visual representations of objects and actions, users can carry out tasks rapidly and can observe the results immediately (for example, dragging and dropping an icon into a trash can). Context-aware, embedded, natural, and wearable user interfaces often extend the capacity of direct manipulation designs by allowing users to gesture, point, move, or even dance to achieve their goals. Direct manipulation is appealing to novices, is easy to remember for intermittent users, and, with careful design, can be rapid for frequent users.

****

**Navigation and menu selection:** In navigation and menu-selection systems, users review choices, select the one most appropriate to their task, and observe the effect. If the terminology and the meaning of the items are understandable and distinct, users can accomplish their tasks with little learning or memorization and just a few actions. The greatest benefit may be that there is a clear structure to decision making, since all possible choices are presented at one time. This interaction style is appropriate for novice and intermittent users and can be appealing to frequent users if the display and selection mechanisms are rapid. For designers, menu-selection systems require careful task analysis to ensure that all functions are supported conveniently and that terminology is chosen carefully and used consistently. User-interface–building tools that support menu selection provide an enormous benefit by ensuring consistent screen design, validating completeness, and supporting maintenance.



**Form fill-in :**When data entry is required, menu selection alone usually becomes cumbersome, and form fill-in (also called fill in the blanks) is appropriate. Users see a display of related fields, move a cursor among the fields, and enter data where desired. With the form fill-in interaction style, users must understand the field labels, know the permissible values and the data-entry method, and be capable of responding to error messages. Since knowledge of the keyboard, labels, and permissible fields is required, some training may be necessary. This interaction style is most appropriate for knowledgeable intermittent users or frequent users. However, error rates are typically high, training is necessary, and retention may be poor. Error messages and online assistance are hard to provide because of the diversity of possibilities. Command languages and query or programming languages are the domain of expert frequent users, who often derive great satisfaction from mastering a complex language. Powerful advantages include easy scripting and history keeping.

**Command language** :For frequent users, command languages provide a strong feeling of being in control. Users learn the syntax and can often express complex possibilities rapidly without having to read distracting prompts. However, error rates are typically high, training is necessary, and retention may be poor. Error messages and online assistance are hard to provide because of the diversity of possibilities. Command languages and query or programming languages are the domain of expert frequent users, who often derive great satisfaction from mastering a complex language. Powerful advantages include easy scripting and history keeping.

**Natural language :** Increasingly, user interfaces respond properly to arbitrary spoken (for example, Siri on the Apple iPhone) or typed natural-language statements (for example, web search phrases). Speech recognition can be helpful with familiar phrases such as “Tell Catherine that I’ll be there in ten minutes,” but with novel situations users may be frustrated with the results

4.Eight Golden Rules of Interface Design:

This section covers eight golden rules that are applicable in most interactive systems. These principles are derived from experience and refined over two decades. These rules are well-received as a useful guide to students and designers.

1. **Strive for consistency:** This rule is the most frequently violated one, but following it can be tricky because there are many forms of consistency. Consistent sequences of actions should be required in similar situations; identical terminology should be used in prompts, menus, and help screens; and consistent color, layout, capitalization, fonts, and so on should be employed throughout. Exceptions, such as required confirmation of the delete command or no echoing of passwords, should be comprehensible and limited in number.
2. **Cater to universal usability:** Recognize the needs of diverse users and design for plasticity, facilitating transformation of content. Novice-expert differences, age ranges, disabilities, and technology diversity each enrich the spectrum of requirements that guides design. Adding features for novices, such as explanations, and features for experts, such as shortcuts and faster pacing, can enrich the interface design and improve perceived system quality.
3. **Offer informative feedback**: For every user action, there should be system feedback. For frequent and minor actions, the response can be modest, whereas for infrequent and major actions, the response should be more substantial. Visual presentation of the objects of interest provides a convenient environment for showing changes explicitly
4. **Design dialogs to yield closure:** Sequences of actions should be organized into groups with a beginning, middle, and end. Informative feedback at the completion of a group of actions gives operators the satisfaction of accomplishment, a sense of relief, the signal to drop contingency plans from their minds, and a signal to prepare for the next group of actions. For example, e-commerce web sites move users from selecting products to the checkout, ending with a clear confirmation page that completes the transaction.
5. **Prevent errors:** As much as possible, design the system such that users cannot make serious errors; for example, gray out menu items that are not appropriate and do not allow alphabetic characters in numeric entry fields .If a user makes an error, the interface should detect the error and offer simple, constructive, and specific instructions for recovery. For example, users should not have to retype an entire name-address form if they enter an invalid zip code, but rather should be guided to repair only the faulty part. Erroneous actions should leave the system state unchanged, or the interface should give instructions about restoring the state.
6. **Permit easy reversal of actions:** As much as possible, actions should be reversible. This feature relieves anxiety, since the user knows that errors can be undone, thus encouraging exploration of unfamiliar options. The units of reversibility may be a single action, a data-entry task, or a complete group of actions, such as entry of a name and address block.

7.**Support internal locus of control:** Experienced operators strongly desire the sense that they are in charge of the interface and that the interface responds to their actions. Surprising interface actions, tedious sequences of data entries, inability to obtain or difficulty in obtaining necessary information, and inability to produce the action desired all build anxiety and dissatisfaction. Gaines (1981) captured part of this principle with his rule avoid acausality and his encouragement to make users the initiators of actions rather than the responders to actions.

8**.Reduce short term memory load**: The limitation of human information processing in short-term memory (the rule of thumb is that humans can remember "seven plus or minus two chunks" of information) requires that displays be kept simple, multiple-page displays be consolidated, window-motion frequency be reduced, and sufficient training time be allotted for codes, mnemonics, and sequences of actions. Where appropriate, online access to command-syntax forms, abbreviations, codes, and other information should be provided.

5.Prevent errors

The importance of error prevention (the fifth golden rule) is so strong that it deserves its own section. Users of cellphones, e-mail, digital cameras, e-commerce websites, and other interactive systems make mistakes far more frequently than might be expected. One way to reduce the loss in productivity due to errors is to improve the error messages provided by the interface. Better error messages can raise success rates in repairing the errors, lower future error rates, and increase subjective satisfaction. Superior error messages are more specific, positive in tone, and constructive (telling users what to do rather than merely reporting the problem).

Rather than using vague (? or What?) or hostile (Illegal Operation or Syntax Error) messages, designers are encouraged to use informative messages, such as Printer is off, please turn it on or Months range from 1 to 12. Improved error messages, however, are only helpful medicine. A more effective approach is to prevent the errors from occurring. This goal is more attainable than it may seem in many interfaces.

Norman’s analysis provides practical examples and a useful theory. Additional design techniques to reduce errors include the following:

1. Correct actions: Industrial designers recognize that successful products must be safe and must prevent users from dangerously incorrect usage of the products. Airplane engines cannot be put into reverse until the landing gear has touched down, and cars cannot be put into reverse while traveling forward at faster than five miles per hour.
2. Complete sequences: Sometimes an action requires several steps to reach completion. Since users may forget to complete every step of an action, designers may attempt to offer a sequence of steps as a single action.

As another example, users of a text editor can indicate that all section titles are to be centered, set in uppercase letters, and underlined without having to make a series of selections each time they enter a section title. Then if users want to change the title style—for example, to eliminate underlining—a single change will guarantee that all section titles are revised consistently.

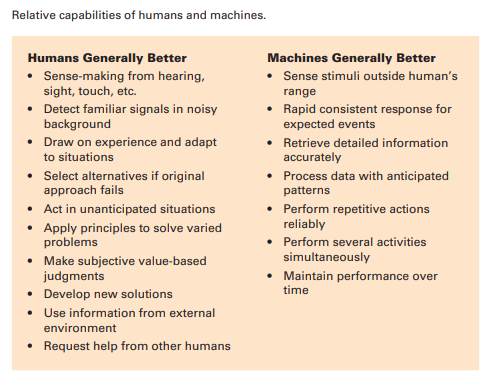
Thinking about universal usability also contributes to reducing errors—for example, a design with too many small buttons may cause unacceptably high error rates among older users or others with limited motor control, but enlarging the buttons will benefit all users.

6.Ensuring human control while increasing automation

The guidelines and principles described in the previous sections are often devoted to simplifying the users’ tasks. Users can then avoid routine, tedious, and error-prone actions and can concentrate on making critical decisions, selecting alternatives if the original approach fails, and acting in unanticipated situations. Users can also make subjective value-based judgments, request help from other humans, and develop new solutions.

Computer system designers have generally been increasing the degree of automation over time as procedures become more standardized and the pressure for productivity grows. With routine tasks, automation is desirable, since it reduces the potential for errors and the users’ workload (Cummings, 2014). However, even with increased automation, informed designers can still offer the predictable and controllable interfaces that users usually prefer. The human supervisory role needs to be maintained because the real world is an open system (that is, a nondenumerable number of unpredictable events and system failures are possible). By contrast, computers constitute a closed system (only a denumerable number of normal and failure situations can be accommodated in hardware and software).

For example, in air-traffic control, common actions include changes to a plane’s altitude, heading, or speed. These actions are well understood and potentially can be automated by scheduling and route-allocation algorithms, but the human controllers must be present to deal with the highly variable and unpredictable emergency situations. An automated system might deal successfully with high volumes of traffic, but what would happen if the airport manager closed a runway because of turbulent weather? The controllers would have to reroute planes quickly. Now suppose that one pilot requests clearance for an emergency landing because of a failed engine, while another pilot reports a passenger with chest pains who needs prompt medical attention. Value-based judgment, possibly with participation from other controllers, is necessary to decide which plane should land first and how much costly and risky diversion of normal traffic is appropriate

****

The goal of design in many applications is to give users sufficient information about current status and activities to ensure that, when intervention is necessary, they have the knowledge and the capacity to perform correctly, even under partial failures (Endsley and Jones, 2004). The U.S. Federal Aviation Agency stresses that designs should place the users in control and automate only to “improve system performance, without reducing human involvement” (U.S. FAA, 2012). These standards also encourage managers to “train users when to question automation.”

The success of Apple’s Siri’s speech recognition and personality-rich voice response system shows that with careful development, useful tools can be developed, but there is little evidence of the benefit of a talking face (Moreno and Mayer, 2007). Robot designers have perennially used human and animal forms as an inspiration, encouraging some researchers to pursue human-like robots for care of older adults or as team members in work situations. These designs attract journalists and have entertainment value but have yet to gain widespread acceptance.

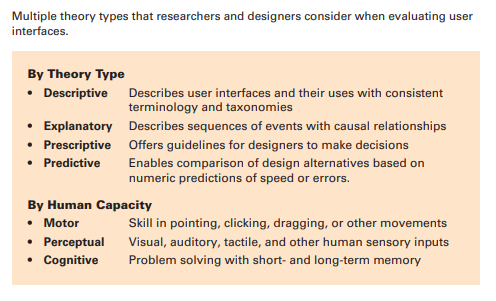
**Theories**

One goal for the discipline of human-computer interaction is to go beyond the specifics of guidelines and build on the breadth of principles to develop tested, reliable, and broadly useful theories. Of course, for a topic as large as user interface design, many theories are needed (Carroll, 2003; Rogers, 2012; Bødker, 2015). Some theories are descriptive; these are helpful in developing consistent terminology and useful taxonomies, for objects and actions, thereby supporting collaboration and training. Other theories are explanatory, describing sequences of events and, where possible, cause and effect, making interventions possible. Still other theories are prescriptive, giving designers clear guidance for their choices. Finally, the most precise theories are predictive, enabling designers to compare proposed designs for execution time, error rates, conversion rates, or trust levels.

Another way to group theories is according to the types of skills involved, such as motor (pointing with a mouse), perceptual (finding an item on a display), or cognitive (planning the sequence of steps needed to pay a bill) (Norman, 2015). Motor skill performance predictions are well established and accurate for predicting keystroking or pointing times. Perceptual theories have been successful in predicting reading times for free text, lists, formatted displays, and other visual or auditory tasks. Cognitive theories, involving short-term, working, and long-term memory, are central to problem solving and play a key role in understanding productivity as a function of system response time .However, predicting performance on complex cognitive tasks (combinations of subtasks) is especially difficult because of the many strategies that might be employed and the many opportunities for going astray.

The ratio of times needed to perform complex tasks between novices and experts or between first-time and frequent users can be as high as 100 to 1.

Web designers have emphasized information-architecture theories with navigation as the keys to user success. Web users can be considered as foraging for information, and therefore the effectiveness of the information scent of links is the issue (Pirolli, 2007).



Taxonomies can be an important part of descriptive and explanatory theories. A taxonomy imposes order by classifying a complex set of phenomena into understandable categories. For example, a taxonomy might be created for different kinds of input devices: direct versus indirect, linear versus rotary, 1-, 2-, 3- or higher-dimensional, and so on (Card et al., 1990). Other taxonomies might cover tasks (structured versus unstructured, novel versus regular) or user-interface styles (direct manipulation, menus, form fill-in).

Any theory that might help designers to predict performance for even a limited range of users, tasks, or designs is a contribution. At the moment, the field is filled with hundreds of theories competing for attention while being refined by their promoters, extended by critics, and applied by eager and hopeful—but skeptical—designers (Carroll, 2003, 2014; Rogers, 2012).

Critics raise two challenges:

1. Theories should be more central to research and practice. A good theory should guide researchers in understanding relationships between concepts and generalizing results. It should also guide practitioners when making design tradeoffs for products. The power of theories to shape design is most apparent in focused theories such as Fitts’s Law; it is more difficult to demonstrate for explanatory theories, whose main impact may be in educating the next generation of designers.

2. Theories should lead rather than lag behind practice. Critics remark that too often a theory is used to explain what has been produced by commercial product designers. A robust theory should predict or at least guide practitioners in designing new products. Effective theories should suggest novel products and services while helping to refine existing ones.

Design-by-levels theories One approach to developing descriptive theories is to separate concepts according to levels. Such theories have been helpful in software engineering and network design. An appealing and easily comprehensible design-by-levels theory for interfaces is the four-level conceptual, semantic, syntactic, and lexical theory (Foley et al., 1995):

1. The conceptual level is the user’s “mental model” of the interactive system. Two mental models for image creation are paint programs that manipulate pixels and drawing programs that operate on objects. Users of paint programs think in terms of sequences of actions on pixels and groups of pixels, while users of drawing programs think in terms of sequences of actions on objects and groups of objects. Decisions about mental models affect each of the lower levels.

2. The semantic level describes the meanings conveyed by the user’s input and by the computer’s output display. For example, deleting an object in a drawing program could be accomplished by undoing a recent action or by invoking a delete-object action. Either action should eliminate only a single object and leave the rest untouched.

3. The syntactic level defines how the user actions that convey semantics are assembled into complete sentences to perform certain tasks. For example, the delete-files action could be invoked by dragging an object to a trash can followed by a click in a confirmation dialog box.

4. The lexical level deals with device dependencies and with the precise mechanisms by which users specify the syntax (for example, a function key or a mouse double-click within 200 milliseconds).

This four-level theory is convenient for designers because its top-down nature is easy to explain, matches the software architecture, and allows for useful modularity during design. Over the years, the success of graphical directmanipulation interfaces has shifted attention up to the conceptual level, which is closest to the task domain (Parush, 2015). For example, designers of personal financial interfaces often use direct-manipulation interfaces. These interfaces build on the users’ mental model of writing checks by showing the image of a check for users to fill in. The same image of a check serves as the query template so users can specify dates, payees, or amounts.

Several researchers have successfully predicted the time required for complex tasks by adding up the times required for each component action. This predictive approach, based on goals, operators, methods, and selection rules (GOMS), decomposes goals into many operators (actions) and then into methods. Users apply selection rules to choose among alternate methods for achieving goals (Card et al., 1983; Baumeister et al., 2000). The GOMS approach works best when the users are expert and frequent users who are working on their own, fully focused on the task, and make no mistakes. Advocates of GOMS have developed software tools to simplify and speed up the modeling process in the hope of increasing usage (John, 2011). Critics complain that broader theories are needed to predict novice user behavior, the transition to proficiency, the rate of errors, and retention over time.

Stages-of-action theories Another approach to forming explanatory theories is to portray the stages of action that users go through in using interactive products such as information appliances, web interfaces, or mobile devices (e.g., music players). Norman (2013) offers seven stages of action, arranged in a cyclic pattern, as an explanatory theory of human-computer interaction:

1. Forming the goal

2. Forming the intention

3. Specifying the action

4. Executing the action

5. Perceiving the system state

6. Interpreting the system state

7. Evaluating the outcome

Some of Norman’s stages correspond roughly to Foley et al.’s (1995) separation of concerns; that is, users form a conceptual intention, reformulate it into the semantics of several commands, construct the required syntax, and eventually produce the lexical token by the action of moving the mouse to select a point on the screen. Norman makes a contribution by placing his stages in the context of cycles of action and evaluation, which take place over seconds and minutes. This dynamic process of action distinguishes Norman’s approach from the other theories, which deal mainly with knowledge that must be in the user’s mind. Furthermore, the seven stages of action lead naturally to identification of the gulf of execution (the mismatch between the user’s intentions and the allowable actions) and the gulf of evaluation (the mismatch between the system’s representation and the user’s expectations).

This theory leads Norman to suggest four principles of good design:

1. The state and the action alternatives should be visible.

2. There should be a good conceptual model with a consistent system image.

3. The interface should include good mappings that reveal the relationships between stages.

4. Users should receive continuous feedback.

Norman places a heavy emphasis on studying errors, describing how errors often occur in moving from goals to intentions to actions and to executions.

The stages-of-action theory helps designers to describe user exploration of an interface (Polson and Lewis, 1990). As users try to accomplish their goals, there are four critical points where user failures can occur:

(1) users may form inadequate goals,

(2) users might not find the correct interface object because of an incomprehensible label or icon,

(3) users may not know how to specify or execute a desired action, and

(4) users may receive inappropriate or misleading feedback.

Refinements of the stages-of-action theory have been developed for other domains. For example, information seeking has been characterized by these stages:

* Recognize
* accept the information problem
* formulate and
* express the query, then
* examine the results,
* reformulate the problem, and
* use the results (Marchionini and White, 2007).

Of course, there are variations with users skipping stages or going back to earlier stages, but the model helps guide designers and users.

Commercial website designers know the benefit of a clear stages-of-action theory in guiding anxious users through a complex process. For example, the Amazon.com website converts the potentially confusing checkout process into a comprehensible four-stage process:

(1) Sign-in,

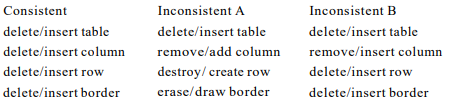
(2) Shipping & Payment,

(3) Gift-Wrap, and

(4) Place Order. Users can simply move through these four stages or back up to previous stages to make changes. Amazon.com also recognizes the need for a frequent user shortcut, the 1-click purchase, for products such as a Kindle book.

Consistency theories

An important goal for designers is a consistent user interface. The argument for consistency is that if terminology for objects and actions is orderly and describable by a few rules, users will be able to learn and retain them easily. This example illustrates consistency and two kinds of inconsistency (A illustrates lack of consistency, and B shows consistency except for a single violation):



Each of the actions in the consistent version is the same, whereas the actions vary for inconsistent version A. The inconsistent action verbs are all acceptable, but their variety suggests that they will take longer to learn, will cause more errors, will slow down users, and will be harder for users to remember. Inconsistent version B is somehow more startling, because there is a single unpredictable inconsistency; it stands out so dramatically that this language is likely to be remembered for its peculiar inconsistency

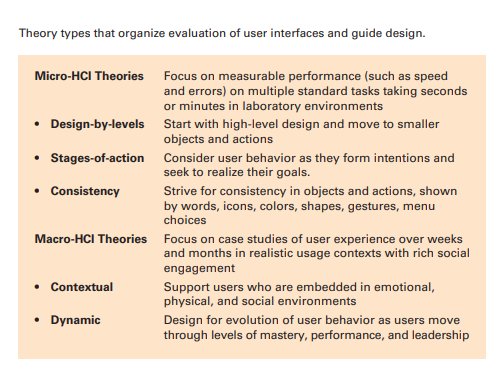
Consistency issues are critical in the design of mobile devices. In successful products, users get accustomed to consistent patterns, such as initiating actions with a left-side button while terminating actions with a right-side button. Similarly, up and down scrolling actions should be done consistently using buttons that are vertically aligned. A frequent problem is the inconsistent placement of the Q and Z characters on phone buttons.

Contextual theories

The design-by-levels, stages-of-action, and consistency theories address the specifics of how objects and actions appear on displays and what actions users take to carry out their tasks. These theories and design aspects might be called micro-HCI, since they cover measurable performance in terms of speed and errors. Micro-HCI is best studied with the scientific methods of experimental and cognitive psychology using 30- to 120-minute controlled experiments and statistical tests for significant differences between groups working on well defined tasks.

Micro-HCI has been and continues to be a great success story, but there is a growing awareness that tightly controlled laboratory studies of isolated phenomena are only one part of the story. The rise of macro-HCI, which emphasizes the user experience, the usage context, and social engagement, has opened up new possibilities for researchers and practitioners.

Macro-HCI thinking leads to different kinds of theories that might best be called contextual, since they consider the emotional, physical, and social contexts of use. Happy users will persevere in the face of frustrations, cope with interruptions from neighbors, and ask for help when they need it.



Contextual theories also address the shift from use of a computer to interaction with a device-rich environment filled with sensors, responsive appliances, display walls, and audio generators.

Contextual theories often emphasize the social environment in which users are engaged with other people who can provide assistance or can be distractions. Advocates of contextual theories believe that the turbulence of actual usage (as opposed to idealized task specifications) means that users have to be more than test subjects—they have to be participants in design processes.

Contextual theories are especially relevant to mobile devices and ubiquitous computing innovations. Such devices are portable or installed in a physical space, and they are often designed specifically to provide place-specific information (for example, a city guide on a portable computer or a museum guide that gives information on a nearby painting). Location information by way of GPS systems enables new services but raises concerns about misuse of tracking information.

While contextual theories emphasize the changes to observation and research, contextual theories can also guide design. If interruptions are an impediment, then users might be given the option of blocking them. If usage outdoors is a requirement, then contrast setting or font sizes should be easily adjustable. If collaboration with others is a high priority, then easy sharing of screens or texting should be possible. A taxonomy of mobile device applications could guide innovators:

• Monitor blood pressure, stock prices, or air quality and give alerts when normal ranges are exceeded.

• Gather information from meeting attendees or rescue team members and spread the action list or current status to all.

• Participate in a large group activity by voting and relate to specific individuals by sending private messages.

• Locate the nearest restaurant or waterfall and identify the details of the current location.

• Capture information or photos left by others and share yours with future visitors.

These five pairs of actions could be tied to a variety of objects (such as photos, annotations, or documents), suggesting new mobile devices and services.

While contextual theories emphasize the changes to observation and research, contextual theories can also guide design. If interruptions are an impediment, then users might be given the option of blocking them. If usage outdoors is a requirement, then contrast setting or font sizes should be easily adjustable. If Theories collaboration with others is a high priority, then easy sharing of screens or texting should be possible. A taxonomy of mobile device applications could guide innovators:

• Monitor blood pressure, stock prices, or air quality and give alerts when normal ranges are exceeded.

• Gather information from meeting attendees or rescue team members and spread the action list or current status to all.

• Participate in a large group activity by voting and relate to specific individuals by sending private messages.

• Locate the nearest restaurant or waterfall and identify the details of the current location.

• Capture information or photos left by others and share yours with future visitors.

These five pairs of actions could be tied to a variety of objects (such as photos, annotations, or documents), suggesting new mobile devices and services

Dynamic theories

A key aspect of macro-HCI is how users evolve over weeks and months, especially as they move from novices to experts, from new customers to frequent buyers, or from readers of Wikipedia to active collaborators or administrators. These theories address design for evolutionary development of skills mastery, behavior change, reputation growth, and leadership capacities. Dynamic theories owe much to the theories of adoption or innovation diffusion (Rogers, 2003), which include five attributes:

1. relative advantage: faster, safer, more error free usage, or cheaper

2. compatibility: fitting for users’ need, consistent with existing values

3. trial-ability: availability to experiment with innovation

4. observability: visibility of innovation to others

5. less complexity: ease of learning and use

These attributes lead to macro-HCI design guidelines, such as suggesting specific user-interface features, combining features to make some more visible than others, and providing informative feedback to users about their usage history. Other macro-HCI design guidelines will suggest ways of training users about features (informing them about new features), rewarding them for successes (showing their progress in reading a book or their score in a game), and sharing their progress with others (notifying friends about an exercise achievement or business associates about a price change).

Dynamic theories deal with long-term (weeks or months) changes in behavior for health (smoking cessation, diet, exercise, or performance in memory games) or education (completing an online course or demonstrating increased familiarity with a body of knowledge).

Dynamic theories are strong among designers of online communities and user-generated content sites. They know that users often move through stages as they gain confidence and a greater sense of responsibility for quality. There are many paths, but a study of Wikipedia contributors (Bryant et al., 2005) suggests at least these stages:

(1) reader of articles related to personal interests,

(2) fixer of mistakes and omissions in familiar topics,

(3) registered user and caretaker for a collection of articles,

(4) author for new articles,

(5) participant in community of authors, and

(6) administrator who is active in governance and future directions

Following these results, the Reader-to-Leader Model described how to design user-interface and social engagement features to promote movement through these stages over a period of weeks or months (Preece and Shneiderman, 2009). At early stages, there are user-interface design guidelines, such as highlighting key features and valuable content, and social engagement design guidelines, such as encouragement from friends, family, and respected authorities. At later stages, there are user-interface design guidelines, such as visible recognition for contributions, and social engagement design guidelines, such as promoting empathy, supporting mentoring, raising trust, and facilitating conflict resolution.

**Advantages and disadvantages of the five primary interaction styles**

