

**HUMAN-COMPUTER INTERACTION** THIRD EDITION DIX FINLAY ABOWD BEALE

## Chapter 7

# Design rules

1

**HUMAN-COMPUTER INTERACTION**

## Design rules

Designing for maximum usability  
 – the goal of interaction design

A designer follows design rules to increase the usability of the eventual software product, classified along two dimensions:

- authority: must be followed or only suggested
- generality: applicable in many design situations or in specific ones

- Principles of usability
  - general understanding
- Standards and guidelines
  - direction for design
- Design patterns
  - potentially generative approach capture and reuse design knowledge

2

**HUMAN-COMPUTER INTERACTION**

## Types of design rules

- Principles
  - abstract design rules
  - low authority
  - high generality
- Standards
  - specific design rules
  - high authority
  - limited application
- Guidelines
  - lower authority
  - more general application

3

**HUMAN-COMPUTER INTERACTION**

## Principles to support usability

**Learnability**  
 the ease with which new users can begin effective interaction and achieve maximal performance

**Flexibility**  
 the multiplicity of ways the user and system exchange information

**Robustness**  
 the level of support provided to the user in determining the successful achievement and assessment of goal-directed behaviour

4

**HUMAN-COMPUTER INTERACTION**

## Principles of learnability

**Predictability**

- determining effect of future actions based on past interaction history
- related to the principle of operation visibility, i.e. how the user is shown what he can do next

**Synthesizability**

- assessing the effect of past actions on the current state
- related to the principle of honesty, either immediate (requiring no further action by the user) or eventual (after explicit user directives); e.g. command line vs visual interface
- global change “the the” -> “the”, eventual honesty would not detect the problem in “... the theorem ...”

5

**HUMAN-COMPUTER INTERACTION**

## Principles of learnability (ctd)

**Familiarity**

- how prior knowledge applies to a new system
- related to the principles of guessability (e.g. you guess how to use a word processor if you know how to use a typewriter) and affordance (shapes of objects suggest how they can be used)

**Generalizability**

- extending specific interaction knowledge to new situations, e.g. copy/cut/paste across different applications
- can be seen as a form of consistency

**Consistency**

- likeness in input/output behaviour arising from similar situations or task objectives

6

## Principles of flexibility

Dialogue initiative

- freedom from system imposed constraints on input dialogue
- related to the principle of system vs. user pre-emptiveness
- in general the latter gives more flexibility to the user than the former but it may not be always desirable (e.g. in a cooperative environment with more than one user, where one user may erase the work of another user; or a pilot doing something dangerous)

Multithreading

- ability of system to support user interaction for more than one task at a time
- related to the principles of concurrent vs. interleaved multithreading and multimodality
- concurrent multithreading allows simultaneous communication pertaining to separate tasks
- interleaved multithreading permits temporal overlap of tasks but at any given time the dialogue is restricted to one task
- multimodality refers to the use of different modes of communication (e.g. open a window with a double click or a keyboard shortcut)

7

## Principles of flexibility (ctd)

Task migratability

- passing responsibility for task execution between user and system
- e.g. a spell checker

Substitutivity

- allowing equivalent values of input and output to be substituted for each other (e.g. a margin is 1.5 inches or 2/3 (8.5-6.5) inches)
- related to the principles of representation multiplicity (e.g. representing temperature in a digital or graph form) and equal opportunity (blurring the distinction between input and output, e.g. entering a value in some cell of a spreadsheet and the system automatically determining the value of some other cell)

8

## Principles of flexibility (ctd)

Customizability

- modifiability of the user interface by the user or the system
- related to the principles of adaptability (user-initiated modification) and adaptivity (system-initiated modification)
- an example of adaptability is to adjust the position of soft buttons or redefine command names
- an example of adaptivity is for a system to detect that the user is a novice or an expert and provide him with a suitable interface

9

## Principles of robustness

Observability

- ability of user to evaluate the internal state of the system from its perceivable representation
- related principles:
  - browsability: allows the user to explore the current internal state of the system via a limited view (e.g. a document in outline view)
  - defaults: the system provides either static (not evolving with the system) or dynamic (evolving during a session) values that reduce the number of physical actions by the user
  - reachability: navigating through observable system states
  - persistence: duration of the effect of a communication act (e.g. a beep vs a raised flag for a new email message received)
  - operation visibility (related to predictability)

10

## Principles of robustness (ctd)

Recoverability

- ability of user to take corrective action once an error has been recognized
- related principles:
  - reachability: allowing the user to reach a desired state from an undesired one
  - forward error recovery: accepting the current state and negotiating towards the desired state (if the effects of interaction are not revocable)
  - backward error recovery: undo the effects of previous interaction to return to a prior state (e.g. an undo button)
  - commensurate effort: if it is difficult to undo an action then it should also be difficult to do the action (e.g. if it is difficult to recover a deleted file it should also be difficult to delete it in the first place)

11

## Principles of robustness (ctd)

Responsiveness

- how the user perceives the rate of communication with the system
- in general, response time should be instantaneous or of short duration
- related to the principle of stability, i.e. similar actions should take the same amount of time for the system to respond (e.g. appearance of pull-down menus immediately after pressing a mouse button)

12

## Principles of robustness (ctd)

### Task conformance

- degree to which system services support all of the user's tasks in a way that the user understands them
- related principles:
  - task completeness: the level to which the system services can be mapped onto all the user tasks
    - but furthermore system services should be general enough to allow the user to define new tasks
  - task adequacy: the task, as it is represented by the user interface, matches the understanding of the user for this task

13

## Standards

- Set by national or international bodies to ensure compliance by a large community of designers
- Standards can apply to either the h/w or the s/w used to build an interactive system but there are some differing characteristics between h/w and s/w that affect the utility of design standards
  - standards for h/w are based on physiology or ergonomics/human factors, and can directly relate to h/w specifications; standards for s/w are based on psychology or cognitive science and have to be worded more vaguely
  - h/w is more difficult and expensive to change than s/w, which by nature is more flexible; consequently, requirements changes for h/w are less frequent than for s/w; as standards are by nature relatively stable, they are more suitable to h/w than s/w

14

## UK MoD standard on human factors for designers of equipment

- Part 1 Introduction
- Part 2 Body Size
- Part 3 Body Strength and Stamina
- Part 4 Workplace Design
- Part 5 Stresses and Hazards
- Part 6 Vision and Lighting
- Part 7 Visual Displays
- Part 8 Auditory Information
- Part 9 Voice Communication
- Part 10 Controls
- Part 11 Design for Maintainability
- Part 12 Systems
- Only part 12 is concerned with the software design process

15

## Sample design standards for display

11.3 Arrangement of displays  
11.3.1 Vertical Grouping. The engine display parameters shall be arranged so that the primary or most important display for a particular engine and airplane (thrust, torque, RPM, etc.) be located at the top of the display group if a vertical grouping is provided. The next most important display parameter shall be positioned under the primary display progressing down the panel with the least important at the bottom.

(a) A typical example of a military standard

5.1 Subdivision of the display area  
In consideration of a simple, fast and accurate visual acquisition, the display area shall be divided into different sub-areas. Such a division should be:

- Input area
- Output area
- Area for operational indications (such as status and alarms)

(b) From German standard DIN 66 234 Part 3 (1984), adapted from Smith [324]

5.15.3.2.1 Standardization  
The content of displays within a system shall be presented in a consistent manner.

(c) From US military standard MIL-STD-1472C, revised (1983), adapted from Smith [324]

16

## ISO standard 9241

- This standard, entitled "Ergonomic requirements for office work with visual display terminals", has 17 parts seven of which are related to h/w and another 7 for s/w
- One component is related to usability specification and offers the following definitions:
  - Usability: The effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments
  - Effectiveness: The accuracy and completeness with which specified users can achieve specified goals in particular environments
  - Efficiency: The resources expended in relation to the accuracy and completeness of goals achieved
  - Satisfaction: The comfort and acceptability of the work system to its users and other people affected by its use
- These definitions can be used to describe explicit metrics for usability and support usability engineering

17

## Guidelines

- Majority of design rules for interactive systems are suggestive and more general guidelines
  - the more abstract a guideline, the more it resembles the principles we examined before and it is suitable more to requirements specification
  - the more specific a guideline (style guide), the more it is suited to detailed design
- There are many published guidelines for interactive system design and each set of guidelines can comprise a long list of items
- Understanding justification for guidelines aids in resolving conflicts

18

## Smith and Mosier guidelines

- Classic example compiled in 1986, comprised of the following basic categories:
  - Data Entry
  - Data Display
  - Sequence Control
  - User Guidance
  - Data Transmission
  - Data Protection
- Each of these categories is broken down in more specific subcategories which contain the particular guidelines

19

## Sample guideline from Smith and Mosier

**1. Data Entry**

**1.1 Position Designation**

**1.1-1 Distinctive Cursor**  
 For position designation on an electronic display, provide a movable cursor with distinctive visual features (shape, blink, etc.).

**Exception** When position designation involves only selection among displayed alternatives, highlighting selected items might be used instead of a separately displayed cursor.

**Comment** When choosing a cursor shape, consider the general content of the display. For instance, an underscore cursor would be difficult to see on a display of underscored text, or on a graphical display containing many other lines.

**Comment** If the cursor is changed to denote different functions (e.g. to signal deletion rather than entry), then each different cursor should be distinguishable from the others.

**Comment** If multiple cursors are used on the same display (e.g. one for alphanumeric entry and one for line drawing), then each cursor should be distinguishable from the others.

**Reference** Whitfield, Ball and Bird, 1983

**See also** 1.1-17 Distinctive multiple cursors  
 4.0-9 Distinctive cursor

20

## Dialog styles

- These are the means by which the user communicates input to the system, including how the system presents the communication device
- There are a number of dialog styles and they can be intermixed in one application
- A number of standards provide guidelines for deciding how to mix dialog styles

21

## Comparison of dialog styles mentioned in guidelines

Smith and Mosier [325]	Mayhew [230]
Question and answer	Question and answer
Form filling	Fill-in forms
Menu selection	Menus
Function keys	Function keys
Command language	Command language
Query language	-
Natural language	Natural language
Graphic selection	Direct manipulation

22

## From abstract guidelines to concrete ones

- To move from abstract guidelines to more specific and automated ones, it is necessary to introduce assumptions about the computer platform on which the interactive system is designed
- Apple has the following abstract guideline for consistency
  - "Effective applications are both consistent within themselves and consistent with one another"
- A more concrete guideline by Apple is the "noun-verb" operation
  - the user first selects an object (the noun) from the visible set on the Desktop and then selects an operation (the verb) to be applied to the object
  - for the sake of consistency, this ordering guideline is to be followed for all operations, involving an operation and an object

23

## Guidelines and programming environments

- GUI systems have published guidelines for adhering to abstract principles of usability in the narrower context of programming
- Some examples are the OpenLook and the Open Software Foundation (OSF) Motif
- It involves the use of toolkits which provide high-level widgets
- Each one of these GUIs has its own "look and feel"
- E.g. the OpenLook style suggests the following for grouping items in the same menu:
  - "Use white space between long groups of controls on menus or in short groups when screen real estate is not an issue"
  - the more options on a menu, the longer will take a user to locate and point to a desired item

24

**Golden rules and heuristics**

- An alternative to principles and guidelines which does not require commitment on the part of the designer to interpret the former or track down the latter
- "Broad brush" design rules, not necessarily applicable to all situations
- But, useful check list for good design
- Better design using these than using nothing!
- Different collections e.g.
  - Nielsen's 10 Heuristics (see Chapter 9)
  - Shneiderman's 8 Golden Rules
  - Norman's 7 Principles

25

**Shneiderman's 8 golden rules**

1. *Strive for consistency* (in action sequences, layout, command use, etc)
2. *Enable frequent users to use shortcuts* (such as abbreviations or macros, to do things faster)
3. *Offer informative feedback*
4. *Design dialogs to yield closure* (completion of a task)
5. *Offer error prevention and simple error handling* (allowing users to recover from their errors)
6. *Permit easy reversal of actions*
7. *Support internal locus of control* (user in control)
8. *Reduce short-term memory load* (by keeping things, such as interfaces, simple)

26

**Norman's 7 principles**

1. *Use both knowledge in the world and knowledge in the head* (user should be able to build an appropriate mental model of what he wants to do)
2. *Simplify the structure of tasks* (provide mental aid, use technology to provide more info, automate the task)
3. *Make things visible: bridge the gulfs of Execution and Evaluation* (interface should make clear what the user can do)
4. *Get the mappings right* (user intentions mapped clearly onto system events)
5. *Exploit the power of constraints, both natural and artificial* (e.g. a jigsaw puzzle)
6. *Design for error* (anticipate human errors)
7. *When all else fails, standardize*

27

**HCI design patterns**

- An approach to reusing knowledge about successful design solutions
- Originated in architecture: Alexander
- A pattern is an invariant solution to a recurrent problem within a specific context.
- Examples
  - Light on Two Sides of Every Room (architecture)
  - Go back to a safe place (HCI)
- Patterns do not exist in isolation but are linked to other patterns in *languages* which enable complete designs to be generated

28

**HCI design patterns (ctd)**

- Characteristics of patterns
  - capture design practice not theory
  - capture the essential common properties of good examples of design
  - represent design knowledge at varying levels: social, organisational, conceptual, detailed
  - embody values and can express what is humane in interface design
  - are intuitive and readable and can therefore be used for communication between all stakeholders
  - a pattern language should be generative and assist in the development of complete designs

29

**The pattern 'go back to a safe place'**

From [www.learni.co.uk](http://www.learni.co.uk)). An example of a toolbar that provides the option to return home and keeps track of the learning journey, providing the option to link back to a safe place.

It is easy to get lost in the tangle of links in a website. People don't use the web like a TV or magazine. They use the web to find what they are looking for and then stop. They may select a wrong link and not be able to find their way back to something relevant. They do not always keep track of where they've been. They may forget where they have been if they are interrupted when using the site. You are more likely to explore a website if you are sure that you can easily get out of an undesired state or space; that assurance engenders a feeling of security. Backtracking out of a long navigation path can be very tedious. Therefore: Always include a way back to a place that acts as a 'vantage point' from where you can reorientate yourself.

This pattern can be used in conjunction with GO BACK ONE STEP and CONTINUE TO NEXT STEP.

30

SLAVEN, JIMMY POLY,  
LECTURE 14: HUMAN-COMPUTER  
INTERACTION

## Summary

**Principles for usability**

- repeatable design for usability relies on maximizing benefit of one good design by abstracting out the general properties which can direct purposeful design
- The success of designing for usability requires both creative insight (new paradigms) and purposeful principled practice

**Using design rules**

- standards and guidelines to direct design activity
- golden rules and heuristics as a less demanding and authoritative alternative
- design patterns as a way to capture design practice and provide a generative structure to support the design process