Chapter 17

User Interface Design

Learning Objective
... Define the basic design principles, user-system interaction, information presentation, user guidance, and interface evaluation

Frederick T Sheldon
Assistant Professor of Computer Science
Washington State University

User interface design

⊗ Designing graphical interfaces for software systems

Objectives
⊗ To suggest some general design principles for user interface design
⊗ To describe direct manipulation interfaces
⊗ To discuss factors to be considered when designing information presentations
⊗ To describe the user support which should be built-in to user interfaces
⊗ To introduce usability attributes and system approaches to system evaluation
Topics covered

- Design principles
- User-system interaction
- Information presentation
- User guidance
- Interface evaluation

The user interface

- System users often judge a system by its interface
- A poorly designed interface can cause a user to make catastrophic errors
- Poor user interface design is the reason why so many software systems are never used
- Focus in this chapter is on graphical user interface design

Graphical user interfaces

- User interfaces which rely on windows, iconic representation of entities, pull-down or pop-up menus and pointing devices.
- Previously called WIMP interfaces - now generally referred to as GUIs.
- The standard form of interface for workstations and high-power personal computers.
GUI characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows</td>
<td>Multiple windows allow different information to be displayed simultaneously on the user’s screen.</td>
</tr>
<tr>
<td>Icons</td>
<td>Icons represent different types of information. On some systems, icons represent files; on others, icons represent processes.</td>
</tr>
<tr>
<td>Menus</td>
<td>Commands are selected from a menu rather than typed in a command language.</td>
</tr>
<tr>
<td>Pointing</td>
<td>A pointing device such as a mouse is used for selecting choices from a menu or indicating items of interest in a window.</td>
</tr>
<tr>
<td>Graphics</td>
<td>Graphical elements can be mixed with text on the same display.</td>
</tr>
</tbody>
</table>

GUI advantages

- They are easy to learn and use.
  - Users without experience can learn to use the system quickly.
- The user may switch quickly from one task to another and can interact with several different applications.
  - Information remains visible in its own window when attention is switched.
- Fast, full-screen interaction is possible with immediate access to anywhere on the screen.

Design principles

- UI design must take account of the needs, experience and capabilities of the system users.
- Users should be involved in the design process and user interface designs should be refined through rapid prototyping.
- There are cognitive factors, such as the size of short-term memory, which user interface designers must be aware of.
User interface design principles

<table>
<thead>
<tr>
<th>Principle</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User familiarity</td>
<td>The interface should use terms and concepts which are drawn from the experience of the anticipated class of user.</td>
</tr>
<tr>
<td>Consistency</td>
<td>The interface should be consistent in that comparable operations should be activated in the same way.</td>
</tr>
<tr>
<td>Minimal surprise</td>
<td>Users should never be surprised by the behaviour of a system.</td>
</tr>
<tr>
<td>Recoverability</td>
<td>The interface should include mechanisms to allow users to recover from their errors.</td>
</tr>
<tr>
<td>User guidance</td>
<td>The interface should incorporate some form of context-sensitive user guidance and assistance.</td>
</tr>
</tbody>
</table>

Design principles

- The interface should be based on user-oriented terms and concepts rather than computer concepts
  - For example, an office system should use concepts such as letters, documents, folders etc. rather than directories, file identifiers, etc.
- The system should display an appropriate level of consistency
  - Commands and menus should have the same format, command punctuation should be similar, etc.

Design principles

- The system should not surprise the user
  - If a command operates in a known way, the user should be able to predict the operation of comparable commands.
- The system should provide some resilience to user errors and allow the user to recover from errors
  - This might include an undo facility, confirmation of destructive actions, ‘soft’ deletes, etc.
- Some user guidance should be supplied
  - Help systems, on-line manuals, etc.
User-system interaction

- Two problems must be addressed in interactive systems design
  - How should information from the user be provided to the computer system?
  - How should information from the computer system be presented to the user?
- User interaction and information presentation may be integrated through a coherent framework such as a user interface metaphor

Direct manipulation

- A direct manipulation interface presents the user with a model of their information space which is modified by direct action e.g. names are changed by typing the new name over them
- A form interface is a simple example of a direct manipulation interface
- GUI's provide some direct manipulation e.g. files can be deleted by moving icons to a trashcan

Form-based interface

<table>
<thead>
<tr>
<th>Title</th>
<th>ISBN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author</td>
<td>Price</td>
</tr>
<tr>
<td>Publisher</td>
<td>Publication date</td>
</tr>
<tr>
<td>Edition</td>
<td>Number of copies</td>
</tr>
<tr>
<td>Classification</td>
<td>Loan status</td>
</tr>
<tr>
<td>Date of purchase</td>
<td>Order status</td>
</tr>
</tbody>
</table>
Direct manipulation advantages

- Users feel in control of the computer and are less likely to be intimidated by it
- User learning time is relatively short
- Users get immediate feedback on their actions so mistakes can be quickly detected and corrected

Direct manipulation problems

- The derivation of an appropriate information space model can be very difficult
- Given that users have a large information space, what facilities for navigating around that space should be provided?
- Direct manipulation interfaces can be complex to program and make heavy demands on the computer system

Interface models

- Desktop metaphor.
  - The model of an interface is a (kind of) desktop with icons representing files, cabinets, etc.
- Control panel metaphor.
  - The model of an interface is a hardware control panel with interface entities including:
    - Buttons
    - Switches
    - Menus
    - Lights
    - Displays
    - Sliders
**Control panel interface**

<table>
<thead>
<tr>
<th>Title</th>
<th>JSD example</th>
<th>Grid</th>
<th>Busy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>JSD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>Network</td>
<td>Units</td>
<td></td>
</tr>
<tr>
<td>Selection</td>
<td>Process</td>
<td>Reduce</td>
<td>Pull</td>
</tr>
</tbody>
</table>

**Menu systems**

- Users make a selection from a list of possibilities presented to them by the system
- The selection may be made by pointing and clicking with a mouse, using cursor keys or by typing the name of the selection
- May make use of simple-to-use terminals such as touch screens

**Advantages of menu systems**

- Users need not remember command names as they are always presented with a list of valid commands
- Typing effort is minimal
- User errors are trapped by the interface
- Context-dependent help can be provided. The user’s context is indicated by the current menu selection
Problems with menu systems

- Actions which involve logical conjunction (and) or disjunction (or) are awkward to represent
- Menu systems are best suited to presenting a small number of choices. If there are many choices, some menu structuring facility must be used
- Experienced users find menus slower than command language

Menu structuring

- Scrolling menus
  - When a choice is not displayed, the menu can be scrolled to reveal more choices. Not practical if there is a very large number of choices
- Hierarchical menus
  - The menus are organized in a hierarchy. Selecting a menu item causes the menu to be replaced by a sub-menu
- Walking menus
  - A menu selection causes another menu to be revealed
- Associated control panels
  - When a menu item is selected, a control panel pops-up offering further options

Walking menus
Command interfaces

- User types commands to give instructions to the system e.g. UNIX
- May be implemented using cheap terminals.
- Easy to process using compiler techniques
- Commands of arbitrary complexity can be created by command combination
- Concise interfaces requiring minimal typing can be created

Problems with command interfaces

- Users have to learn and remember a command language. Command interfaces are therefore unsuitable for occasional users
- Users make errors in command. An error detection and recovery system is required
- System interaction is through a keyboard so typing ability is required

Command languages

- Often preferred by experienced users because they allow for faster interaction with the system
- Not suitable for casual or inexperienced users
- May be provided as an alternative to menu commands (keyboard shortcuts). In some cases, a command language interface and a menu-based interface are supported at the same time
Multiple user interfaces

**Graphical user interface**
- GUI manager

**Command language interface**
- Command language interpreter

Application software system

Information presentation

- **Static information**
  - Initialized at the beginning of a session. It does not change during the session
  - May be either numeric or textual

- **Dynamic information**
  - Changes during a session and the changes must be communicated to the system user
  - May be either numeric or textual

Information display factors

- Is the user interested in precise information or data relationships?
- How quickly do information values change?
  - Must the change be indicated immediately?
- Must the user take some action in response to a change?
- Is there a direct manipulation interface?
- Is the information textual or numeric? Are relative values important?
Information presentation

- Information to be displayed
- Presentation software
- Display

Textual and graphical views

Jan 2842  Feb 2851  Mar 3164  April 2789  May 1273  June 2835

Analogue vs. digital presentation

- Digital presentation
  - Compact - takes up little screen space
  - Precise values can be communicated
- Analogue presentation
  - Easier to get an 'at a glance' impression of a value
  - Possible to show relative values
  - Easier to see exceptional data values
Dynamic information display

- Dial with needle
- Pie chart
- Thermometer
- Horizontal bar

Displaying relative values

<table>
<thead>
<tr>
<th>Pressure</th>
<th>Temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>25</td>
</tr>
<tr>
<td>200</td>
<td>50</td>
</tr>
<tr>
<td>300</td>
<td>75</td>
</tr>
<tr>
<td>400</td>
<td>100</td>
</tr>
</tbody>
</table>

Textual highlighting

The filename you have chosen has been used. Please choose another name

Ch. 17. User interface design

OK  Cancel
Data visualization

- Concerned with techniques for displaying large amounts of information
- Visualization can reveal relationships between entities and trends in the data
- Possible data visualizations are:
  - Weather information collected from a number of sources
  - The state of a telephone network
  - A model of a molecule displayed in 3 dimensions

Color displays

- Color adds an extra dimension to an interface and can help the user understand complex information structures
- Can be used to highlight exceptional events
- Common mistakes in the use of color in interface design include:
  - The use of color to communicate meaning
  - Over-use of color in the display

Color use guidelines

- Don't use too many colors
- Use color coding to support use tasks
- Allow users to control color coding
- Design for monochrome then add color
- Use color coding consistently
- Avoid color pairings which clash
- Use color change to show status change
- Be aware that color displays are usually lower resolution
User guidance

- The user guidance system is integrated with the user interface to help users when they need information about the system or when they make some kind of error.
- User guidance covers:
  - System messages, including error messages
  - Documentation provided for users
  - On-line help
- The help and message system may be integrated

Help and message system

- Application
- Help interface
- Error message system
- Message presentation system
- Help frames
- Error message texts

Error message design

- Error message design is critically important. Poor error messages can mean that a user rejects rather than accepts a system.
- Messages should be polite, concise, consistent and constructive.
- The background and experience of users should be the determining factor in message design.
Design factors in message wording

<table>
<thead>
<tr>
<th>Factor</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context</td>
<td>The user guidance system should be aware of what the user is doing and should adjust the output message to the current context.</td>
</tr>
<tr>
<td>Experience</td>
<td>As users become familiar with a system they become irritated by long, 'meaningful' messages. However, beginners find it difficult to understand short terse statements of the problem. The user guidance system should provide both types of message and allow the user to control message conciseness.</td>
</tr>
<tr>
<td>Skill level</td>
<td>Messages should be tailored to the user's skills as well as their experience. Messages for the different classes of user may be expressed in different ways depending on the terminology which is familiar to the reader.</td>
</tr>
<tr>
<td>Style</td>
<td>Messages should be positive rather than negative. They should use the active rather than the passive mode of address. They should never be insulting or try to be funny.</td>
</tr>
<tr>
<td>Culture</td>
<td>Wherever possible, the designer of messages should be familiar with the culture of the country where the system is sold. There are distinct cultural differences between Europe, Asia and America. A suitable message for one culture might be unacceptable in another.</td>
</tr>
</tbody>
</table>

Nurse input of a name

Please type the patient name in the box then click on OK

Bates, J.

OK  Cancel

Good and bad error responses

<table>
<thead>
<tr>
<th>Error #27</th>
<th>Invalid patient id</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient J. Bates is not known to the system</td>
<td></td>
</tr>
<tr>
<td>Click on Patients for a list of known patients</td>
<td></td>
</tr>
<tr>
<td>Click on Retry to re-input a patient name</td>
<td></td>
</tr>
<tr>
<td>Click on Help for more information</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Patients</th>
<th>Help</th>
<th>Retry</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Help system design

- Help? means ‘help I want information”
- Help! means “HELP. I'm in trouble”
- Both of these requirements have to be taken into account in help system design
- Different facilities in the help system may be required

Help information

- Should not simply be an on-line manual
- Screens or windows don't map well onto paper pages.
- The dynamic characteristics of the display can improve information presentation.
- People are not so good at reading screen as they are text.

Help system use

- Multiple entry points should be provided so that the user can get into the help system from different places.
- Some indication of where the user is positioned in the help system is valuable.
- Facilities should be provided to allow the user to navigate and traverse the help system.
Entry points to a help system

Help system windows

User documentation

- As well as on-line information, paper documentation should be supplied with a system
- Documentation should be designed for a range of users from inexperienced to experienced
- As well as manuals, other easy-to-use documentation such as a quick reference card may be provided
User document types

- Functional description
  - Brief description of what the system can do
- Introductory manual
  - Presents an informal introduction to the system
- System reference manual
  - Describes all system facilities in detail
- System installation manual
  - Describes how to install the system
- System administrator’s manual
  - Describes how to manage the system when it is in use

Document types

- Functional description
- Introductory manual
- System reference manual
- System installation manual
- System administrator’s manual

User interface evaluation

- Some evaluation of a user interface design should be carried out to assess its suitability
- Full scale evaluation is very expensive and impractical for most systems
- Ideally, an interface should be evaluated against a usability specification. However, it is rare for such specifications to be produced
Usability attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learnability</td>
<td>How long does it take a new user to become productive with the system?</td>
</tr>
<tr>
<td>Speed of operation</td>
<td>How well does the system response match the user’s work practice?</td>
</tr>
<tr>
<td>Robustness</td>
<td>How tolerant is the system of user error?</td>
</tr>
<tr>
<td>Recoverability</td>
<td>How good is the system at recovering from user errors?</td>
</tr>
<tr>
<td>Adaptability</td>
<td>How closely is the system tied to a single model of work?</td>
</tr>
</tbody>
</table>

Simple evaluation techniques

- Questionnaires for user feedback
- Video recording of system use and subsequent tape evaluation.
- Instrumentation of code to collect information about facility use and user errors.
- The provision of a grip button for on-line user feedback.

Key points

- Interface design should be user-centered. An interface should be logical and consistent and help users recover from errors.
- Menu systems are good for casual or occasional system users.
- Graphical displays should be used to present trends and approximate values. Digital displays when precision is required.
- Color should be used sparingly and consistently.
Key points

- Systems should provide on-line help. This should include “help, I’m in trouble” and “help, I want information”
- Error messages should be positive rather than negative.
- A range of different types of user documents should be provided.
- Ideally, a user interface should be evaluated against a usability specification.