Interface Design Issues
Lecture 8

Overview

- Quality, need to reduce
  - Response time
- Interaction styles
  - Direct manipulation
  - Interaction devices
- Function versus look
  - Field layout – gizmos/ widgets
  - Fill-in forms
  - Menu design

Response time

- What is acceptable? What is tolerated?
- User expectations
- 2 second rule (Nielsen, 1999) – local systems versus network systems
- Anything longer than 15 seconds is generally detrimental
### Good design results in...

Users achieve rapid task performance, low error rates, and high satisfaction if the following criteria are met:
- Uses have adequate knowledge of the objects and actions necessary for the problem-solving task
- Solution can be carried out without delays
- Distractions are eliminated
- User anxiety is low
- Feedback about progress is given
- Errors can be avoided or handled easily

<table>
<thead>
<tr>
<th>Good design results in...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Users achieve rapid task performance, low error rates, and high satisfaction if the following criteria are met:</td>
</tr>
<tr>
<td>• Uses have adequate knowledge of the objects and actions necessary for the problem-solving task</td>
</tr>
<tr>
<td>• Solution can be carried out without delays</td>
</tr>
<tr>
<td>• Distractions are eliminated</td>
</tr>
<tr>
<td>• User anxiety is low</td>
</tr>
<tr>
<td>• Feedback about progress is given</td>
</tr>
<tr>
<td>• Errors can be avoided or handled easily</td>
</tr>
</tbody>
</table>

### WYSIWYG - We now take for granted

• 'What you see is what you get', principles:
  - Users see a full page of text
  - Document printed as it appears on the screen
  - Cursor action is visible
  - Cursor motion is natural
  - Labelled icons make frequent actions rapid
  - Immediate display of results of action
  - Rapid response and display
  - Easily reversible actions

<table>
<thead>
<tr>
<th>WYSIWYG - We now take for granted</th>
</tr>
</thead>
<tbody>
<tr>
<td>• ‘What you see is what you get’, principles:</td>
</tr>
<tr>
<td>- Users see a full page of text</td>
</tr>
<tr>
<td>- Document printed as it appears on the screen</td>
</tr>
<tr>
<td>- Cursor action is visible</td>
</tr>
<tr>
<td>- Cursor motion is natural</td>
</tr>
<tr>
<td>- Labelled icons make frequent actions rapid</td>
</tr>
<tr>
<td>- Immediate display of results of action</td>
</tr>
<tr>
<td>- Rapid response and display</td>
</tr>
<tr>
<td>- Easily reversible actions</td>
</tr>
</tbody>
</table>

### Direct manipulation

• What is it?
• Insight and understanding user’s model of reality important in designing direct manipulation functionality.
• Used in a wide range of devices/systems. PDA, electronic diaries, on line banking.
• Used in industrial robotics, for example robotic arms the building cars.
• Virtual reality, flight simulators (S&P ch 6 provide good examples)

<table>
<thead>
<tr>
<th>Direct manipulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>• What is it?</td>
</tr>
<tr>
<td>• Insight and understanding user’s model of reality important in designing direct manipulation functionality.</td>
</tr>
<tr>
<td>• Used in a wide range of devices/systems. PDA, electronic diaries, on line banking.</td>
</tr>
<tr>
<td>• Used in industrial robotics, for example robotic arms the building cars.</td>
</tr>
<tr>
<td>• Virtual reality, flight simulators (S&amp;P ch 6 provide good examples)</td>
</tr>
</tbody>
</table>
Issues with direct manipulation

- Consumes valuable screen space
- Users must learn/understand visual representations such as icons
- Can be misleading, users draw incorrect conclusions about what is possible.
- Need to make sure that visual feedback is rapid particularly where there is rapid keyboard interactions.
- Selecting right objects and actions for direct manipulation may be difficult. (S&P ch 6)

Control manipulation – THE MOUSE

- Cooper and Reimann (2003) describe two mouse actions: clicking and clicking and dragging.
- Most important (and frequently ignored) part of a drag operation is how the user gets out of it.
- Think of the different mechanisms used to alert us as to the success or otherwise of a click and drag operation.
- What are some of the other design considerations for click and drag operations?

Other operations

- Charged cursor tools
- Object manipulation – repositioning, resizing and reshaping.
- How do you know that an object can be repositioned, resized reshaped?
- How does the user know what to ‘grab’ to activate this command?
Input device considerations

- Kiosks – must be clear, easy to master
- Small screen displays have limited display 'real estate' serious consideration for designers.
- How is device carried or used? Does it require two hands?
- Effectiveness for the task
- Time to learn
- Cost and reliability
- Size and weight

(good discussion in chapter 9 S&P)

Field layout: Gizmos and widgets (interface components)

- Assist in data entry
- Reduce human data entry (therefore reduce error)
- Need to select the most appropriate gizmo/widget for the user, the application and the screen design

Imperative gizmos

- Based on verbs
- Immediate action
- User expects to see change in state of button
- Addition of ToolTips
Selection gizmos

- latch button
- flip flop (toggle)
- listbox
- combo box
- tree gizmo
- radio buttons
- check boxes

Entry gizmos

- Enable the user to enter new information into the program rather than selecting information from an existing list
- Bounded entry restricts entry to a set of values

Form fillin

- Some tasks too cumbersome with menus.
- Where data entry of text or numbers is required keyboard typing becomes more attractive.
- If many fields of data necessary appropriate interaction style is 'form fillin'.
- Very frequent style used in Web based systems.
## Fill-in forms

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simplifies or reduces data entry</td>
<td>Assumes typing skills (and ability)</td>
</tr>
<tr>
<td>Self explanatory</td>
<td>Error prone due to incorrect data entry</td>
</tr>
<tr>
<td>Requires modest training</td>
<td>Inflexible and restrictive</td>
</tr>
<tr>
<td>Little cognitive load</td>
<td>Requires understanding users and their workflow</td>
</tr>
<tr>
<td>Gives convenient assistance</td>
<td></td>
</tr>
<tr>
<td>Efficient use of screen space</td>
<td></td>
</tr>
<tr>
<td>Permits use of form management tools</td>
<td></td>
</tr>
</tbody>
</table>

## Fill-in form guidelines

- Should be some indication of data type to be entered into field or message indicating if for example characters entered instead of numbers.
- Corrections should be obvious.
- Often an electronic form is designed to look like paper version
- Ensure the required numeric format is clear (currency, data, telephone)

## Fill-in form guidelines

- Meaningful title:
- Comprehensible instructions:
- Logical grouping and sequencing of fields
- Visually appealing layout of the form
- Use familiar field labels, consistent terminology and abbreviations
- Visible space and boundaries for data entry fields
- Use colour to group items
- Convenient cursor and Tab movement
Fill in form guidelines

- Cursor should move from one data item to next.
- Should be some indication of data type to be entered into field or message indicating if for example characters entered instead of numbers.
- Corrections should be obvious.
- Often designed so forms look like paper version.
- Error correction and prevention messages should be simple
- Optional fields clearly marked
- Completion signal

Other interface design considerations -- Colour

Colour can improve task performance and increase interface attractiveness. Colour can:
- Soothe or strike the eye
- Add accents to an interesting display
- Facilitate subtle discriminations in complex displays
- Emphasise logical organisation of information
- Draw attention to warnings
- Evoke strong emotional reactions of joy, excitement, fear, or anger. (Shneiderman 398)

Shneiderman recommends (398-400):

- Use colour conservatively
- Limited number of colours
- Recognise power of colour as a coding technique
- Recognise the power of colour to speed or slow tasks
- Ensure colour coding supports task
- Consider needs of colour deficient users
• Use colour to helping formatting
• Be alert to problems with colour pairings
• Use colour changes to indicate status changes
• Use colour in graphic displays for greater information density
• Consider why word processors are black on white.

Metaphors

• Metaphors describe overall concept within which objects and actions are organised (then converted to icons).
• Help users think about screen objects as they would about real-world objects.
• When designing the visual representation of current process need to stay close to real world objects.
• Metaphors are not always good representations of real-world objects, in designing this has to be considered.

Icons

• Used to reduce complexity of systems
• Make systems easier to learn
• Difficult to design because designers must always ensure the underlying concept does convey a distinct meaning.
Designing menus

- Must remember that even with short response times frequent menu users can become annoyed if they are required to make several menu selections to complete simple tasks.
- Three possible approaches can be used to improve this: to allow typeahead, assign names to menus, create menu macros.

(S&P ch 7)

Menu types

- Single menu: two or more items, maybe permanently available or pop up.
- Binary menus: yes/no, true/false e.g. printer icon.
- Multiple menu items
- Pull down and pop up menus
Menus with Typeahead: BLT approach

- This allows users to not have to wait to see menus before choosing items.
- Users can type a string of letters or numbers when presented with main menus. Many applications allow for this.
- Advantaged users can move from novice menu users to knowledgeable command users easily. No new commands to learn.

Menu names or bookmarks

- Numbered menu items assigned, menu names to each menu frame.
- Users can follow menus or if no name of function can type it in and go there directly.
- Saves users having to work through several levels of menus.
- Strategy useful if there is only a small number of destinations each user needs to remember.

Menu macros, custom toolbars and style sheets

- Allows users to record as macros regularly used paths, place them on toolbars or create as an icon.
- Mechanism allows tailoring of system and can provide simplified access mechanisms for users with special needs.
Menu layout

- Titles: need to be carefully chosen, particularly descriptive or memorable, title can make a big difference in responses.
- For single menus simple descriptive title that identifies situation is all that is necessary.
- With linear sequence of menus title should accurately represent stages in sequence.
- Consistent placement of titles and other features, menu screens important, inconsistent placement increases users’ time to select items.

Menus and navigation

- Designers must display menus so that they are natural to use.
- Useful to organise commands in a hierarchical way. Problem is deciding which items to include at different levels and which items to group.

Alternatives

- Empirical evidence suggests 4 alternatives:
  - Alphabetical
  - Categorical – relies on selection of suitable Categories can be difficult
  - Conventional – neither categorical nor alphabetical
  - Frequency – based on use (Preece 265-66)
Phrasing of menu items

- Guidelines:
  - Use familiar and consistent terminology – must be familiar to user community
  - Ensure individual items are distinct from one another
  - Use consistent and concise phrasing
  - Bring keyword to the left - first word user likely to recognise should be on the left.

(S&P ch 7)

---

Graphic layout and design

- “Consistent formats help users to locate necessary information, focus users attention on relevant material, and reduce users’ anxiety by offering predictability.”
  (Shneiderman 260)
- Guidelines:
  - Titles centred or left justified?
  - Item placement – typically left justified with item number or letter proceeding item description.
  - Instructions – should be identical in each menu
  - Error messages – should be consistent

---

Schneiderman: 8 golden rules of interface design (74)

- Strive for consistency
- Enable frequent users to use shortcuts
- Offer informative feedback
- Design dialogues to yield closure
- Offer error prevention and simple error handling
- Permit easy reversal of actions
- Support internal locus of control
- Reduce short-term memory load
Guidelines

• Don’t create cluttered displays overusing techniques
• Novice users require simple logical well-organised, well labelled displays.
• Expect users need extensive field label more subtle highlighting etc

Functionality

• A major task for developers is to determine functionality of a system.
• Usually done by studying users’ task domain.
• Outcome is list of task actions and objects-abstracted into set of interface actions and objects, in turn these represented with low level interface syntax.

Creeping featurism

• Common problem is to provide excessive number of objects / actions which can overwhelm users.
• Excessive objects / actions take more code to maintain, cause more bugs, slowdown execution times, require more help screens, error messages etc.
• For users excess functionality slows learning.
Function versus look

- Good screen design is the key to successful interface design
- Start with:
  - Knowledge of users' tasks
  - Meaningful grouping of items
  - Consistent sequencing of groups
  - Justification (text, images, numbers, dots)

Readings

- Schneiderman & Plaisant Chapter 11, 12
  http://melody.syr.edu/hci/jais04/JAIS_Galletta.pdf
- Usability by Design