1

User Interface Design

Designing effective interfaces for software systems

Importance of user interface

2

- System users often judge a system by its interface rather than its functionality
- 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

Graphical user interfaces

3

- Most users of business systems interact with these systems through graphical user interfaces (GUIs)
  - although, in some cases, legacy text-based interfaces are still used

GUI characteristics

4

- Windows
- Icons
- Menus
- Pointing Devices
- Graphics

GUI advantages

5

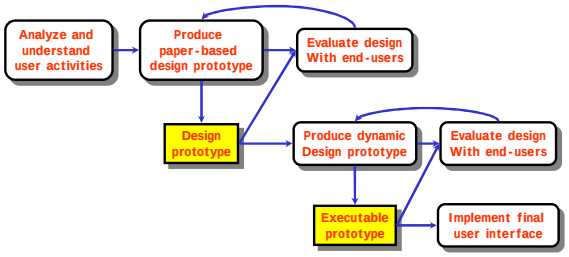
- They are easy to learn and use
  - Users without experience can learn to use the system quickly
- The user may switch quickly from one task to another and can interact with several different applications
  - Information remains visible in its own window when attention is switched
- Fast, full-screen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

User-centred design

6

- User-centred design is an approach to UI design where the needs of the user are paramount and where the user is involved in the design process
- UI design ***always*** involves the development of prototype interfaces

User interface design process



UI design principles

- UI design must take account of the needs, experience and capabilities of the system users
- Designers should be aware of people's physical and mental limitations (e.g. limited short-term memory) and should recognize that people make mistakes
- UI design principles underlie interface designs although not all principles are applicable to all designs

User interface design principles

Principle	*Description*
User familiarity	The interface should use terms and concepts which are drawn from the experience of the people who will make most use of the system.
Consistency	The interface should be consistent in that, wherever possible, comparable operations should be activated in the same way.
Minimal surprise	Users should never be surprised by the behaviour of a system.
Recoverability	The interface should include mechanisms to allow users to recover from errors.
User guidance	The interface should provide meaningful feedback when errors occur and provide context-sensitive user help facilities.
User diversity	The interface should provide appropriate interaction facilities for different types of system user.

Design principles

- User familiarity
  - 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.
- Consistency
  - The system should display an appropriate level of consistency. Commands and menus should have the same format, command punctuation should be similar, etc.
- Minimal surprise
  - If a command operates in a known way, the user should be able to predict the operation of comparable commands

Design principles

- Recoverability
  - 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.
- User guidance
  - Some user guidance such as help systems, on-line manuals, etc. should be supplied
- User diversity
  - Interaction facilities for different types of user should be supported. For example, some users have seeing difficulties and so larger text should be available

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

Interaction styles

13

- Direct manipulation
- Menu-based (Menu selection)
- Form fill-in
- Command language
- Natural language

Advantages and disadvantages

Interaction style	Main advantages	Main disadvantages	Application examples
Direct manipulation	•Fast and intuitive interaction •Easy to learn	•May be hard to implement •Only suitable where there is a visual metaphor for tasks and objects	•Video games •CAD systems
Menu selection	•Avoids user error •Little typing required	•Slow for experienced users •Can become complex if many menu options	•Most general-purpose systems
Form fill-in	•Simple data entry •Easy to learn	•Takes up a lot of screen space	•Stock control, Personal loan processing
Command language	•Powerful and flexible	•Hard to learn •Poor error management	•Operating systems, Library information retrieval systems
Natural language	•Accessible to casual users •Easily extended	•Requires more typing •Natural language understanding systems are unreliable	•Timetable systems •WWW information retrieval systems

Direct manipulation advantages

15

- 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

16

- 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

Menu systems

17

- 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

18

- 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

19

- Actions that 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

Command interfaces

20

- 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

Command interfaces problems

21

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

Command languages

22

- 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

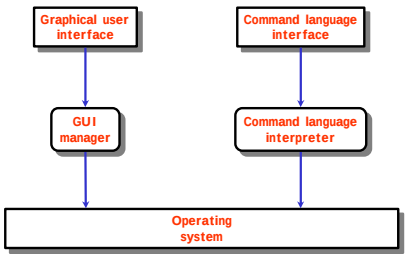
Natural language interfaces

23

- The user types a command in a natural language. Generally, the vocabulary is limited and these systems are confined to specific application domains (e.g. timetable enquiries)
- NL processing technology is now good enough to make these interfaces effective for casual users but experienced users find that they require too much typing

Multiple user interfaces

24



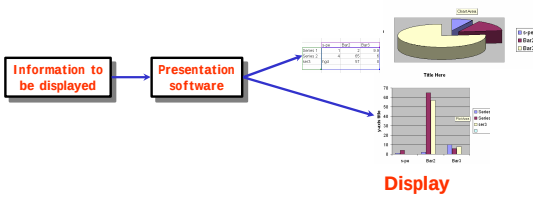
Information presentation

25

- Information presentation is concerned with presenting system information to system users
- The information may be presented directly (e.g. text in a word processor) or may be transformed in some way for presentation (e.g. in some graphical form)
- The Model-View-Controller approach is a way of supporting multiple presentations of data

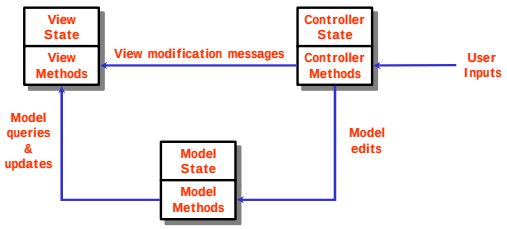
Information presentation

26



Model-view-controller

27



Information presentation

28

- Static information
  - Initialized at the beginning of a session. It does not change during the session
- Dynamic information
  - Changes during a session and the changes must be communicated to the system user
- Both may be either numeric or textual

Information display factors

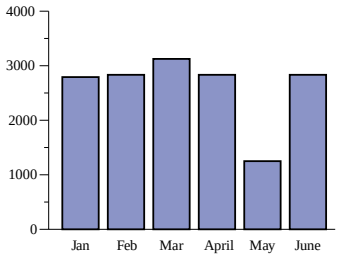
29

- 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?

Alternative information presentations

30

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



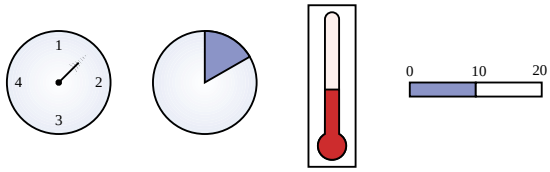
Analog vs. digital presentation

31

- Digital presentation
  - Compact - takes up little screen space
  - Precise values can be communicated
- Analog 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

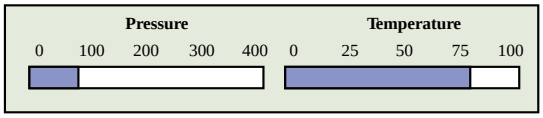
32



Dial with needle | Pie chart | Thermometer | Horizontal bar

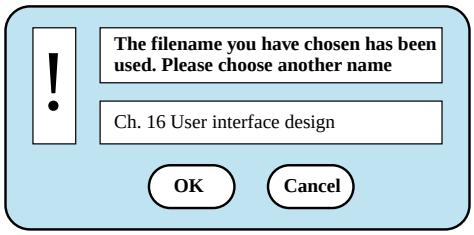
Displaying relative values

33



Text highlighting

34



User documentation

35

- 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

Document types

36

- 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

User interface evaluation

37

- 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

38

Attribute	Description
Learnability	How long does it take a new user to become productive with the system?
Speed of Operation	How well does the system response match the user's work practice?
Robustness	How tolerant is the system of user error?
Recoverability	How good is the system at recovering from user error?
Adaptability	How closely is the system tied to a single model of work?

Simple evaluation techniques

39

- 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 for on-line user feedback