

502045

Software Engineering

Chapter 09

Lesson 13: Graphic User Interface

User interface design

- User interface design principles
- User interaction
- Information presentation
- User support
- Interface evaluation

Information presentation

- 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

The user interface

- 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**

GUI characteristics

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

User-centred design

- The aim of this chapter is to sensitise software engineers to key issues underlying the design rather than the implementation of user interfaces
- **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

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.

Design principles

- **Consistency**
 - 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

- 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.

Design principles

- User guidance
 - Some user guidance such as help systems, on-line manuals, etc. should be supplied

Design principles

- User diversity (variety)
 - Interaction facilities for different types of user should be supported. For example, some users have seeing difficulties and so larger text should be available
 - Provide guidance, but power users require shortcuts

User interface design

- User interface design principles
- User interaction
- Information presentation
- User support
- Interface evaluation

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?

Interaction styles

- Direct manipulation
- Menu selection
- Form fill-in
- Command language
- Natural language

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
- Direct manipulation interfaces can be **complex to program and make heavy demands on the computer system**

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

- 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

Form-based interface

NEW BOOK	
Title	<input type="text"/>
Author	<input type="text"/>
Publisher	<input type="text"/>
Edition	<input type="text"/>
Classification	<input type="text"/>
Date of purchase	<input type="text"/>
ISBN	<input type="text"/>
Price	<input type="text"/>
Publication date	<input type="text"/>
Number of copies	<input type="text"/>
Loan status	<input type="text"/>
Order status	<input type="text"/>

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

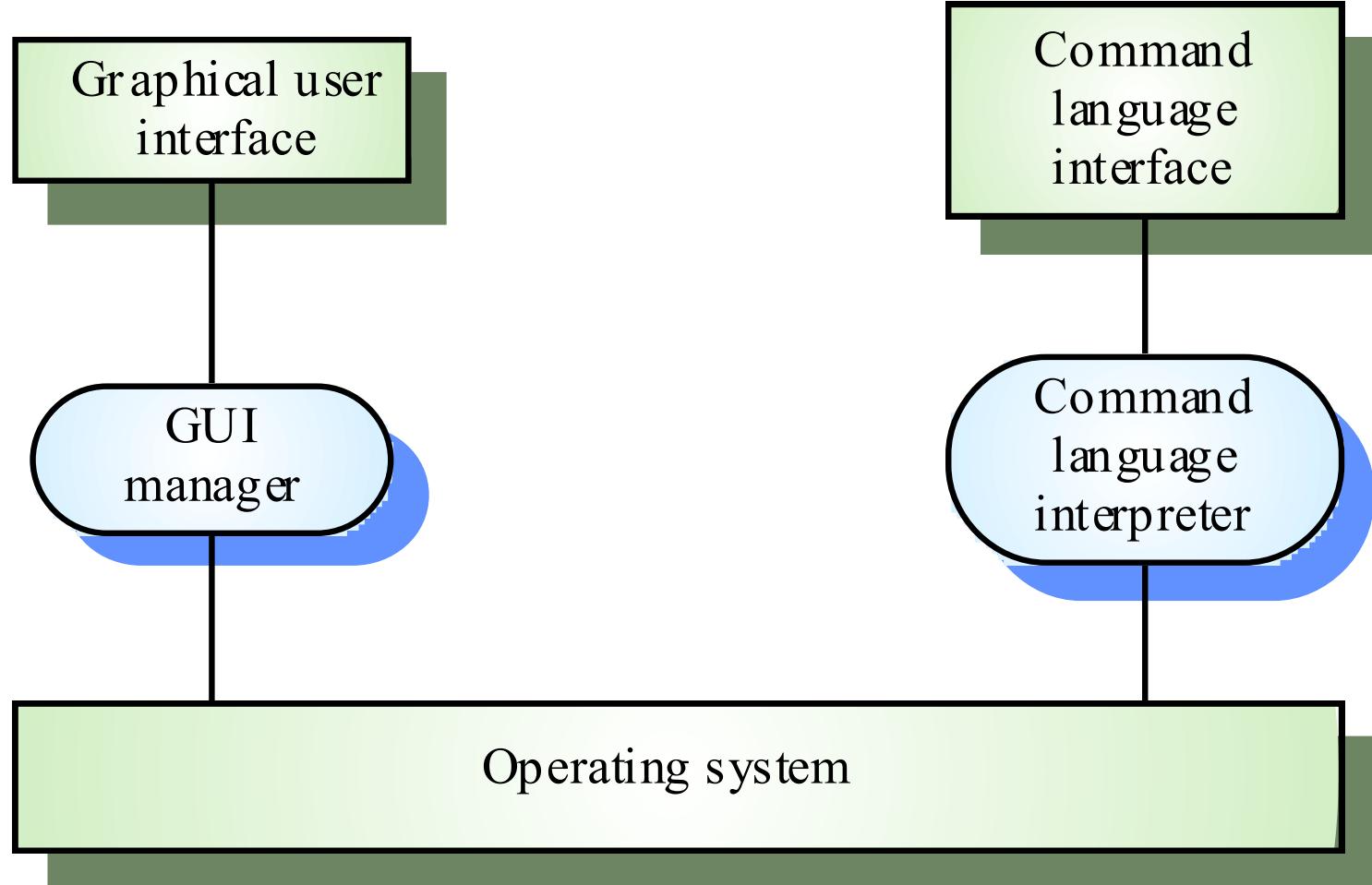
Problems with command interfaces

- 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

Natural language interfaces

- 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
- Search engine

Multiple user interfaces



User interface design

- User interface design principles
- User interaction
- Information presentation
- User support
- Interface evaluation

Information presentation

- Text
- Graphic
- Color

Information presentation

- 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

- **Static information**
 - Initialised 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

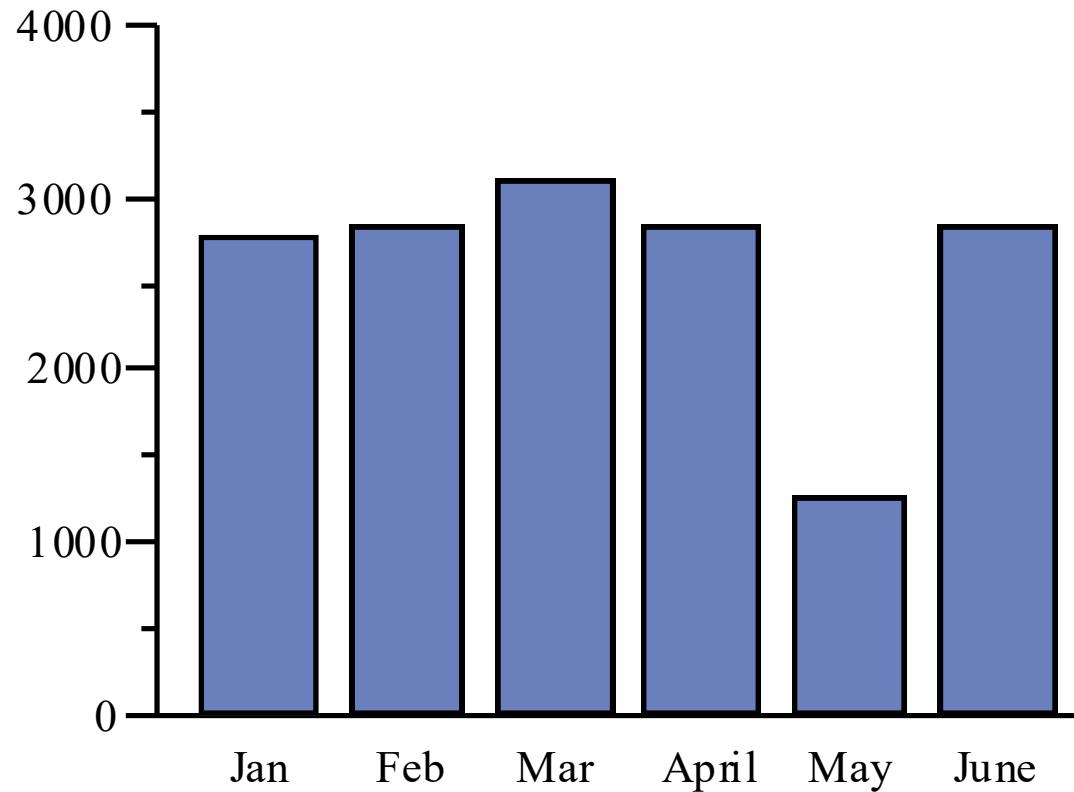
- Is the user interested in precise information or data relationships?
- How quickly do information values change? Must the change be indicated immediately?
- Must the user take some action in response to a change?
- Is there a direct manipulation interface?
- Is the information textual or numeric? Are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

Textual vs graphics

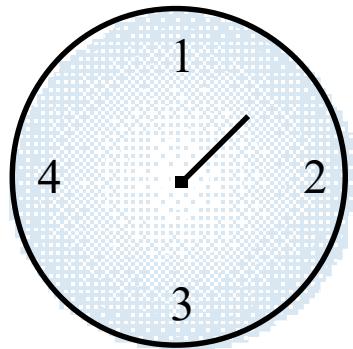
- can take a long time to download if the user is working over a slow, dial-up connection (**not graphic**)
- precise information (**text**) vs relationships between data (**graphic**)
- changes relatively slowly (**text**) vs changes quickly (**graphic**)

Alternative information presentations

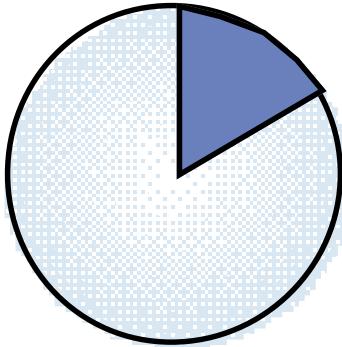
Month	Value
Jan	2842
Feb	2851
Mar	3164
April	2789
May	1273
June	2835



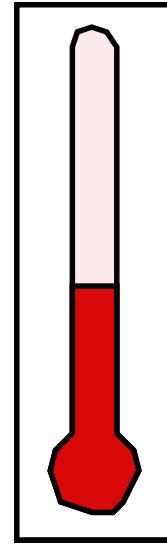
Dynamic information display



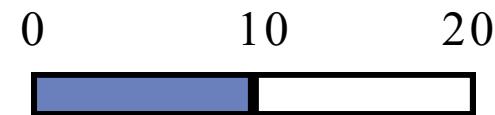
Dial with needle



Pie chart

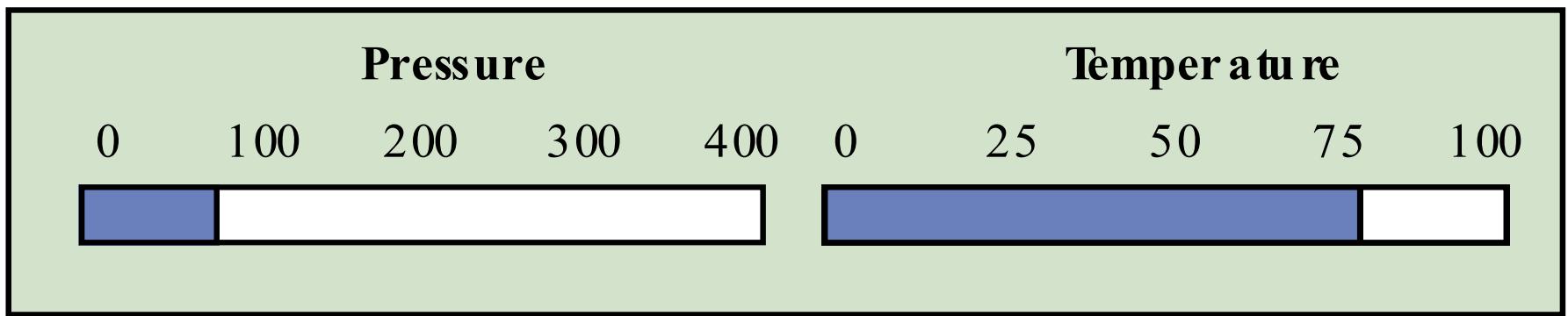


Thermometer



Horizontal bar

Displaying relative values



Colour displays

- Colour 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 colour in interface design include:
 - The use of colour to communicate meaning
 - Over-use of colour in the display

Colour use guidelines

- Don't use too many colours
- Use colour change to show a change in system status
- Allow users to control colour coding
- Design for monochrome then add colour

Colour use guidelines

- Use colour coding consistently
- Avoid colour pairings which clash
- Be aware that colour displays are usually lower resolution
- you should use colour for highlighting, but you should not associate meanings with particular colours

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