More about Graphical User Interfaces

*CIS*2750*

Advanced Computing Techniques

Disadvantages of GUIs

- **Complexity increases**.
  - The design task is now far more complex -- there are many alternatives for techniques, colours, style
  - The designer needs many skills - graphics designer, psychologist, ergonomics expert, etc.
  - The chances of clutter and confusion are increased.

Disadvantages of GUIs

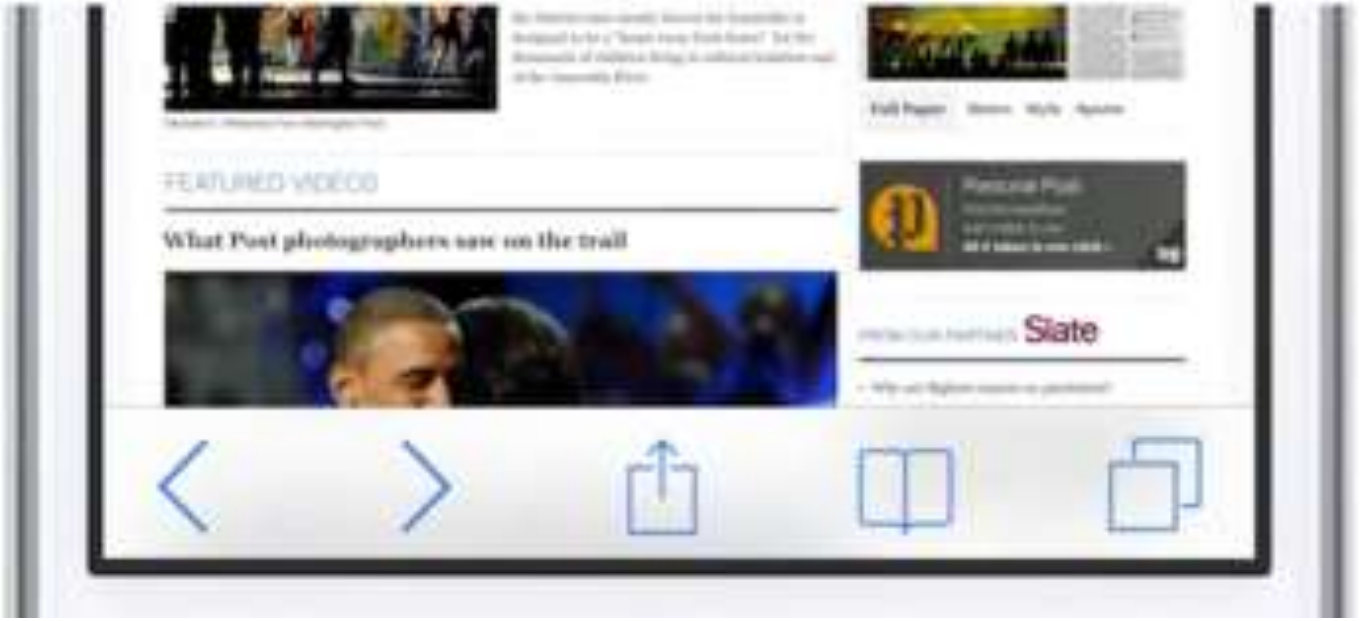
- **Learning time is not insignificant**.
- **Lack of experimentally derived design guidelines**.
  - Because of the complexity of today's GUIs it is difficult to study the relationship of GUI elements to productivity and usability
- **Inconsistencies in techniques and terminology**.
  - This is a consequence of both the difficulties and time involved in developing standards and the commercial and *legal aspects* of user interface design.

On next page…

Disadvantages of GUIs

- **Symbols and graphics are not always as expressive as language can be for searching and describing actions and things**.
  - Symbols are not as familiar as numbers and words.
  - The human ability to differentiate between different symbols is more limited than with text.
  - The human ability to effectively deal with a large number of icons is limited.

Q2. #Which iOS 7 Safari icon represents “share”?



A B C D E

Disadvantages of GUIs

- The development of new icons is expensive and largely an unknown domain.
  - In contrast, typeface design has existed for 300 years.
- Some studies have found that people prefer a display that incorporates textual captions.

Disadvantages of GUIs

- **GUIs can be inefficient**.
  - Window manipulations are excessive and repetitive in some GUIs.
  - Pointing and clicking is slower than typing for an experienced typist.
  - Expert users often can make very effective use of command languages and are slowed down by GUIs.

Disadvantages of GUIs

– Graphics based formats may consume more screen space.
  - Example: compare a textual list of names and telephone numbers with a card file representation of the same.

Characteristics of GUIs

- **Sophisticated Visual Presentation**
  - Interface elements
    - windows (primary, secondary, dialog boxes)
    - menus (menu bar, pull-down, pop-up)
    - icons (representing files, programs)
    - screen-based controls (list boxes, scroll bars, buttons)
    - mouse pointer and cursor
  - The objective is to visually represent the real world of the user in a meaningful, simple and clear way.

Characteristics of GUIs

- **Pick and Click Interaction**
  - The *identification* of an action to be performed is called a pick and the signal to *perform* the action is a click.
  - Primary mechanism: mouse and mouse buttons.
  - Secondary mechanism: keyboard.
- **Restricted Set of Interface Options**
  - All currently available alternatives are the only ones presented to the user (WYSIWYG).

Characteristics of GUIs

- **Visualization**
  - Visualization is the process of representing graphical information that is hard to understand because of its volume or abstract nature.
  - The graphic image must convey the relevant information about the data.
  - **"Periodic Table" of visualization methods:** **http://www.visual-literacy.org/periodic_table/periodic_table.html#**
- **Object Orientation**
  - Objects are what users see on the screen -- they can be manipulated.

Characteristics of GUIs (objects)

- SAA (System Application Architecture) from IBM defines three types of objects :
  - **Data objects** represent information. They can be used for information collection or presentation.
  - **Container objects** hold other objects. They provide a grouping for ease of access and retrieval.
    - *Workplace:* storage area for all objects.
    - *Folders:* general purpose containers for long term storage of objects.
    - *Work areas:* temporary storage folders for objects currently being worked on.
  - **Device objects** represent physical objects in the real world.

Characteristics of GUIs (objects)

- *Attributes* of objects define the unique characteristics of an object.
- Objects can be acted upon or manipulated (commands) or have their attributes modified (attribute specification).
  - *Command examples:* opening a document, printing a file.
  - *Attribute specification:* selecting a font or colour.
- A series of actions can be performed on a specific object.

Characteristics of GUIs

- A **view** is a way of looking at an object.
  - **Composed** views
    - present information and objects contained within an object.
  - **Contents** views
    - list the components of objects.
  - **Settings** views
    - permit the manipulation of object attributes.
  - **Help** views
    - provide help functions.

Characteristics of GUIs

- **Use of Recognition Memory**
  - Continuously viewed objects do not have to be *remembered*.
- **Concurrent Performance of Functions**

Principles of Graphical User Interface Design

- **Aesthetically Pleasing**
  - Provide visual appeal by
    - meaningful contrast between screen elements
    - groupings
    - alignment of screen elements and groups
    - 3-D representation
    - effective and simple use of colour

Clarity

- Provide visual, conceptual and linguistic clarity for
  - visual elements
  - functions
  - metaphors
  - words and text

Compatibility

- Provide compatibility with
  - the user (adopt the user's perspective)
  - the task
  - the product

Comprehensibility

- The system should be easy to learn and understand. The sequence of actions to be taken should be in an order that is sensible and easy to remember.
- The user should realize:
  - what to look at
  - what to do
  - when to do it
  - why to do it
  - how to do it

Configurability

- The system should permit easy configuration and modification of settings.
  - allow for personal preferences
  - enhance an understanding of the system

Consistency

- The system should look and perform the *same way* at all times. Similar components should
  - have a similar look
  - have similar uses
  - operate similarly
  - the same action should always have the same result
  - the function of components should not change
  - the position of elements should not change

Control

The *user* must control the system.

- actions are the consequences of explicit user requests
- actions should be performed quickly
- actions should be interruptible
- the user should not be punished for errors

- The user's skills should be considered.
- Avoid over-constraining the user (i.e. many restrictive modes).
- Allow for customization, but
- Provide a good set of defaults.

Directness

- Tasks should be accomplished by direct and intuitive means.

Efficiency

- Eye, hand and other control movements should be minimized.
- Transitions between input/control modes should be infrequent, easy to accomplish and not distracting.

Familiarity

- Exploit things that the user is familiar with, such as concepts and languages.
- A natural interface mimics the user's behavioural patterns.
- Make use of real world metaphors.

Flexibility

- System behaviour should be based upon the knowledge and skill of the user. User characteristics that should be considered include
  - knowledge and skills
  - experience
  - personal preferences
  - habits

Forgiveness

- Tolerate and forgive usual human errors.
- Prevent errors whenever possible.
- Protect against catastrophic errors.
- Provide constructive error messages.

Predictability

- The user should be able to anticipate the behaviour of the system. This is aided by
  - distinct and recognizable screen elements
  - cues to the results of any action to be performed

Recovery

- The system should permit
  - commands to be terminated or reversed
  - return to a previous state if difficulties arise

Responsiveness

- The system must respond rapidly to requests.
- The system must acknowledge all user actions (visual, textual, auditory).

Simplicity

- KISS : Keep It Simple, Stupid !! Always prefer the simple over the complex.
  - Hide things until they are necessary.
    - Present common and necessary functions first.
    - Feature important functions.
    - Hide sophisticated and rarely used functions
  - Provide defaults.
  - Make common actions simple (*uncommon* actions can be more complex)
  - Provide uniformity and consistency.

Transparency

- Permit the user to concentrate on the task at hand and not be distracted by the mechanics of the interface.

Trade-Offs

- Design is always a balancing act between conflicting principles.
- The user's needs should always take precedence over the system's needs.

The Design Process (Overview)

- Understand the user.
- Involve the user.
- Perform rapid GUI prototyping and testing.
- Modify and iterate the design as many times as needed.
- Integrate all the system elements which have been developed concurrently.

The Steps of GUI Design (1)

- Know the user/client.
- Understand the task.
- Study and understand the principles of good screen design.
- Select the appropriate kinds of windows.
- Develop system menus.

The Steps of GUI Design (2)

- Select the appropriate control devices.
- Choose the proper screen-based controls.
- Organize and lay out the windows.
- Select the proper colours.
- Create/select meaningful icons.
- Provide meaningful messages/feedback.
- Test the system.