HCI -7- INTERACTION STYLES 2

Aims and Objectives: the previous lecture introduced the topic of interaction styles and described a number of common styles including command driven interaction, menus, question and answer dialogues, forms and the use of natural language. This lecture completes the discussion by describing a particularly important interaction style called direct manipulation.

Lecture Notes

Introduction to Direct Manipulation Interfaces

The term '' manipulation interface'' was coined in 1981 by the CI researcher and author, Ben Shneiderman. Direct manipulation interfaces are characterised by the following features:

- The visibility of the ***objects of interest***. The objects of interest will depend on the system. In a chess example (see below) they will include the chessboard and pieces. In a file system they will be the files and directories. In a network configuration system they may be network nodes and routers.

- Ability to manipulate the objects of interest in a direct manner usually using some pointing device (e.g. mouse). Not only should the user be able to see the objects of interest but they should also be able to manipulate them in some direct way.

- Incremental actions with rapid feedback. When the user manipulates some objects (e.g. moving a file between folders by dragging it) they should get constant feedback about the operation i.e. they will see the movement of the file during the dragging action and can change direction if they wish.

- Reversibility of actions. It should be easy for the user to change their mind and reverse their previous actions. This encourages the user to explore the system.

- All possible actions should be legal. Using a command driven interface it is easy for the user to make syntactic errors (e.g. by typing a command name wrongly). A direct manipulation interface should prevent the user from making this type of mistake.

Direct Manipulation Interfaces are usually highly graphical so Graphical User Interface (GUI) is sometimes used as an equivalent term. You will however see below that just because an interface is graphical it doesn't necessarily mean it is a direct manipulation interface.

There is a relationship between direct manipulation and WIMP interfaces but the terms aren't interchangeable. WIMP stands for Windows, Icons, Mice, Pull-down Menus or Windows, and Pointers. The first commercially successful WIMP interface was for the

Apple Macintosh in 1984. WIMP interfaces have since become the most common type of interface for PCs (Microsoft Windows) and are also widely used with UNIX (using X windows) some aspects of WIMP systems are direct manipulation (e.g. dragging files to move them) but they also include other interaction styles such as menus (obviously) and question and answer dialogue.

The Use of Metaphors in Direct Manipulation Interfaces

Choosing how to make the ''objects of interest'' visible is very easy in some cases (e.g. the chess example below) but may be less obvious in others. Often a ***Metaphor*** has to be adopted. Some example metaphors are described below.

The ***desktop metaphor*** (e.g. folders, wastebasket/recycle bin, clipboard etc) is now so familiar to most of us that we forget that it is a metaphor at all. It seems very natural to think about storing related documents in folders with labels that indicate their contents. The desktop metaphor was first introduced with the Xerox Alto and Xerox Star in the early 1980s.

The ***toolbox Metaphor*** is widely used. For instance when using the graphics package PaintShop Pro you have available to you a toolbox containing things such as a paintbrush, pencil, eraser, spray can etc. To use a tool you pick it up from the toolbox and apply it to the image you are working on.

The ***book metaphor*** is often used for online documentation. For example when accessing Windows Help you have access to a hierarchical table of contents and a cross-referenced index just as you would have in a book.

A good metaphor helps the user to:

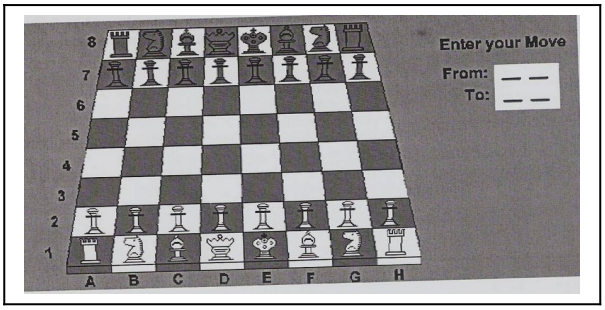
- build an appropriate metal model of the system
- feel comfortable with the system
- make use of ''real world'' knowledge in using the system

Quick Quiz Question 1	
a)	What are the five characteristics of direct manipulation interface?
b)	What does your real-world knowledge about wastebaskets

Do systems with graphical interfaces necessarily have direct manipulation interfaces?

Compare the following three chess games.

Chess Program 1



In this version of the game the user enters their move by typing in the grid reference of the ''from'' and ''to'' squares.

Below we are checking the interface against the criteria for direct manipulation visible.

Are the objects interests visible? Yes, the chessboard and pieces are constantly visible.

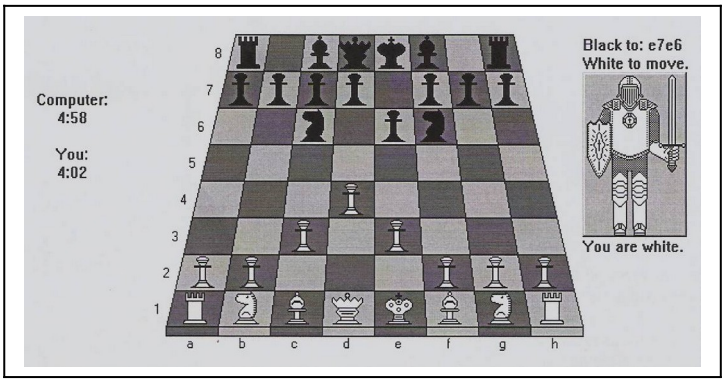
Are the objects of Interest manipulated in the direct manner? No, typing in the to and from references is not really manipulating the objects directly.

Are the actions incremental with rapid feedback? Not really incremental. The move is all or nothing. When the user has typed in the to and from square the piece will be moved immediately.

Are all possible actions legal? Probably not. There is nothing to stop the user typing in an illegal move (e.g. to a square that is already occupied). Presumably if that happens the game will detect the error and give an error message but it won't actually have prevented the user from attempting the wrong move in the first place.

So overall – is this a direct manipulation interface? No.

Chess program 2



In this example the move is made by clicking on '' from'' square and then clicking on ''to'' squares.

Are the objects of interest visible? Yes

Are the objects of interest manipulated in direct manner? Not really, It is more direct way of indicating the move than the previous example but it is still not the pieces themselves that are operated on.

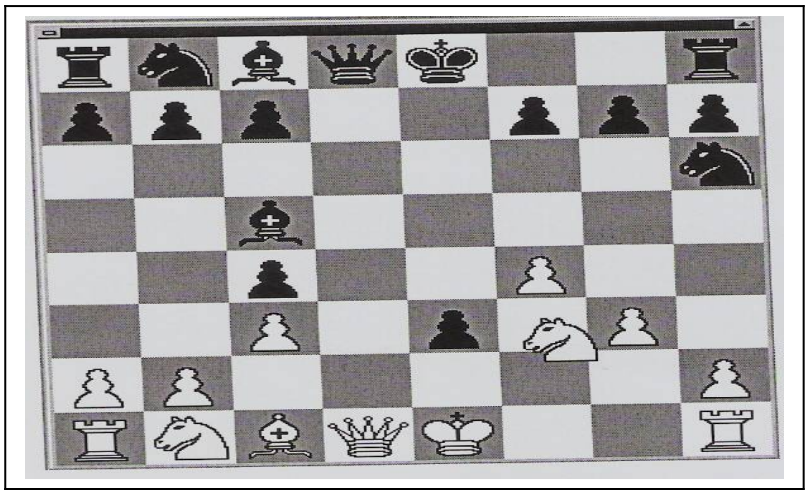
Are the actions incremental with rapid feedback? Again, not really**.** When the user clicks on the 'to' square the move will occur. As in the previous example it is an all or nothing action.

Are the actions reversible? Again, we can't be sure about this.

Are all possible actions legal? Possibly. The game could indicate to the user that a move they are contemplating is illegal by perhaps changing the appearance of the cursor as they move it over the board.

So overall – is this direct manipulation interface? No.

Chess program 3



With this version of the game the movement is by clicking on piece and dragging it to this new position.

Are the Objects of interest visible? Yes.

Are the objects of interest manipulated in direct manner? Yes, they are in effect picked up and moved my the mouse.

Are the actions incremental with rapid feedback? Yes, as the user moves the piece they can see where it is. They may start to move a piece to a certain square and then change their mind and put it down on a different square.

Are the actions reversible? We cant be sure if the user can change their once they dropped the piece onto a square at least they can get a sort of preview of each move by holding the piece over a particular square. They can then change their mind and move the piece elsewhere.

Are all possible actions legal? As in the previous example it is possible that they are.

So overall – is this a direct manipulation interface? Yes

Quick Quiz Question 2

a) What does WIMP stands for?

b) Are the chess games shown above examples of WIMP interfaces?

Advantages and Possible Problems of Direct Manipulation Interfaces

There are a number of advantages claimed for well designed direct manipulation interfaces. Some of the most important points are outlined below.

- Novices can learn basic system operation quickly. This is partly because if a good metaphor is used they can bring a certain amount of prior real-world knowledge to bear.

- System knowledge is retained. Having once learned how to do something then it is easy to remember it. The visual nature of the interface itself acts as a memory prompt.

- Reduction in errors. The system will prevent the user from making certain types of elementary (i.e. syntactic) errors. Of course the user can still make other types of errors (e.g. moving a file to the wrong directory), but often these can be revered easily (e.g. using a ‘’undo’’ operation).

- Users are encouraged to experiment and explore. This is good because it helps people learn. People feel confident enough to do this because of the familiarity of the system (provided by the metaphor), the incremental nature of actions and the ability to reverse unwanted actions.

- People enjoy using such systems and find them satisfying to use.

Some possible problems with direct manipulation interfaces are described below.

- They are much more suited to some tasks than others and will appear clumsy if applied inappropriately. For example, imagine a direct manipulation interface for entering customer orders representing the user's order form.

- Direct manipulation interfaces are ore difficult to program than other types of interface. This is so even when using visual development tools as Visual Basic.

- System requirements (e.g. screen resolution, memory, processing power) are much greater for direct manipulation interfaces. This may not seem much of a problem at first after all memory etc are relatively cheap. However this can be a problem for small mobile devices such as mobile phones.

Quick Quiz Question 3

What interaction styles are commonly used on mobile phones?

Summary:

This lecture has discussed the direct manipulation style of interface. To be categorised as direct manipulation, an interface must have the following characteristics; visibility of the objects of interest, ability to manipulate the objects of interest in a direct manner, incremental actions with rapid feedback, reversibility of actions and legality of all possible actions. Often direct manipulation interfaces rely on the use of a visual metaphor (e.g. the desktop metaphor).
Although direct manipulation interface are highly graphical it does not follow that all graphical interfaces are direct manipulation interface. To demonstrate this we looked at three example chess games. All are graphical but only one really qualifies as being a direct manipulation interface.

A number of advantages are claimed for direct manipulation interface and on the whole these outweigh the possible problems.

Having completed this lecture you should:

- Be able to explain the characteristic of direct manipulation interfaces.
- Be able to explain how direct manipulation interface relate to WIMP and graphical user interfaces.
- Discuss the advantages and problems of direct manipulation interfaces and thus identify whether one is appropriate in a given situation.

Tutorial Exercises

1. Tutorial exercise 1 for the previous lecture presented a table for you to complete Commenting on the various interaction styles that you had studied. Now that you have studied direct manipulation interfaces try commenting on how they measure up against the principles of good interface design.

Style	Naturalness	Consistency	Relevance	Supportiveness	Flexibility
Direct Manipulation					

2. Much software combines more than one interaction style in its interfaces. Choose three packages with which you are familiar and complete the following table identifying which interaction styles they use.

Package Name	Package 1	Package 2	Package 3
Command Driven			
Forms			
Q and A dialogues			
Menus			
Natural Language			
Other			

3. Imagine that you are to design the interface software that enables members of the public to buy tickets for performance at a particular theatre. The system will allow the user to choose the date of the performance they wish to attend and choose their seats. Outline how you could make use of the direct manipulation style in this interface. Would some aspects of the interface be better implemented using other interaction styles? If so state which interaction styles would be appropriate.

Answers to the Quick Quiz Questions

1. a) The five characteristics identified in the lecture are:

- The visibility of the objects of interest
- Ability to manipulate the objects of interest in a direct manner
- Incremental actions with rapid feedback
- Reversibility of actions
- All possible actions should be legal

b) Things put into a wastebasket stay available until the wastebasket is emptied, so putting something into the wastebasket is not necessarily irreversible but it isn't exactly a place to store your valuables.

2. a) Windows
Icons
Mice or Menus
Pull-down menus or Pointers

b) There is not evidence of windows, icons or menus in what we are told about the games.

3. This depends on the phone of course, but direct manipulation interaction is not common. On my phone the main interaction styles are menus (e.g. setting opinions) and question and answer dialogues (''Store this number?'' etc).