10

Interaction design

1 Introduction

To create a physical UI design, you need to understand how users interact with computer systems. In this chapter, we consider a model of the interaction process known as the **human action cycle**. This builds on the concept of tasks, actions, and goals introduced in Chapter 4. We then consider the model that a designer has of the computer system and how the UI can communicate it to facilitate interaction. Based on this understanding, we then discuss the use of metaphors in UI design.

2 The Human Action Cycle

Users tend to be goal oriented when they use a computer system. In other words, they have something particular they want to achieve by using it. The human action cycle (Norman, 1988) is a psychological model that describes the steps users take when they interact with computer systems. In particular, the cycle shows the way users perform actions and tasks to achieve their goals. We have adapted the human action cycle proposed by Norman and shall discuss its significance for understanding the interaction process.

2.1 The Details of the Human Action Cycle

The flow of the activities in the human action cycle is illustrated in Figure 10.1. This shows that the user:

- Forms a goal
- Creates and executes actions that move toward that goal
- Perceives and interprets the outcome of executing the actions to see whether the goal will be achieved as anticipated
- Recognizes that if the goal cannot be achieved, it may have to be reformulated and the cycle repeated

Thus, the human action cycle involves both cognitive and physical activities.

There are three main stages in the cycle:

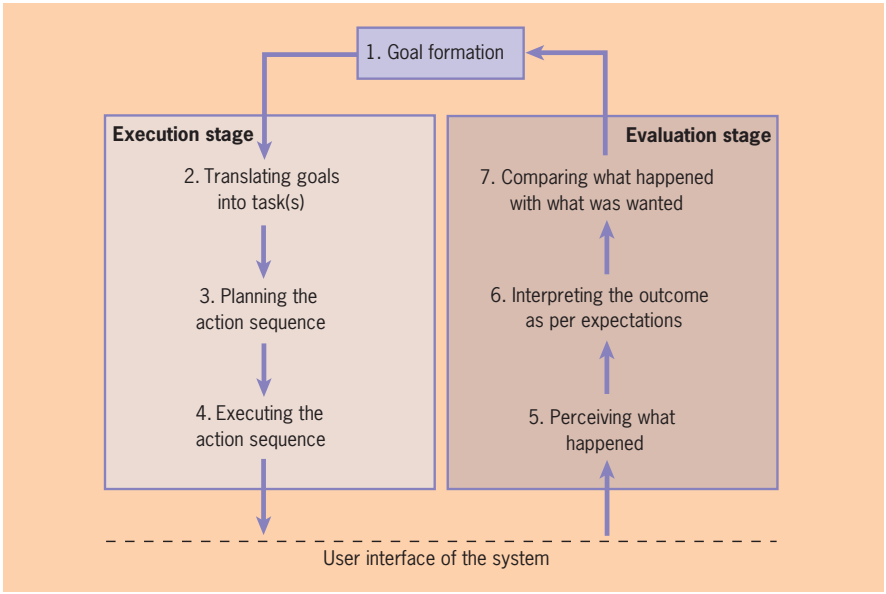

Figure 10.1 The human action cycle.

1. *Goal formation*. This constitutes step one, which is a cognitive activity. Users need to be able to form appropriate goal(s) to use the UI effectively.
2. *Execution stage*. This constitutes steps two, three, and four. During the execution stage, the users perform both cognitive and physical activities. Cognitive activities include translating the goals into tasks and planning the action sequences, whereas physical activities involve executing the sequences of actions.
3. *Evaluation stage*. This constitutes steps five, six, and seven. These are all cognitive activities involving checking what has happened and comparing it with the desired outcome (the goal(s) that were formed in step one).

In some cases, the complete cycle will last only a few seconds, and in others it may last hours. For example, if you want to type in a word, you will probably achieve this goal in a few seconds, whereas typing a report will take much longer. As you can see from this example, there will be a hierarchy of goals — for entering letters, words, paragraphs, and so on. Each of these will have associated tasks and actions. This illustrates why task analysis can be such a complex process.

Another interesting aspect of the human action cycle is the flexibility in the execution stage. Once a user has formed his or her goals, there may well be a range of different tasks and task sequences that will enable the user to achieve these goals, and for each of these tasks there may be a range of different possible actions. For example, the user can print from Word by selecting an icon, choosing a menu item, or by using a keyboard shortcut. A user chooses according to his or her knowledge of the UI and whether he or she wants to alter the print settings.

EXERCISE 10.1 (Allow 10 minutes)

Give an example of each of the seven stages of the human action cycle for sending a parcel at a post office.

DISCUSSION

Table 10.1 lists the steps that Jane, for example, would need to take. In reality, the situation may not be this simple. For example, the post office may be closed, in which case Jane would be unable to achieve her goal. In this situation, the goals may have to be reformulated (she may decide to send the parcel via a courier service) or discarded (she may decide not to send the parcel at all).

Table 10.1 The Human–Action Cycle — An Example

The seven steps of the human–action cycle	Example
1. Jane forms a goal.	Post a parcel.
2. Jane formulates the tasks.	Some of the tasks would be Prepare the parcel. Walk to the post office to send the parcel.
3. Jane specifies the actions.	Some of the actions would be Pick up the parcel. Walk to the front door. Open the front door. Lock the front door behind her.
4. Jane does the actions.	Jane hands over the parcel to the counter assistant. The counter assistant weighs the parcel. The counter assistant affixes the required postage stamps. Jane pays for the postage.
5. Jane perceives the outcome.	Jane observes that the counter assistant has deposited the parcel in the mailbag.
6. Jane interprets the outcome.	The parcel is now in the mailbag, ready for the journey to its destination.
7. Jane evaluates the outcome.	The goal of sending the parcel has been achieved.

EXERCISE 10.2 (Allow 10 minutes)

In Chapter 4, we explained how you should use the following three questions to carry out a cognitive walkthrough, as part of the task analysis process:

1. Is the correct action sufficiently evident to the user?
2. Will the user connect the description of the correct action with what she or he is trying to do?
3. Will the user know if she or he has made a right or wrong choice on the basis of the system's response to the chosen action?

How does the human action cycle justify asking these questions?

DISCUSSION

The first and second questions relate to step four in the human action cycle: executing the action sequence. It is possible to execute the action sequence only if the necessary actions are sufficiently evident to the user and the user can connect the description of the correct action with what she or he is trying to do.

The second question also relates to step three: planning the action sequence. The user's understanding of the actions that are available will influence the planning process. Thus, the single-headed arrow linking steps three and four in Figure 10.1 is a simplification.

The third question relates to steps five, six, and seven: perceiving what happened, interpreting the outcomes, and evaluating what happened. The quality of the feedback will influence whether steps five, six, and seven can be carried out successfully. Confusing or limited feedback may mean that the user cannot perceive or interpret what has happened.

As you can see, the human action cycle helps to explain why the three questions asked during cognitive walkthroughs are so powerful.

2.2 Using the Human Action Cycle to Influence the Design Process

You can see from the human action cycle how important it is to design a UI that helps the users achieve their goals. One way to critically evaluate this aspect of a UI prototype is to walk through the prototype, checking to see if it satisfies the requirements of a use scenario and asking questions based on the human action cycle. Answering these questions can provide you with various types of information:

- You may be able to predict difficulties that the users may face with the design and suggest modifications.
- You may be able to suggest suitable changes in the users' environment and the system's technology.
- You may be able to suggest necessary skills for the users when they work with the UI, or identify training needs.

This section was strongly influenced by Newman and Lamming (1995) and Hackos and Redish (1998).

In addition, you may come up with new requirements for the UI design or changes to the existing requirements. As we have emphasized throughout, developing UIs is an iterative process, so when you are designing, you may well have to go back to the requirements-gathering stage and make some changes. Table 10.2 lists some of the questions you could ask when walking through a prototype.

In Table 10.3 we suggest some solutions that you should consider when you encounter difficulties in the walkthrough and where in the book to look for further help. This list of possible remedies is not comprehensive. It is only indicative and should encourage you to come up with your own remedies, depending on your particular users, the tasks they are carrying out, and the environment in which they are working.

We shall introduce further design issues, principles, and guidelines in the remainder of Part 3, which you could apply depending on your particular circumstances. These include the choice of metaphor and interaction style; the choice of interaction devices; the effective use of text, color, sound, and moving images; and good GUI and web page design. All of these have an impact on a system's usability.

EXERCISE 10.3 (Allow five minutes)

Look at the TV remote control handset in Figure 10.2. Use the human action cycle to step through lowering the volume. Which stage in the cycle highlights the confusing nature of the design? How would you improve the design?

DISCUSSION

When you plan the action sequence, it becomes apparent that the labeling of the increase and decrease volume buttons is unclear. The danger is that a user will push the button indicated by the arrow. This will increase the volume. In Figure 10.2, the letter *V* for volume is on the two volume-control buttons. Although the buttons are shaped like up and down arrows for increasing and decreasing the volume, the letter *V* suggests a down arrow. When users scan the remote control for a down arrow, they perceive the *V* as a down arrow and press it. Unfortunately, the first *V* visible is on the up arrow button.

Instead of the large *V* on the volume-control buttons, "vol" in smaller letters could be used as a label. In general, a control's features — such as placement, labeling, size, and color — need to work together to show how to use the control. In this example, the shape and the label convey conflicting information.

3 Communicating the Designer's Understanding of the System

In Chapter 4, we considered the user's understanding of the UI. We referred to this as a mental model or, more briefly, as a model. We all develop these models, albeit unconsciously, from our day-to-day experiences. We then apply these models to

Table 10.2 Questions to Ask According to the Stage in the Human–Action Cycle

Step in the goal formation and execution stages	Questions
1. Forming a goal.	Do the users have sufficient domain and task knowledge and sufficient understanding of their work to form goals? Does the UI help the users form these goals?
2. Translating the goal into a task or a set of tasks.	Do the users have sufficient domain and task knowledge and sufficient understanding of their work to formulate the tasks? Does the UI help the users formulate these tasks?
3. Planning an action sequence.	Do the users have sufficient domain and task knowledge and sufficient understanding of their work to formulate the action sequence? Does the UI help the users formulate the action sequence?
4. Executing the action sequence.	Can typical users easily learn and use the UI? Do the actions provided by the system match those required by the users? Are the affordance and visibility of the actions good? Do the users have an accurate mental model of the system? Does the system support the development of an accurate mental model?
5. Perceiving what happened.	Can the users perceive the system's state? Does the UI provide the users with sufficient feedback about the effects of their actions?
6. Interpreting the outcome according to the users' expectations.	Are the users able to make sense of the feedback? This depends on both the users and the UI. If the task is complex and the users have not been trained, they may be unable to interpret the outcome of their actions. Does the UI provide enough feedback for this interpretation?
7. Evaluating what happened against what was wanted.	Can the users compare what happened with what they were hoping to achieve? This depends on their domain and task knowledge and their understanding of their work, as well as the feedback from the UI.

Table 10.3 Solutions to Problems That May Arise When Walking through the Human Action Cycle

Problem	Cause	Solution	Where to look for help
The user cannot form the goal or formulate the tasks and action sequence needed to achieve it.	Users have insufficient domain or task knowledge, or do not documentation or the skills required for the work.	Provide additional documentation or training in the domain.	
	The UI does not provide the necessary support for the users.	Redesign the UI so that it suggests possible goals, tasks, and actions.	Chapter 11 for interaction styles.
The users are unable to execute the desired action sequence for the task.	The users find the UI difficult to use.	Provide additional documentation or training in the use of the UI.	Chapter 5 for an explanation of visibility, affordance, and feedback.
		Ensure the UI is well designed and uses all the appropriate design principles and guidelines.	Chapter 9 for an explanation of design guidances, design principles, and acessibility guidelines.
	The UI does not support the necessary actions.	Redesign the UI so it supports the actions that users are likely to need.	Chapter 8 for conceptual design. Also, Chapter 4 on task analysis.
	The users have an incorrect mental model of the system.	Redesign the UI so that it supports the users' mental models.	The discussion of mental models and metaphors later in this chapter.
The users are unable to perceive the system's state.	The UI provides insufficient or inappropriate feedback about the effect of the users' actions and the current system state.	Redesign the UI so that it provides adequate feedback.	Chapter 5 for an explanation of feedback. Chapter 9 for design principles.
The users are unable to interpret the outcome or compare the outcome with their goals.	Users have insufficient domain or task knowledge or do not have the skills required for the work.	Provide additional documentation or training in the domain.	

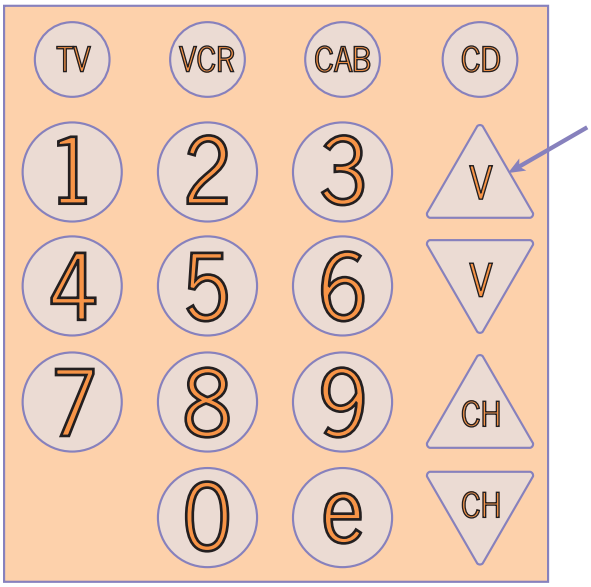

Figure 10.2 Lowering the volume on a television.

similar situations in order to predict outcomes. To interact effectively with a UI, we need an accurate model of its behavior. Here we consider a view developed by Norman and Draper (1986). It is illustrated in Figure 10.3.

Norman and Draper described the three elements in Figure 10.3 as follows:

- *The designer's model*. This is the designer's understanding of a system. It includes both the structural and functional aspects of the system — both how it works and how to operate it. The designer's model is usually complete and correct.
- *The user's model*. This is the user's understanding of the system. Typically it contains mainly functional information, possibly incomplete and incorrect.
- *The system image*. This includes the UI, supporting documentation, training, and so on. The purpose of the system image is to communicate enough of the designer's model to the users so that they can create an accurate mental model of the system. The design of the system image should also take into account the users' existing knowledge of the area.

Let's look at each of these elements briefly, considering how the system image can promote a more accurate user model and hence improve the interaction between the user and the computer system.

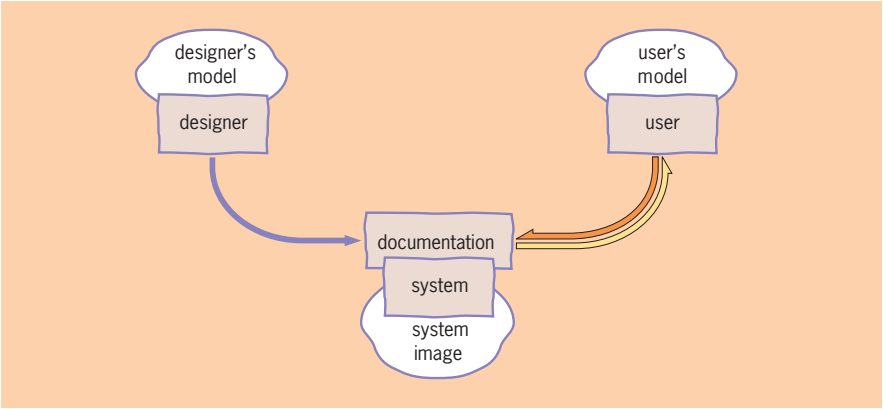

Figure 10.3 The designer's model, the user's model, and the system image. (From Norman and Draper, 1986.)

3.1 Designer's Model

The **designer's model** is an explicit and consciously developed model, derived from:

- The structure and organization of the UI, as represented by the content diagram (for more detail on the content diagram and conceptual design, see Chapter 8)
- An understanding of the domain and the system's purpose and functionality
- An understanding of users' requirements, their characteristics, tasks, and expectations of the new system
- The system's technology and the environment
- Any hardware platform or implementation constraints or any trade-offs

For example, the designer's model of a car might include a knowledge of the way the engine, brakes, steering, and other parts operate; a knowledge of the design of other cars; and a knowledge of how to drive a car. The designer's model is typically not the same as the user's model, as it contains much more detail than the user would normally require. As cars become increasingly reliable, most drivers just need to know how to drive the car.

This list indicates that the requirements-gathering activity of the life cycle is important to the development of the designer's model. It is during this stage that the designer meets the users, observes them, and talks to them.

3.2 System Image

The **system image** is the means by which the functionality and state of a system is presented to the user, and this can be through a variety of channels: the UI, training, documentation including instruction manuals, and online help.

The users acquire their knowledge of the system from the system image. The most important part of the system image is the UI since, through interaction with it, users build up their own mental model of the system and how it works. Users do not always read instruction documents, use the help function, or receive training. Therefore, the maximum burden of influencing the user's mental model is on the UI.

Norman and Draper (1986) noted:

> *[I]n many ways, the primary task of the designer is to construct an appropriate System Image, realising that everything the user interacts with helps to form that image: the physical knobs, dials, keyboards, and displays, and the documentation, including instruction manuals, help facilities, text input and output, and error messages.*

They go on to discuss the implications for the system designer:

> *If one hopes for the user to understand the system, to use it properly, and to enjoy using it, then it is up to the designer to make the system image explicit, intelligible, and consistent. And this goes for everything associated with the system.*

Because it is the system image and largely the UI that influence the users' mental models, the designers needs to translate their knowledge of users' mental models onto the design of the UI. For an effective, efficient, and satisfying user–computer system interaction, it is important for you, as a designer, to understand the users' mental models.

EXERCISE 10.4 (Allow five minutes)

Choose a system that you use either at home or at your workplace. List the constituents of its system image. How well does the system image communicate the designer's model? You might like to consider, say, a toaster.

DISCUSSION

Caroline writes: The system image of the toaster in my kitchen comprises the following:

- Written instructions on the box that the toaster came in and in a small leaflet in the box (both long since discarded)
- A small rotary dial with some markings
- Two extra buttons, one marked RESET and another with a snowflake symbol
- A slot across the top long enough and wide enough for a slice of bread
- A lever-knob on the side that lowers the bread into the slot
- Feedback from the system — a clicking noise while the bread is being toasted, the sound of the bread being popped up, the appearance of the freshly cooked toast, and the smell of toast (or, sometimes, of burning bread)

Usually I rely on the putting the bread in the slot and pushing down on the lever knob. Sometimes I recall that RESET will pop up the toast if I press it. I often forget to press the snowflake button (the snowflake is a symbol for frozen bread), so I have to toast the bread twice (two loops around the human action cycle) to get the desired level of doneness. Sometimes this goes wrong, and I end up with burnt toast. The two extra buttons accurately communicate that two further functions are available, but they are less satisfactory as reminders of what the functions are or when the functions might be needed. The written instructions are no longer any use at all because I have thrown them away.

For brevity, we concentrate on the UI, but as a designer you should consider all the system image elements.

3.3 How the User Interface Enables the User to Develop an Accurate Mental Model

As Figure 10.3 illustrates, the UI needs to take the following points into account:

1. *The existing user model.* The users will have expectations of the UI, and if these are not met they may experience difficulties.
2. *The designer's model.* The UI needs to communicate effectively the relevant parts of the designer's model, in particular the functional aspects — how the users need to operate the UI in order to achieve their goals.

In this section, we concentrate on taking the existing user model into account when designing a UI. Communicating the designer's model through the UI is a matter of good design.

When confronted with a new computer system, users — often unconsciously — "run" their existing mental model to interpret the new situation, explain it, and make predictions about the outcome of their interaction with the system. The mental model helps the users to answer questions such as "What am I now seeing?" "What did the system just do?" and "What have I done to make it do that?"

EXERCISE 10.5 (Allow five minutes)

Consider the process of buying a train ticket from a railway clerk. You probably have a mental model of this process. How would this mental model help you purchase a ticket from an automatic ticket machine?

DISCUSSION

Your mental model probably includes telling the clerk where you are going, specifying the type of fare (single, return, off peak, etc.), and paying. You probably then expect the clerk to give you the ticket and a receipt. Therefore, in your head, you know what is involved in buying a ticket. You have a mental model of the process.

Suppose you want to buy a train ticket from a self-service ticket machine. From your original mental model, you probably know what you must do to get the ticket. You might anticipate the following: pressing various buttons to enter the information that you would have said to the railway clerk, inserting the money or a credit card into a slot and taking the ticket and receipt from other slots in the machine. In this way you are able to apply your old mental model of buying a ticket from a railway clerk to this situation of interacting with a machine.

This may work perfectly well, or it may cause problems if the automated machine wants information in a different order such as requiring ticket type before destination.

When users interact with a new computer system, they apply their old (existing) mental models and expect the new system to work in a way that their mental models suggest. For this reason, it is important to take into account the users' prior knowledge and experience when designing a new UI. Users may become uncertain if their existing mental models do not explain how to use a new system (unless they can develop a new model quickly). They may proceed hesitantly and become frustrated, perhaps abandoning the new system entirely because it is unfamiliar.

EXERCISE 10.6 (Allow 10 minutes)

How might an inaccurate mental model of a television and video recorder create problems for users? Relate these problems to each stage in the human action cycle.

DISCUSSION

Problems may arise as follows:

- *Forming a goal*. The users may not know what is a reasonable goal. For example, they may want to surf the Web on a television that is not equipped for this.
- *Translating the goal into a task or a set of tasks*. Users may not have a clear view of the tasks needed. For example, with some video recorders it is possible to record television programs by entering a code; others will require entry of the date, start time, end time, and channel.
- *Planning the action sequence*. The method for carrying out a particular task may be unexpected. For example, programming the time to record a television program may involve stepping through the times using arrow keys on the remote control, which may be a surprise to a user who is used to typing in the times using buttons on the remote control.
- *Executing the action sequence*. The UI may not appear to support the actions required. For example, the user may expect the volume control on a handset to be represented by two buttons, one with a $\wedge$ for louder, the other with a $\vee$ for quieter. In reality, these may be represented by two buttons, with a + and –, respectively.

- *Perceiving what happened*. No problems resulting from an incorrect mental model should arise here.
- *Interpreting what happened*. If the feedback does not match the users' expectations, they will probably misinterpret it. For example, suppose you expect a red light to flash when an error has occurred when you are programming a video recorder, but on this particular video recorder the light flashes when it is programmed successfully. You might assume that you have failed and keep on trying.
- *Comparing (evaluating) what happened with what was wanted (the goal)*. A user who has incorrectly interpreted the feedback may not be sure whether she or he has satisfied the goal. In the previous example, the user is unsure whether he or she has successfully programmed the video recorder.

To take into account the user's model, you need to consider a number of design issues:

- *Matching the users' expectations*. As a designer, if you are aware of the users' expectations, then you will be able to create a UI design that is more immediately understandable and easier to learn. It may not always be possible for you to exactly match the UI to the users' expectations, either because of technological constraints or because the system requirements are different, but it is always worth trying.
- *Shaping the user's model*. A user may find a system daunting if it includes a lot of functions. One way to overcome this is to organize the UI so that the familiar functions are easy to pick out. This means the user can gain confidence in using these functions before moving on to the unfamiliar functions. For example, on a cell phone it should be immediately apparent how to dial a number, so users can try this before experimenting with more advanced features such as changing the ringing tone.
- *Flexibility*. Typically, the user's model is not static but grows through interaction with the UI. A user slowly develops a picture of what is in the system, what it is capable of doing, and how it responds to various actions. Since user's models change with time, the designer should create a flexible UI — for example, providing different ways of accomplishing the same tasks to suit both novices and experts.

In conclusion, the designer's aim is to create a UI that effectively communicates his or her model while reflecting an awareness of the user's existing mental model.

4 Using Metaphors to Develop Accurate Mental Models

In the previous section, we emphasized the importance of designing a UI that matches the users' mental models or supports the adaptation of the users' mental models toward the one desired. In this section, we show you how metaphors can help users develop accurate mental models.

4.1 The Benefits of Metaphor

Hackos and Redish (1998) and Neale and Carroll (1997) strongly influenced the content of this section.

Webster's Third New International Dictionary defines **metaphor** as follows:

> *A figure of speech in which a word or phrase denoting one kind of object or action is used in place of another to suggest a likeness or analogy between them.*

There is a sense in which everything generated by a computer is metaphorical. Even the text on the screen could be seen as a metaphor for text on paper, and the command line interaction style, which will be described in Chapter 11, could be seen as a metaphor for a conversation between the computer and the user. All UIs are artificial environments created by the computer, which we can understand only because of our experience of the physical world.

In this context, we use the word metaphor in a more restricted way, to refer to the metaphorical use of the following elements. These elements are often combined to create an integrated environment. Probably the best-known example of this is Microsoft Windows:

- Words on the screen, such as screen titles and labels (for example, the use of the word "shop" in an e-commerce web site)
- Static images and icons, such as the printer icon in Word
- Interactive graphics, such as the calculator that comes with Windows

These elements are metaphorical when the designer deliberately draws on the users' existing knowledge to enable the user to formulate an accurate mental model of the UI.

Metaphors are often used to unify UIs. Consider the example of e-shopping. The experience of shopping online — selecting goods, proceeding to the checkout, and entering details of your credit card and delivery address — is very different from the shopping experience in a physical store. To help bridge this gap, many e-shopping (or e-commerce) sites present a shopping metaphor by using terminology, icons, and graphics that are analogous to shopping in the real world. For example, Figure 10.4 shows a page from the Amazon web site. In this, the shopping basket acts as a metaphor. Shoppers are familiar with putting items to be purchased into a basket when they shop in a store. When they are shopping on the Amazon web site, it is possible to put the selected items into the virtual basket. The content of the basket is then displayed before the consumers purchase the items. The shopping metaphor is further reinforced by the use of metaphorical language, such as "shop" and "store."

EXERCISE 10.7 (Allow five minutes)

Explain why the following are being used metaphorically and how this enables the user to develop a more accurate mental model of the UI.

- The word "shop" in an e-commerce web site
- The printer icon in Word
- The calculator that comes with Windows XP

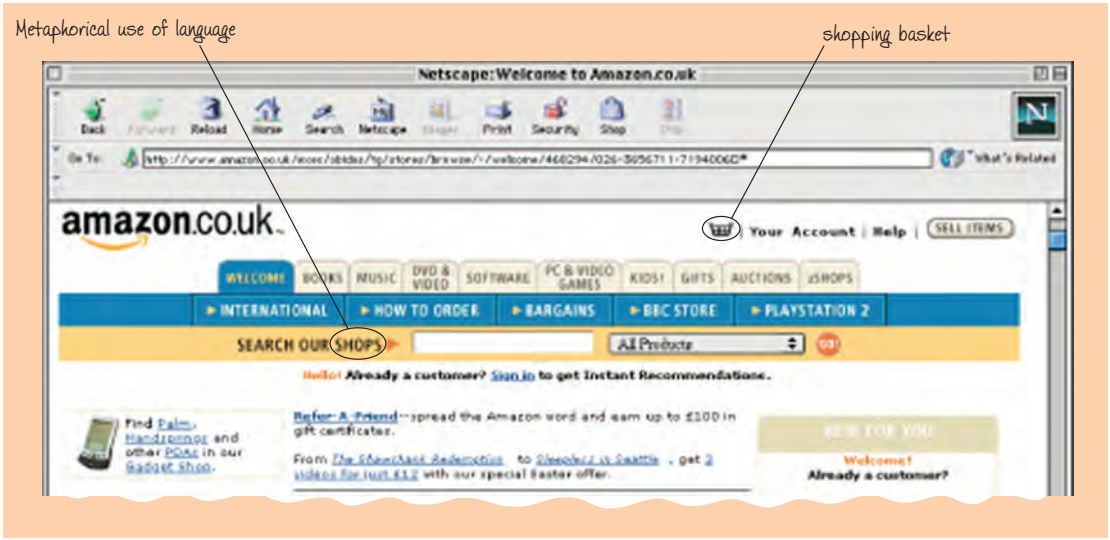

Figure 10.4 A shopping basket metaphor. (From www. amazon.co.uk, December 18, 2000.)

DISCUSSION

- Most people who are likely to use the web are also likely to have visited an ordinary non-web shop. They know what to do in a shop: for example wandering around to see what is for sale, picking up the items they want, and going to the checkout to pay. All this is brought to mind when the word "shop" is used in an e-commerce site, so the users know what to expect when they are using the site.
- The printer icon illustrates a printer, probably not exactly like the user's printer but near enough to be recognizable. As the most common use for a printer is printing, it is likely that the user will assume that selecting this icon will cause a document to be printed.
- The calculator looks very much like a conventional handheld calculator. Thus, users who are familiar with handheld calculators will assume that it operates in an analogous manner, which it does. This makes it extremely easy to use. It also presents an integrated and consistent UI.

4.2 Problems with Metaphor

As you can see, metaphors can be extremely powerful, and it would be easy to conclude that they are the solution to all UI problems. Unfortunately, it is not that simple. The following sections describe two important classes of problem that can arise when using metaphors.

▶ Metaphors That Do Not Match the Users' Experience of the World

When you choose a metaphor, you deliberately choose one that will exploit the users' existing knowledge of the physical analogue. This knowledge brings with it expectations about how to interact with the metaphor.

However, many metaphors do not behave as the user might anticipate. For example, the Windows calculator can be changed from a conventional calculator to a scientific calculator. Although this is unexpected, it is extremely useful. Thus, metaphors are often developed beyond what occurs in the physical world, taking advantage of the power of the computer. This means that the metaphor is more useful, but it risks becoming less usable. For example, the researcher who first named the Microsoft graphical UI "window" may have considered it a metaphor, from windows in a house. When you open a window, a new view emerges, just as when you open the curtains in a house. However, the metaphor starts to break down quickly, because you cannot reduce the windows in a house to icons or move them around the walls.

Adhering to a metaphor too rigidly can be just as problematic, possibly requiring the inefficient use of the system. For example, pressing the printer icon in Word is a slow operation, as in most cases it requires you to move your hand from the keyboard to the mouse, move the mouse pointer, and click a button. For experienced users, a faster way to achieve the same thing is to press ⟨ctrl⟩ p, but this bypasses the metaphor.

As a designer, you need to decide how consistent a metaphor is going to be with the physical analogue and whether the similarities or differences are going to significantly decrease the usability of the UI. If they are, you should consider alternative approaches.

Box 10.1

The Apple Trash Can

The trash can on the Apple desktop is a classic example of a designer's metaphor that confounded users' expectations. The trash can is illustrated in Figure 10.5. Apple designers worked with two rules for the design of desktop objects: that each object has only one function and that they minimize the number of objects on the desktop. When they created the desktop metaphor, they chose to compromise between creating a minimum number of desktop objects and keeping one function per object. They gave the trash can two functions: to delete files and folders from the desktop and to eject floppy disks. This confused the users. The trash can being associated with deleting files and folders is understandable, and users had no trouble with that. However, they could not associate it with ejecting a disk. Some users thought that they would lose the contents of the disk if they used the trash can.

Figure 10.5
The Apple trash can.

▶ Metaphors That Relate to Objects or Concepts That Are Outside the User's Experience

There is no point in using a metaphor if the physical analogue is outside the user's experience. For example, many dialog boxes in Windows contain "radio buttons." This metaphor comes from early radios that used mechanical buttons for selecting wavelengths. The physical design meant that only one of these could be depressed at a time. However, most buttons on radios are now electronic and do not have this built-in limitation, so increasingly users will not encounter buttons of this type. Thus, the metaphor is becoming dated. This is probably why radio buttons are increasingly referred to as "option buttons."

There are many reasons why your users may be unfamiliar with a particular metaphor, including, for instance, cultural differences or age differences. These differences should become apparent during requirements gathering.

When users are unfamiliar with a metaphor, it may not mean that they are unable to use it; rather they will have to learn it, just like any other UI. How many users are able to use the scroll bar because it is like a scroll of paper? Not many, we suspect. Users are either taught or work it out through a process of trial and error. If too many of your users are doing this, it calls into question whether it is worth using the metaphor at all.

4.3 Choosing a Suitable Metaphor or Set of Metaphors

As with other activities in the design of UIs, the approach to the choice of metaphors should be iterative and user centered. You may come up with several to start with, then retain some and reject others. During requirements gathering, the analysis of users, tasks, and environment may provide ideas for metaphors that could be used in the UI design. In particular, the task scenarios and use cases are often a good source of metaphors. Conversations during requirements gathering may also give you an insight into the metaphors that users tend to employ.

EXERCISE 10.8 (Allow five minutes)

Imagine you are designing a web site for a supermarket. The web site allows users to select items they would like to purchase and have them delivered to their homes. Suggest a possible interactive metaphor based on graphics, and identify two possible problems with its use.

DISCUSSION

One metaphor would be an interactive image of the supermarket's layout that allows users to focus on particular aisles and shelves within the aisles. Users then choose their purchases. This would work well for customers who know the store well, but it could be confusing for those who do not — how would they know where to look? This approach could also be slow, as it would probably mean

downloading several web pages to make the purchases. It might be better to have an alphabetical list of the categories of products available. After all, most customers are likely to know the names of items and can just search the list.

When you are designing a metaphor, it is a good idea to walk through some use scenarios with the users. You do not have to draw the UI design in detail to get useful feedback; the idea is just to get a feel for how the metaphor may be presented. You can *then* make appropriate changes. In particular, you are looking to find out how easily those who are unfamiliar with the physical analogue can learn the metaphor.

We look at more examples of metaphor in Chapter 16.

Though metaphors are popular with many designers, you should not feel compelled to use one. Good UI design does not always start with a suitable metaphor. The important issue is not whether there is a metaphor or even a whole set of metaphors incorporated into a UI, but whether the interface is usable.

5 Summary

In this chapter, we have considered a model of the interaction process known as the human action cycle, and we have discussed the use of mental models and metaphors for UI design. In the next chapter, we turn our attention to interaction styles, which are the different ways by which a user and computer system can interact.