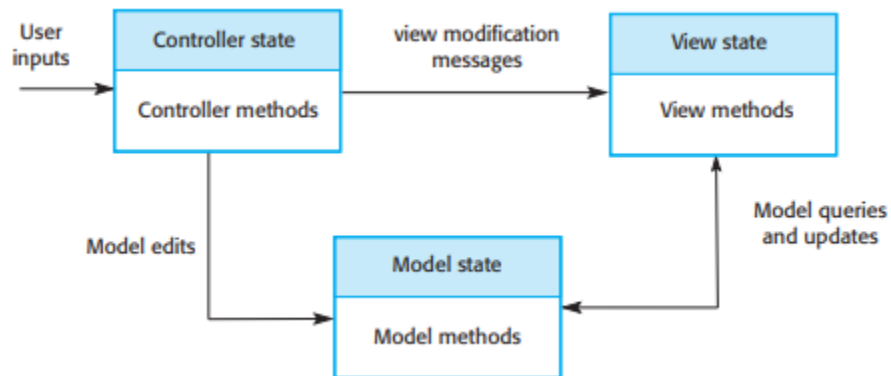

Information presentation

All interactive systems have to provide some way of presenting information to users. The information presentation may simply be a direct representation of the input information (e.g., text in a word processor) or it may present the information graphically. A good design guideline is to keep the software required for information presentation separate from the information itself. Separating the presentation system from the data allows us to change the representation on the user's screen without having to change the underlying computational system. This is illustrated in Figure 29.5.

The MVC approach (Figure 29.6), first made widely available in Smalltalk (Goldberg and Robson, 1983), is an effective way to support multiple presentations of data. I introduced this approach in Chapter 7, where I discussed the design patterns used in its implementation. Users interact with each presentation in a style.



that is appropriate to the presentation. The data to be displayed is encapsulated in a model object. Each model object may have a number of separate view objects associated with it where each view is a different display representation of the model.

Each view has an associated controller object that handles user input and device interaction. Therefore, a model that represents numeric data may have a view that represents the data as a histogram and a view that presents the data as a table. The model may be edited by changing the values in the table or by lengthening or shortening the bars in the histogram.

To find the best presentation of information, you need to know the users' background and how they use the system. When you are deciding how to present information, you should bear the following questions in mind:

1. Is the user interested in precise information or in the relationships between data values?
2. How quickly do the information values change? Should the change in a value be indicated immediately to the user?
3. Must the user take some action in response to a change in information?
4. Does the user need to interact with the displayed information via a direct manipulation interface?
5. Is the information to be displayed textual or numeric? Are relative values of information items important?

You should not assume that using graphics makes your display more 'interesting'. Graphics take up valuable screen space (a major issue with portable devices) and can take a long time to download if the user is working over a slow, wireless connection on a mobile device.

You should use text to present information when precise information is required and the information changes relatively slowly. If the data changes quickly or if the relationships between data rather than the precise data values are significant, then you should present the information graphically.

When large amounts of information have to be presented, abstract visualisations that link related data items may be used. This can expose relationships that are not obvious from the raw data. User interface designers should be aware of the possibilities of visualisation especially when the system user interface must represent physical entities. Examples of data visualisations are:

1. Weather information, gathered from a number of sources, is shown as a weather map with isobars, weather fronts and so on.

2. The state of a telephone network is displayed graphically as a linked set of nodes in a network management centre.

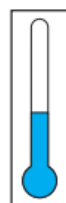
3. The state of a chemical plant is visualised by showing pressures and temperatures in a linked set of tanks and pipes.



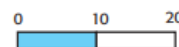
Dial with needle



Pie chart



Thermometer



Horizontal bar

4. A model of a molecule is displayed and manipulated in three dimensions using a virtual reality system.

5. A set of web pages is displayed as a hyperbolic tree (Lamping, et al., 1995).