gram pertinent to the hotel information in Table 3.2 is shown in Figure 3.82: for a much larger collection of 24 hotels the corresponding Venn diagram (Figure 3.83) *might* quickly allow a traveller to identify the hotels which satisfy his or her needs. For example, it can be seen that if all facilities are required (swimming pool + golf + restaurant) then four hotels are candidates. Interaction can add considerable value to a Venn diagram.

FIGURE 3.82
A Venn diagram representation of the hotels listed in Table 3.2

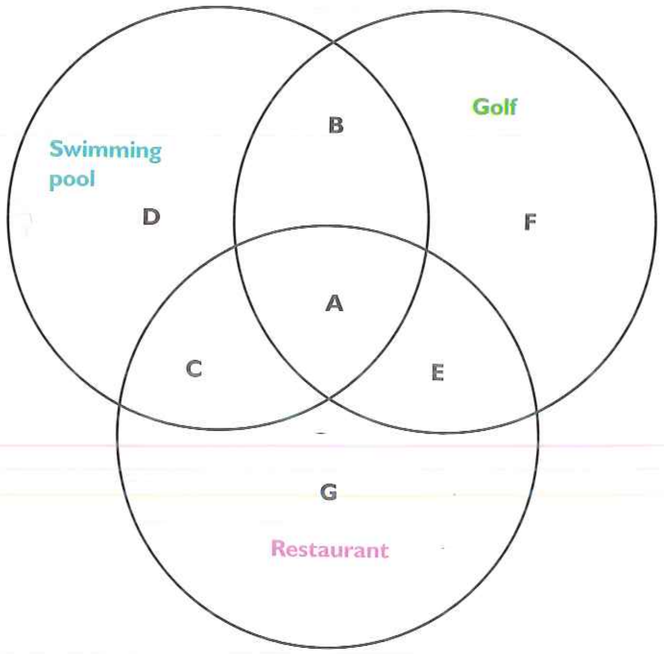
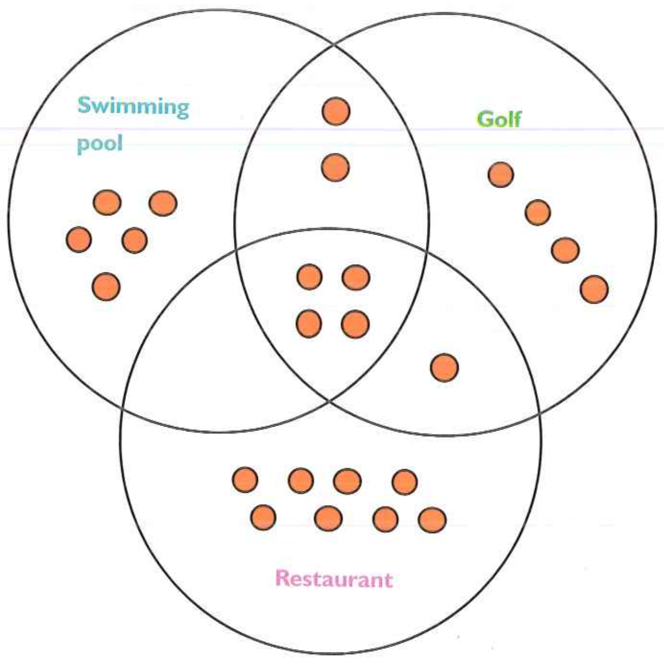


FIGURE 3.83
A Venn diagram representation of the attributes of 24 hotels



InfoCrystal

An improvement on the Venn diagram for the representation of logical relations was proposed by Spoerri (1993). Figure 3.84 shows a progression from a three-attribute Venn diagram to a three-attribute InfoCrystal. All possible logical relations between the three attributes are represented by regions within the crystal. The same colours are used to represent groups of houses as are used in the Attribute Explorer example of Figure 3.58. Thus, the region indicated by an asterisk in Figure 3.84 corresponds to, and can be used as a means of selecting (i.e. as a visual query), those houses which satisfy requirements on price and garden size, but not the number of bedrooms. Spoerri, however, complemented the simple representation of Figure 3.84 by adding interior icons (Figure 3.85) which indicate, by their shape, the number of criteria satisfied (circle = 1, rectangle = 2, triangle = 3) and, by an inscribed number, the number of items in that class. Thus, for the 24 hotels represented in Figure 3.83, four offer all facili-

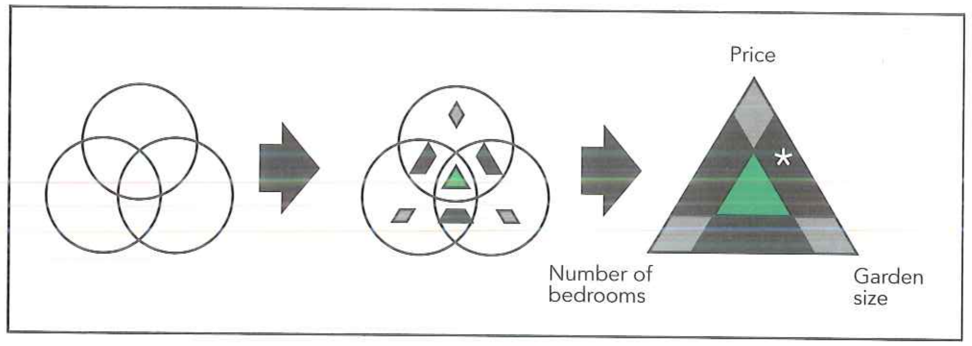


FIGURE 3.84 The development leading from a Venn diagram to an InfoCrystal. The InfoCrystal illustrated allows visual queries to be made concerning price, garden size and number of bedrooms (see the Attribute Explorer of Figure 3.58) The asterisk represents houses satisfying criteria on price and garden size but not number of bedrooms

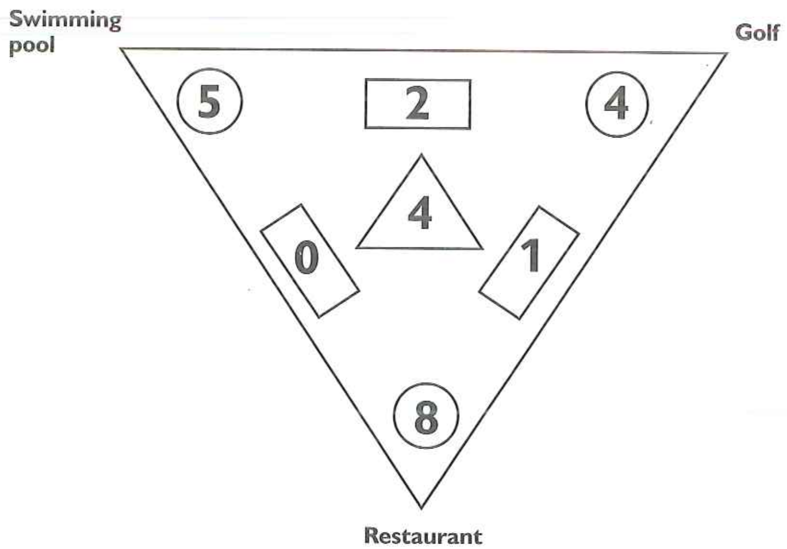


FIGURE 3.85
An InfoCrystal representation of the hotel data of Figure 3.83

ties and the number which offer both golf and a restaurant but no swimming pool is one. Spoerri's InfoCrystal can be extended to more than three attributes, but above four the effort needed to comprehend the representation and to readily formulate a query is probably such that only the specialist user would be able to do so.

Cluster maps

Figure 3.86 shows a cluster map (Fluit *et al.*, 2003) relevant to the hotel data represented by the Venn diagram of Figure 3.83 and the InfoCrystal of Figure 3.85. The labeled nodes represent the three attributes and carry a label identifying that attribute (e.g. 'restaurant') followed by a number indicating how many hotels possess that attribute. Note that the three numbers do not add up to 24, the number of hotels, because a hotel with a restaurant and a gym, for example, will contribute to the count on both those nodes. Associated with each node are one or more circles containing a number of yellow circles, each representing a

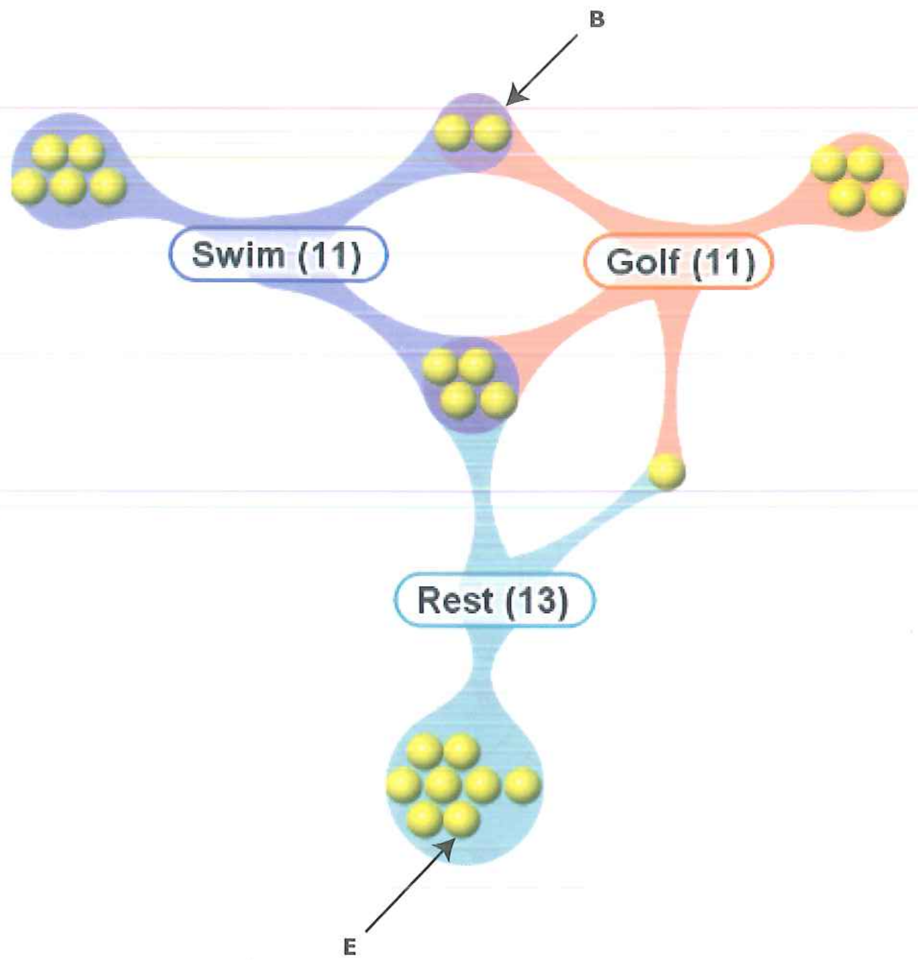


FIGURE 3.86
A cluster map representation of the 24 hotels represented in the Venn diagram of Figure 3.83 and the InfoCrystal of Figure 3.85
(Courtesy Christiaan Fluit, Aduna)

hotel. Each hotel is placed in a circle connected to all the attributes that the hotel possesses. Thus, hotel E has only a restaurant, so is placed in the circle associated only with the restaurant node, but hotel B appears in a circle attached to both the swimming pool node and the golf node because it possesses those two, and only those two, facilities. It is therefore reasonably obvious from the cluster map of Figure 3.86 that there are no hotels which possess both a swimming pool and a restaurant but no golf course. Mouse-over action can disclose more details of each hotel.

The test of any technique which represents logical data is how well it scales to many classes. Some impression of the cluster map's capability in this respect can be gained from an example representing 24 hotels each characterized by *four* attributes (Figure 3.87). The absence of any circular area connected solely to the gym node means, for example, that any hotel that has a gym will always possess at least one of the other three facilities. A cluster map can support many different kinds of interaction including the highlighting of all circles associated with a class node and the animated modification of a map following the inclusion of a new class (Fluit *et al.*, 2003).

3.2.3 Tree representations

In all the examples of relation representation we have discussed so far there has been no restriction upon what is connected to what other than it be meaningful.

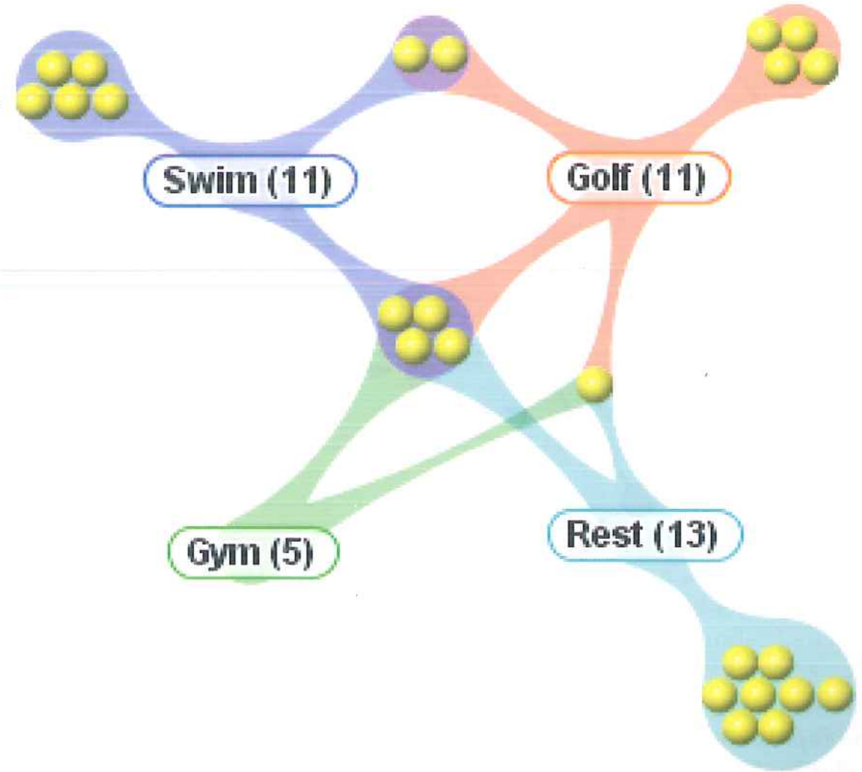


FIGURE 3.87
A cluster map representation of 24 hotels, each described by four attributes
(Courtesy Christiaan Fluit, Aduna)

The term 'network' is often applied to a node-and-link type of relation to which no topological restrictions apply. Examples include the simple network of Figure 3.73 representing telephone calls and the far more complex network of lines representing mortgage arrangements in Figure 3.74. There is, however, a class of networks having significant practical application in which the topological restrictions that apply allow a new family of representations to be derived. The restriction is that the network be a tree.

The mathematical definition of a tree is a network of nodes and links so connected that no loops are present. In other words you cannot start at a node and trace a path which arrives back at the same node without a direct retrace of steps. The reason for the considerable interest in trees is that many situations are described by hierarchical relations. To introduce relevant terminology we examine the simple tree of Figure 3.88. One node has been designated the root node: we use the term 'designated' because you can take hold of *any* node of a tree and call it the root node. In practice, however, a tree often refers to a hierarchy in which the designated root node is the president of a company or perhaps a department store comprising a number of product lines. If there is indeed a hierarchy, then all nodes except the root node are associated with a superordinate node and, if associated with subordinate nodes, these are often referred to as 'children'. If a node has no subordinate node it is called a 'leaf node'.

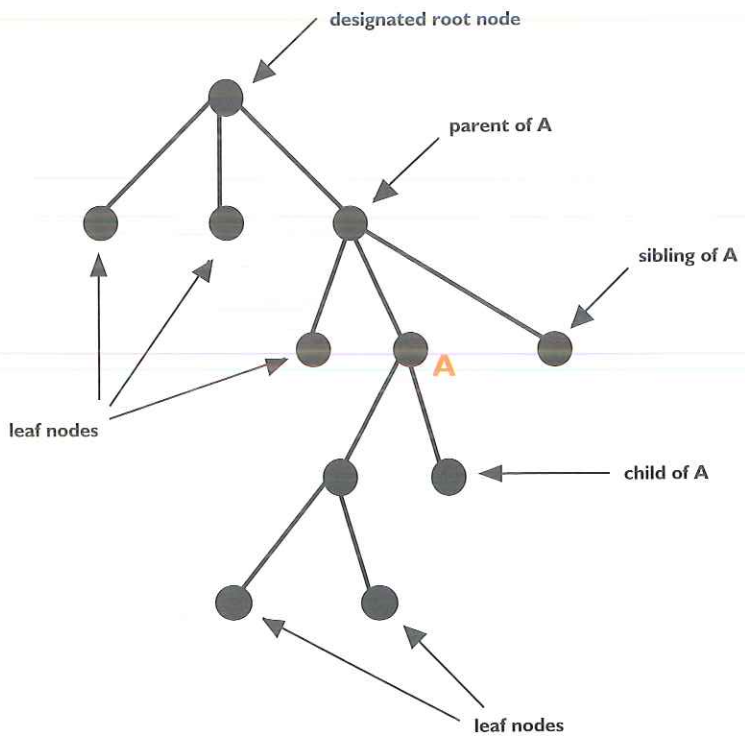


FIGURE 3.88
A tree

If some data describes a tree, the immediate question is how such data can usefully be represented and explored. What is wrong, however, with the representation shown in Figure 3.88? If it is satisfactory we need look no further!

Cone tree

The problem with the tree representation of Figure 3.88 is that in most practical cases the tree has many levels: even with an average fan-out (i.e. the number of subordinate nodes associated with a node) as small as three, the space needed to display a tree can become huge, mainly in the horizontal direction. A solution is to imagine the original tree (Figure 3.89(a)) to be rearranged into a 3D structure such that all nodes subordinate to a given node are arranged in a horizontal circle which, together with the superordinate node, forms a cone, as shown in Figure 3.89(b). The resulting 2D view of that structure, called a cone tree (Robertson *et al.*, 1991), is now more compact than the original representation of Figure 3.89(a) and, notwithstanding some occlusion, is easier to handle. The user of a cone tree may, for example, wish to see the reporting path of an employee within the organization represented by the cone tree, in which case entry of the employee's name by some means will bring about any cone rotations needed to position that employee and his reporting path to the foreground. A horizontal orientation of the cone tree, called the cam tree, may be more convenient for the display of node names (Figure 3.90). No record is available of any evaluation of the usability of the cone tree concept.

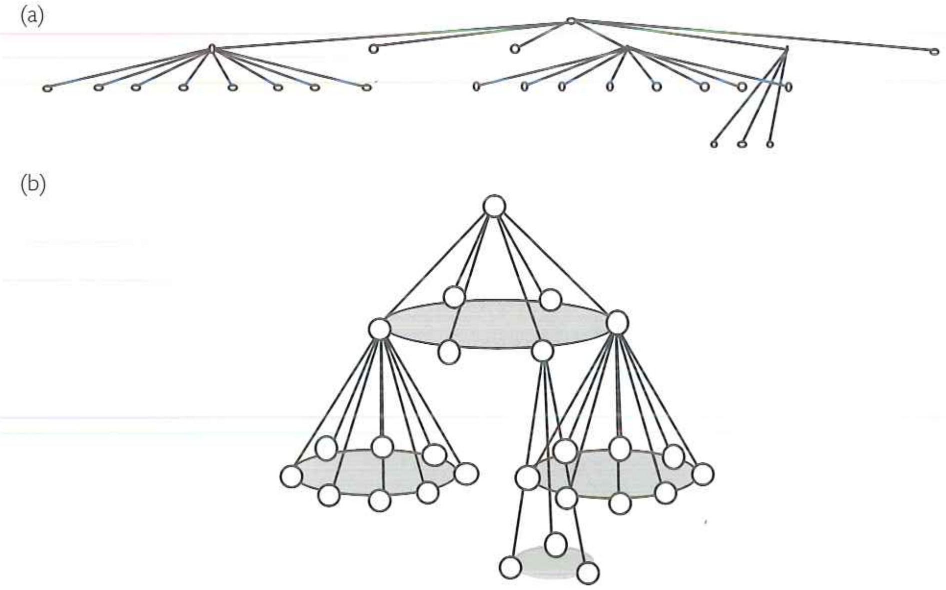
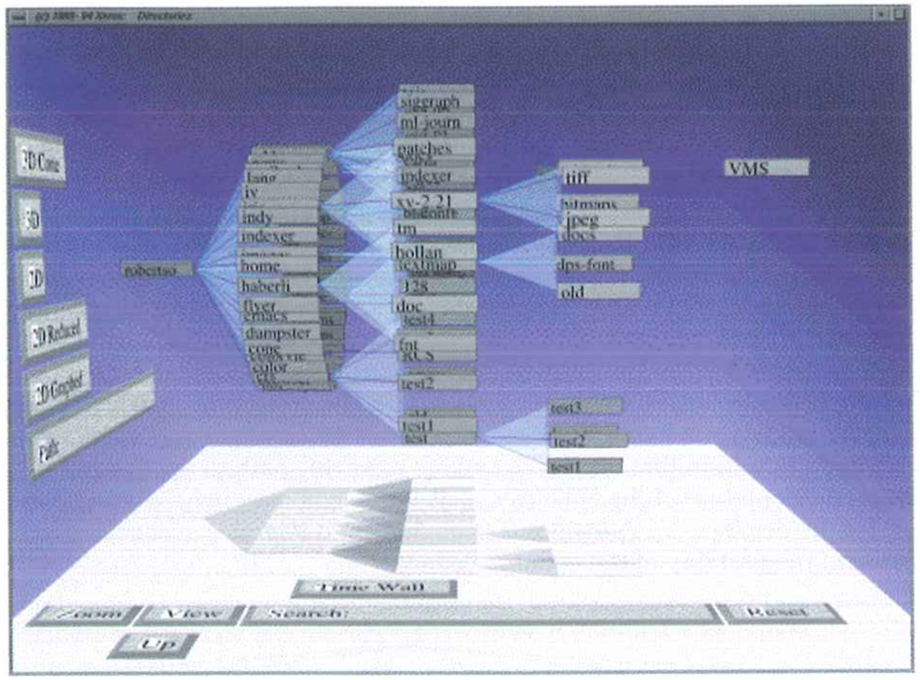


FIGURE 3.89
(a) A tree,
(b) the corresponding cone tree

Tree maps

An alternative representation of a tree is the tree map (Johnson and Shneiderman, 1991). Its derivation from the original tree representation is straightforward and is illustrated in Figure 3.91. Starting with the designated root node one draws a rectangle passing through that node. Usually, to make efficient use of display area, the rectangle will be made as large as conveniently possible. Within that rectangle are smaller rectangles, one for each of the nodes that are immediately subordinate to the root node. This construction is repeated until all nodes are accounted for. There is no constraint (except the resolution of

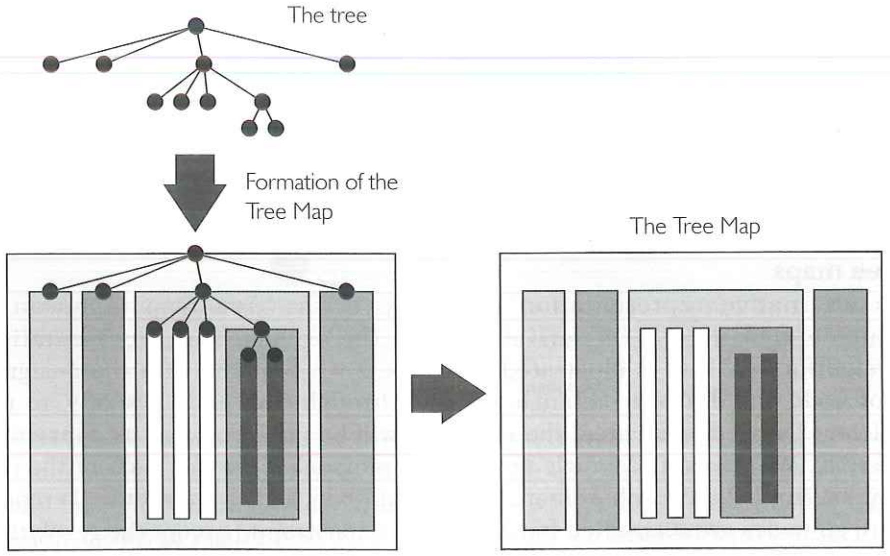
RE 3.90
rientation
e cone
more
enient for
extual
ing of
s



the display) on the depth of the tree and no requirement that all leaf nodes are at the same level or that the fan-out of every node is the same. Once the tree map is derived, colour coding, for example, can be employed to characterize different parts of the tree according to the use to which it will be put.

The main disadvantage of the tree map construction illustrated in Figure 3.91, but one which is easily overcome, is that for typical trees the result is a large number of very thin rectangles in which it is difficult, for example, to display text. The simple solution, illustrated in Figure 3.92, is a 'slice-and-dice' approach in which the generated rectangles are drawn, alternately, vertically and horizontally from the nodes at successive levels. A typical result is shown in Figure 3.93 which is an author's collection of reports. An advantage of a tree map is that, as in Figure 3.93, it supports an awareness of leaf nodes: a disadvantage is that the hierarchy is not easy to discern.

URE 3.91
truction of
e Map



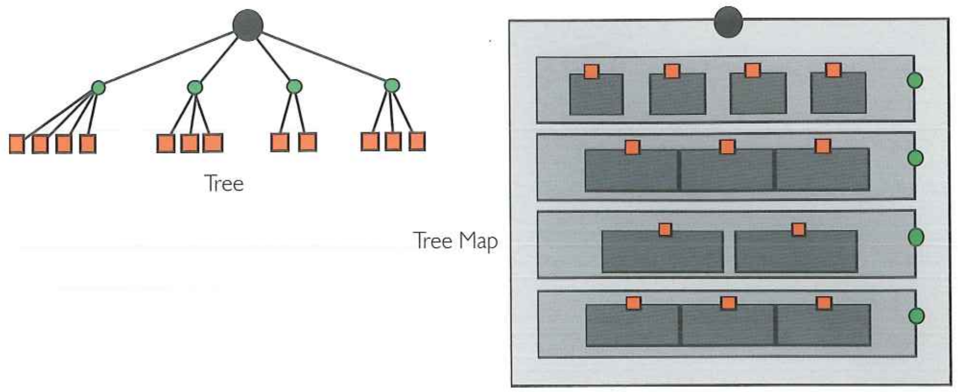


FIGURE 3.92
The 'slice-and-dice' construction of a Tree Map to obtain leaf nodes represented by rectangles more suited to the inclusion of text and images

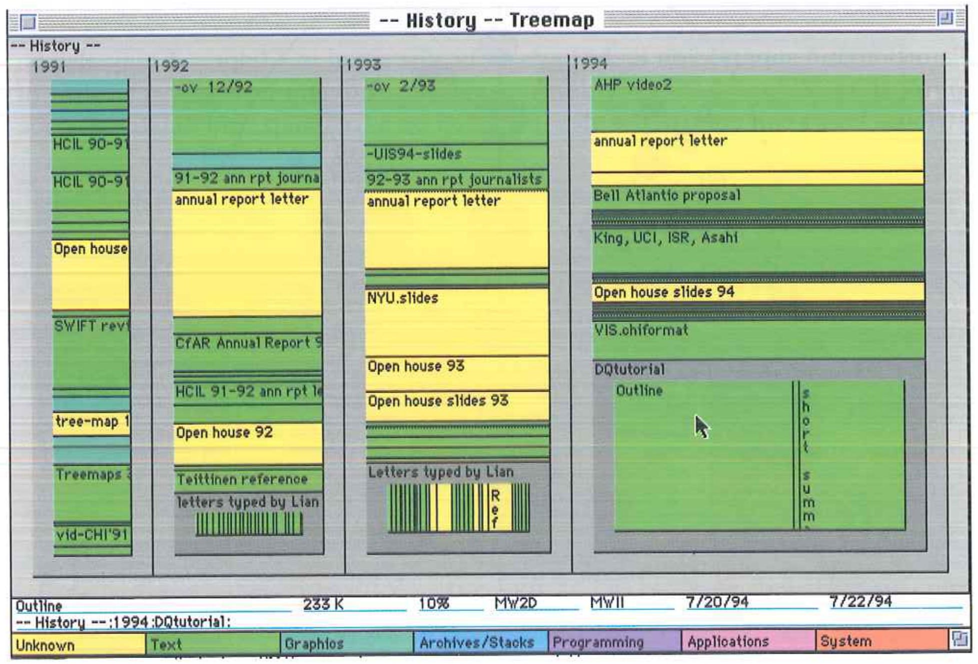


FIGURE 3.93
Tree map display of an author's collection of reports
(Courtesy of Ben Shneiderman)

The tree map technique has found a variety of applications. One is the representation to be found on the website Smartmoney.com (Figure 3.94). Major industrial sectors (e.g. energy, healthcare) are each represented by an area which, in turn, contains other areas representing relevant companies. As illustrated, there are many opportunities for encoding by colour and area, as well as for interaction by mouse-over and the selection of further detail by mouse click.