Pragmatic Subjects and Icon search

As the number of icons on an interface increases, and the range of functions that have to be represented proliferates, there is a tendency to design the icon to 'represent' the function in an almost pictorial way. This has a clear advantage when the icon is presented to users on its own, because it is easy for them to 'see' the relationship between the icon and its function (Figure 4.1). What is not so clear-cut is the effect upon the icons 'findability'.

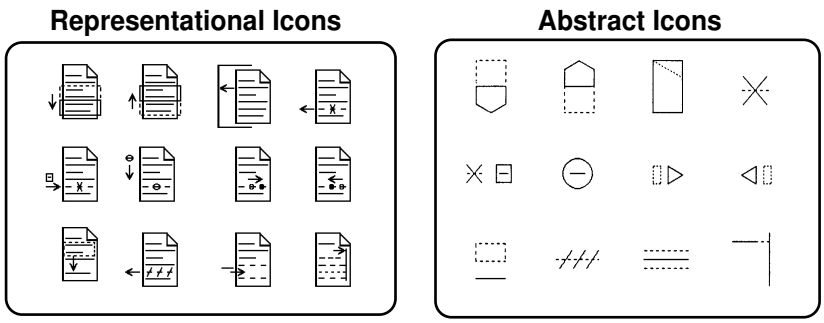


Figure 4.1: the representational and abstract sets of word-processing icons

Figure 4.1 shows representational and abstract icons that have both been used to stand for the same set of word-processing commands. The representational icons all look like pages of a document, with lines of text and arrows or boxes indicating the result of their function. The abstract icons are much simpler, and although they too provide some sort of semantic link between their appearance and their function, you really have to know what the possible functions are to work out what each icon might do. This sort of information, of course, is represented at a propositional level.

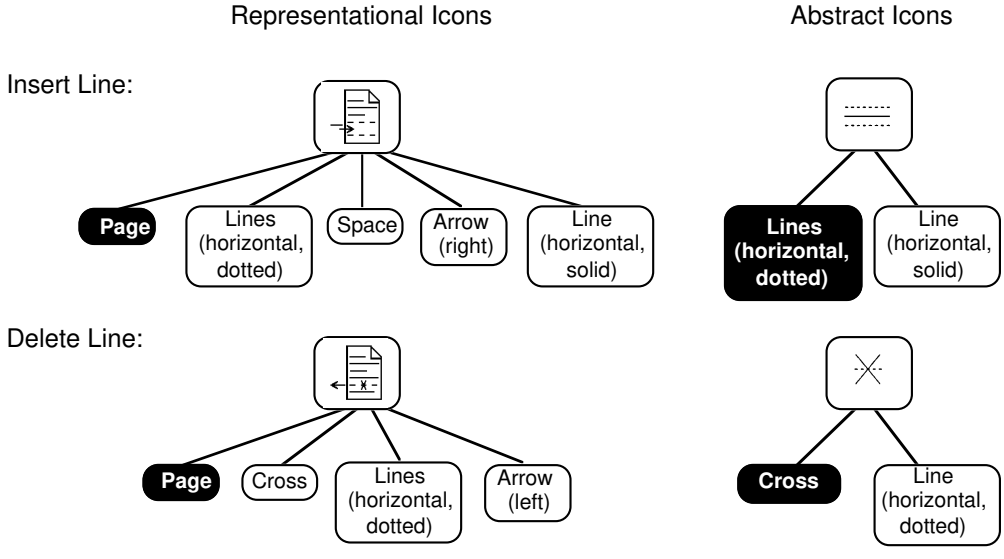


Figure 4.2: structure diagrams of two representational and abstract icons

In experiments where the position of the icons in the array varied, people using the representational set took longer to find the one they wanted than did people using the abstract set. If the icons were kept in the same position from trial to trial, so that users could remember the rough location within the array of each icon, and could 'look' straight for them without searching, the differences between the icon sets narrowed markedly.

The structure diagrams in Figure 4.2 show four of the icons from Figure 4.1: the two from each set that represented 'insert line' and 'delete line'. The icons from the representational set clearly have a more detailed structure than the icons from the abstract set, but they also have the same pragmatic subject as each other. To tell them apart, the user has to attend to their predicate as well as to the subject.

When the time that it took people to find each icon was compared with its internal structure, a clear relationship was found (Figure 4.3). The greater the degree of similarity that the icon's structure had to other icons in its set, the longer it took to find an icon. This suggests that users use their propositional-to-object transformation to access their knowledge about the icon, or about its meaning. This provides them with an object level mental image of the 'target' icon that they are looking for, and they can then compare 'candidate' icons from the array with this internal mental image to see if they match.

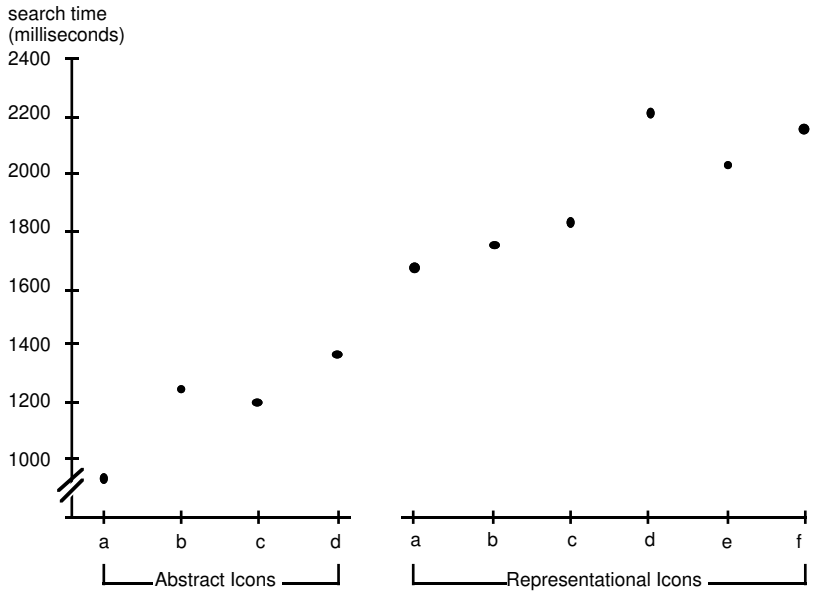


Figure 4.3: the more complex the discrimination, the longer it took to find an icon.

An icon in an array would be a candidate if it had the right pragmatic subject, and it would match the target if it also had the right predicate. Icons that needed more objects of their predicate to be evaluated to be discriminated from other potential candidate icons would take longer, overall, to locate.

People can search for a target rapidly by looking for objects that have its pragmatic subject

This helps explain why the representational icons took longer to find – it wasn't because they were representational, but because they were all so similar – and even within the sets, it was possible to show that the more complex the discrimination, and the more candidates that shared its pragmatic subject, the longer it took to locate an icon. In the abstract icons the pragmatic subjects are mainly different, which means that the search can be carried out at the level of the icon, using the more salient information. You might remember a similar effect of the pragmatic subject from the very first figures in this guide (Figures 1.1 and 1.2). Figure 4.4 again shows an array where all the icons in the window have different pragmatic subjects.

These icons form a group of icons in an array, but their visual structures do not lead the user to see them as forming any subgroups. When an icon is searched for in this array, the icons can be discriminated from one another by their pragmatic subjects, without their structure needing to be evaluated. As you look from icon to icon in this array, you make the visual transitions represented in Figure 4.5.

None of the icons in this array have any real advantage over each other: if you 'know' what icon you are searching for, and can form an 'object' image of the target, then you can probably locate it quite rapidly. Try finding the icon in Figure 4.4 that looks like a dog sitting next to a Macintosh computer.

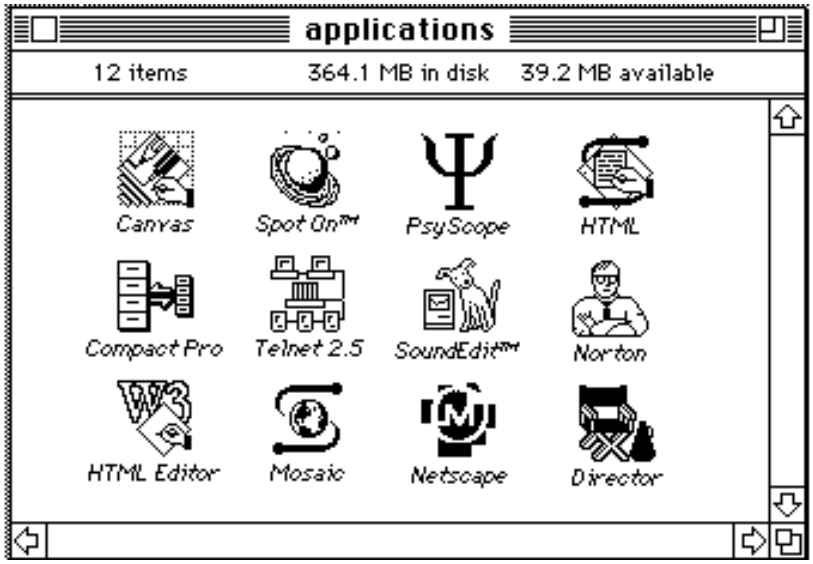


Figure 4.4: icons with different psychological subjects, in a Macintosh Window

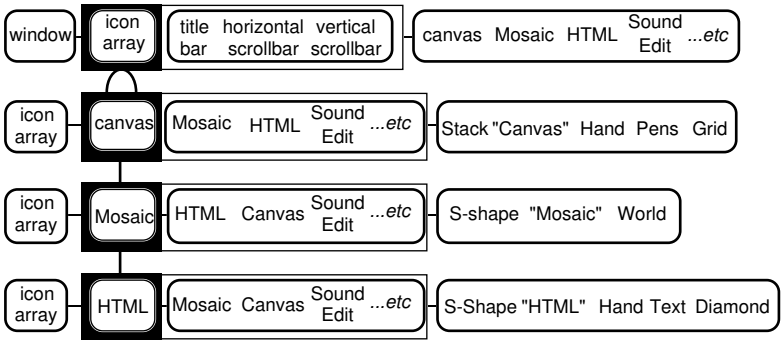


Figure 4.5: a transition path diagram showing the search for an icon in Figure 4.4

In contrast, try finding the icon in Figure 4.6 that contains a picture of a Macintosh computer. Now the subjects of many of the icons are similar, and you have to evaluate more information about each icon, as with the representational icons of Figure 4.1. The corresponding transitions are shown in Figure 4.7.

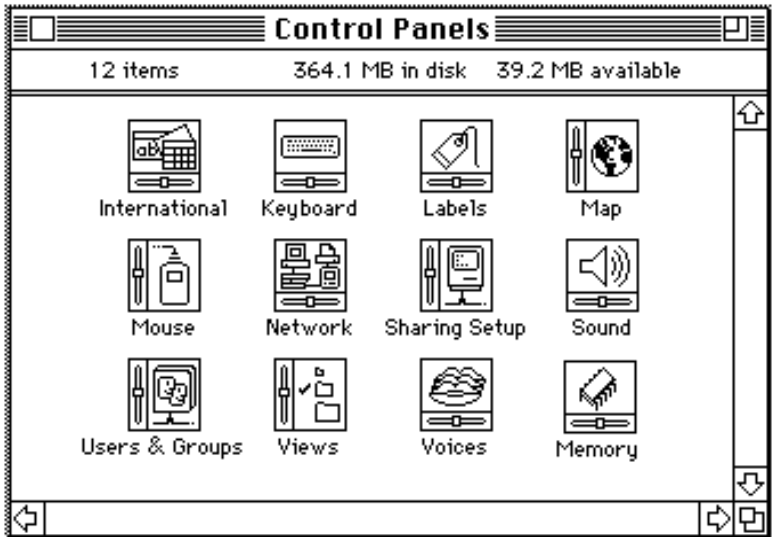


Figure 4.6: different icons that share subjects

The pragmatic subject of each icon is the 'slider box' that surrounds each icon's contents, and so it is not possible to ignore them and locate the Macintosh directly. Each time an icon is attended to, a transition must be made away from the pragmatic subject to examine the rest of the icon's contents. Again, none of the icons have much of an advantage over each other, but this array is harder to search than the one shown in Figure 4.4, because more transitions are required to search through its structure. In Figure 4.7, three icons are searched before the correct one is found – with 12 icons, the average number of icons that would be evaluated in this way would be 6.5!

As with the circles of different colour, and the lines of different orientation, when one

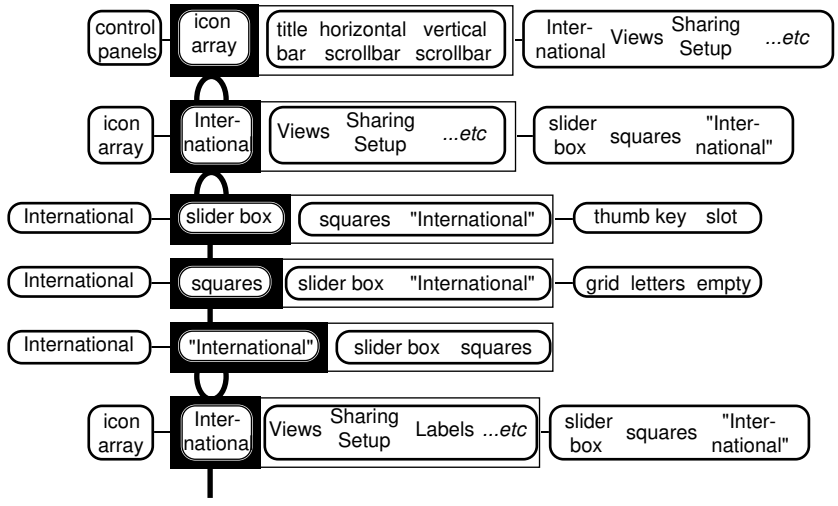


Figure 4.7: the transitions required to locate an icon in the 'Control Panels' array of Figure 4.6

icon in an array has a completely different pragmatic subject to the others, pop-out happens, and that particular icon is very easy to find - this is shown in Figure 4.8. The icon that does not belong to the group of 'Word' document icons pops out from the array. Even though the Word icons are not all identical, and have different text labels, they are difficult to search through. The Word icon that is 'different' to the others still has the same subject (the document shape) and also shares a predicate object (the large 'W') and so is nothing like as easy to locate as the 'ψ' icon.

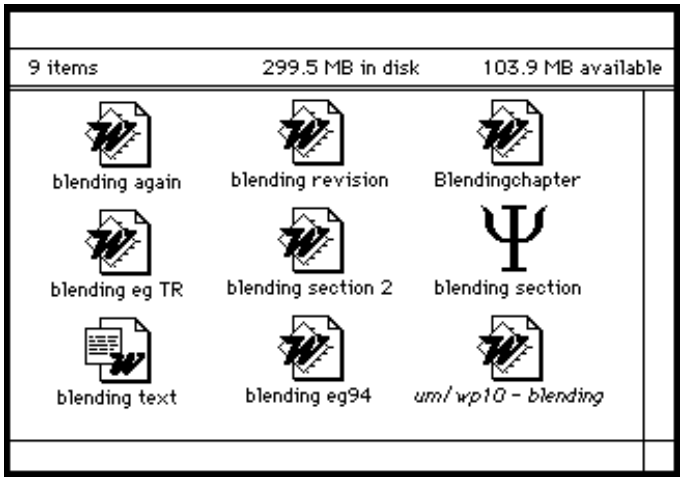


Figure 4.8: pop-out in an array of icons

To summarise, if you are designing an array of icons that people will have to search frequently, it is sensible to give them different pragmatic subjects, rather than different predicates. However, having different pragmatic subjects will make the icons 'look' different, and so the array itself might be harder to pick out.

An object's structure affects grouping

The examples of pop-out we have seen so far have shown that changing one attribute of an object can affect the structure of the scene, by determining which other objects it will or will not form a group with. As well as changes to attributes, changes to an object's own internal structure can also affect grouping: structure affects structure!

The arrays (a) and (b) in Figure 4.9 contain two types of object. One is a simple circle, the other is an incomplete circle – a small part of the circumference is missing. When the incomplete circle is placed amongst an array of complete circles (a), it is easy to see the incomplete circle. The opposite is not true – in array (b) it is much harder to locate the complete circle amongst a number of incomplete circles.

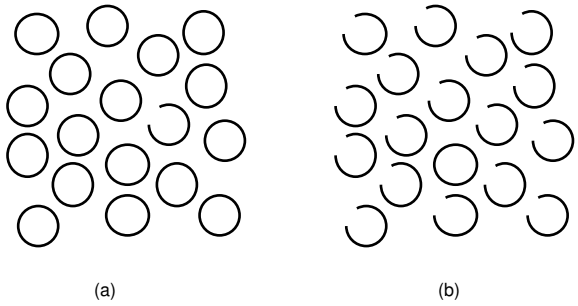


Figure 4.9: asymmetry of pop-out due to structural differences

When we draw structure diagrams for these two sets of circles, we have to show the incomplete circle in set (a) forming a pragmatic subject, to make it clear that it pops-out. For set (b), we have to show the complete circle as part of the same group as the incomplete circles, because it doesn't pop-out. Figure 4.10 shows how these groups are composed (not all of the circles are shown). In the Figure we have described the incomplete circles as 'circles plus gaps' – in effect, we are saying that they actually have two components to their structure, while the complete circles are just circles, and have no further structure.

When most of the objects within a group are simple, and do not have a structure, a similar object that does have a structure cannot form a group with them at the same level. The simple objects form a subgroup, and the complex object becomes a pragmatic subject, as in Figure 4.9(a). In contrast, when most of the objects within a group are complex, with a common pragmatic subject and a structure, simple objects that consist of the same pragmatic subject but nothing else, are able to join the group: as in Figure 4.9(b), they are simply perceived as similar to the other objects, but less complex. They are able to 'hide' amongst the noise of the other objects' complexity.

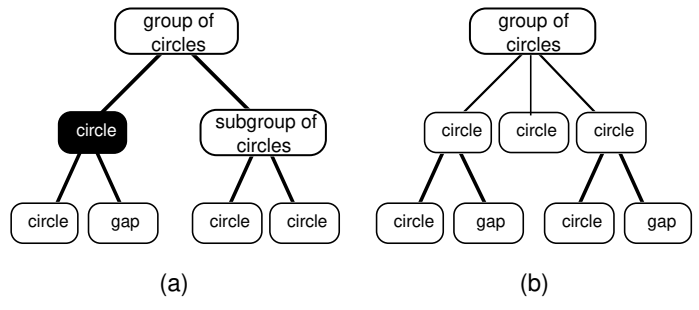


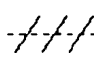
Figure 4.10: complexity of structure affects pop-out

Complex objects that do have a structure pop out from simple objects that do not have a structure.

Simple objects can 'hide' amongst more complex ones, if they have the same pragmatic subject.

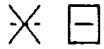
Exercise Set 4

1. Draw structure diagrams for each of these abstract icons (ignore the words):



delete line

word search



replace word

scroll right

2. Draw a structure diagram for the array of icons in Figure 4.8 (ignore the text labels and the 'frame' of the window), with the 'document shape' as the pragmatic subject of the icons that have one.
3. Draw another structure diagram for the icons in Figure 4.8, this time assuming that the 'W" of each icon was its pragmatic subject.
4. On the basis of the two diagrams you have drawn for Figure 4.8, would it be easier to find the 'blending text' icon if it had the 'document' or the 'W' as its pragmatic subject?