and there is hardly a design studio where computers have not replaced at least some of the drawing boards. This is not a book about computer-aided design any more than it is a book about drawing. For these reasons it no longer seems appropriate to continue to devote a special chapter here to what is a major subject in its own right. We are however interested here in how designers interact with computers as part of a design process. There are several questions here. Those questions are not so much about what computers can do as what they cannot do. They are not so much about what happens inside the computer but how we converse with it.

Amongst the most fundamental questions we can ask here are: what knowledge do designers exchange with computers, for what reasons and how? They are also really beyond the scope of this book as I have discussed them more thoroughly in *What Designers Know* (Lawson 2004). However a brief discussion of how we converse with computers is useful in the context of seeing design as conversation. In fact much of what is called computer-aided design is in reality computer-aided drawing. Even this does not interest us here as this kind of drawing is most often for presentational purposes rather than as part of the design process itself.

Computers so far cannot design in anything like the sense that we use the verb in this book. They may be able to solve well-constrained problems, but they cannot design in any of the fields we are discussing here. So if computers appear in the design studio, other than as rather smart drawing boards, their purpose must be to aid design. If this is the case then we must assume that the greatest responsibility and certainly the final say will rest with the human designer. Again logically this tells us that the human designer will necessarily be in a conversational relationship with the computer. In fact the designer is going to have to describe the design state and then interpret some modification of it as suggested by the computer.

In general, designers seem to find this experience of using computers a frustrating one. Many well-known and successful designers have articulated their opposition to using computers in their design process. Santiago Calatrava, although using computers for structural design packages such as finite element modelling, prefers to use real physical models to computer-based ones (Lawson 1994). Others rely on computers but leave specialist staff to interact with them. The amazing work of Frank Gehry relies heavily on a great deal of computer technology for its realisation but Gehry himself prefers not even to see the screens of the computers (Lindsey

2001). Gehry is thus lucky to be able to have conversations with the members of his staff led by Jim Glymph who look after all the technology and effectively hide it from him.

Of course the computer can save designers huge amounts of time in the way my computer did for me when I was writing this book. I well remember that the first book I ever wrote had to be done on an old fashioned typewriter. It was a painfully slow process that invited no reflection or interaction. There was no easy way to make simple changes, you just had to type it all again. So of course the editing and interacting capability of computers helps designers to make images. But even here designers often describe it as rather a remote process. As Nigel Cross rather disappointedly asks (Cross 2001b):

> Why isn't using a CAD system a more enjoyable, and perhaps, also more intellectually demanding experience than it has turned out to be?

So what is the problem here? The answer to this simple question is actually rather complex and much of it beyond the scope of this book and certainly this chapter. I attempt some of the answers in *What Designers Know*. Here we should continue to concentrate on this conversational view of design. A real problem with much computer software in general and much CAD software in particular is the way in which the conversation has to be on the computer's terms rather than the human designer's terms. There are several reasons for this. Often the capabilities of the software to perform a multitude of clever tricks, most of which most users will never even bother with, means that the whole system becomes extremely complex to understand. Again my word processing software offers a good example. I have been writing with this system for many decades now but I have never read the manual or gone on any training courses because I am just too busy. As a result I am aware that there are many menu commands and features that I do not use. I can even see that some of them might be useful but only on rare occasions. I know that the opportunities to exploit these features will be so few and far between that even if I learn them I will have forgotten them by the time the next chance to make use of them arrives. So it is with computer-aided design systems but even more dramatically so.

CAD systems suffer from a much worse problem compared with word processors. Putting the text into a word processor is generally an obvious and straightforward task that does not require attention and therefore does not distract me from thinking about what I am