

One Last Compile...

I know exactly what I want...

Having in the past blamed the management, Delphi, myself and my co-workers for various failings in my software, I'd now like to point the finger at somebody else. I'm fed up with our clients. The ideal client is one who walks in, likes the product you're selling, pays for it immediately (preferably in cash) and who, having installed it, never seems to have any mysterious problems. I have yet to come across an ideal client.

In my experience, clients who don't know what they want are easier to deal with than clients who do. Clients who plead ignorance can usually be kept happy with a splash screen and a file menu until well into the final phase of a project. Clients who say they know what they want are either lying or mistaken, although they will never admit to this. Programming fact of life: clients only know what they want when they've had what they don't want lying around unused for a couple of months.

When you first meet a client your natural instinct is to try and sell them something you've written for somebody else. This seldom works. If you're lucky, they'll like maybe 30% or 40% of what you've done. A goodly chunk they will simply dismiss as *we wouldn't use it*. Another sizeable set of features will be waved away as *too complicated*. And, when it comes to the stuff they do want to use, they invariably want to make the software work in a different way. You agree to all their demands, of course, partly because you want their money, but mainly, and you know this is true, because what you really want is to impress them with how clever you are. I've lost count of the number of times I've returned numb from a meeting with a client with a list of changes the size of the Domesday Book, all of them my fault because I can't bring myself to give an honest answer to questions prefixed with *How difficult would it be to...?* The honest answer is always: *more trouble than it's worth, particularly for the lousy sum you're paying me, Buster*, but somehow that always comes out as *Oh, dead easy*, which really means, *Oh, dead easy because you're in the presence of a software genius*. Frankly, it serves us right that we end up supporting eighteen different variants of the same package.

On no account should you be tempted to start *telling* them what they want, because that way lies madness. On every project there's a tendency to believe that the client's company can be reduced to a set of easily understandable, mutually consistent, business rules, which can be easily mapped onto a computer system. 'No special cases!' is the rallying cry. That's until you spend half an hour with the Finance Director as he explains each of his 13 different ways of producing a profit and loss account, depending on who it's for and what's on special offer at the canteen, and you glumly emerge with more special cases than Dr Kildare.

It can be quite depressing how the initial enthusiasm for a project can tail off. These things often start with a demonstration of the 'old system', and you'll stand around laughing cruelly at the pathetic monochrome DOS screen and its buggy reporting module. Your client is convinced that the right software can solve all his organisational and business problems, or at the very least produce reports that will him make him look cool in front of his customers.

You're thinking that here, at last, is a nice no-brainer database project that you can bash out in a couple of weeks. But, as you stand around that poor doomed DOS system, you may well feel the clammy hands of Doom gently massaging your spine, whispering in your ear and... [*Oh, get on with it. Editor.*] In five years time it'll be your system they will all be sneering at. In five years time the client still won't be able to get a decent colour graph out of his system. But in five years time, you can be sure he'll be telling his new software house that they needn't worry, because this time he knows exactly what he wants...