One Last Compile...

Databases: the painful truth

If only all my customers were like that dive shop that features so heavily in DBDEMOS. They seem undemanding, sensible folk. They seem happy with a simple inventory of stock, and a nice set of linked grids showing their customers, invoices, and the odd fish. When you work through the example applications that Delphi comes with, or browse through an introductory Delphi book, you can see that building database applications from now on is going to be a breeze, a doddle, a no-brainer.

To which the correct response, of course, is a derisive snort and a loud 'Ha!' Here are the chapter headings from my forthcoming volume on *Databases and Delphi: Painful Stuff that you should know now to save tears of frustration later*.

Chapter 1: *Just because they're neat components doesn't mean they're the right components*. In this chapter we'll explore why it is that programmers (ie me) become strangely blinkered the moment they start work on a database application. There's so much there already done for me that I become conservative, reluctant to go off down previously untrodden paths. Tables, Queries, Grids and Data-Aware controls offer such a seductive way of cobbling together an interface that I invariably structure my program entirely in terms of how they can fit together. When I come across something which can't be easily represented using these components, I shy away from it, and tell people it can't be done. I suspect this makes me a bad person.

Chapter 2: *The BDE is your master, not your servant*. It would be nice to pretend otherwise, but let's face it: the Borland Database Engine calls the shots. Admittedly, if you're running Windows for Workgroups then you might periodically lose all your updates, but hey, you didn't become a programmer because you thought life was going to be easy, did you? A large proportion of your time must be devoted to keeping the BDE happy: `dbiUseIdleTime`, `LocalShare = False`, and *Multiple Net Files in use* must all be phrases that hold no mystery for you.

Chapter 3: *Write a 16-bit Database app and wave goodbye to your Windows handles*. Nothing will consume up your precious supply of Windows handles faster than a database application. For morbid entertainment value, write a function which returns `GetFreeFileHandles` periodically and monitor the situation as you run your program and open various windows. It transforms a simple demo into a roller coast ride of emotion.

Chapter 4: *Everybody says use TQueries, but you'll be seduced by TTables anyway*. Use `TQueries`, runs the conventional wisdom, and transporting your application to a beefy client/server system later will be a doddle. This might well be true, but it also means that your original 16-bit Paradox application will be a dog of cosmic proportions. `TTables` are more convenient, more flexible, and, as far as I can tell, faster than `TQueries`. You also don't have the hassle of writing SQL, which leads us neatly into...

Chapter 5: *16-bit SQL sucks like a Hoover*. You'd think, he said peevishly, that a product which features as its logo some kind of Greek goddess with the letters 'SQL' whizzing around her head would have some half-decent support for SQL built into it. At the very least, you'd expect some kind of guide to the syntax of local SQL. You get neither of these. So resign yourself to long days trying variants on the same SQL command until you finally get a syntax that works. And forget trying to do any really complex SQL because Delphi will often die horribly in the attempt.

I could go on (*Database Desktop: A great application that Borland never finished*) but I don't want you to get the impression that I'm overly bitter about all this. (I am, but it would seem uncool to admit to it.) I still love Delphi. It's just that sometimes the database side of things seems more frustrating than it should be.

Profound lesson for this month: Life is not a dive shop.