

One Last Compile...

The rock pools of Delphi

Bear with me for a moment, but I want you to imagine a shrimping net. This is a magic shrimping net, naturally. You can put what you like into it, of whatever size or form, and the net will happily welcome it into its depths. The net itself never seems to bulge, break, or grow any heavier. And whenever you want to get something back, this net will cheerfully pass its contents back up to you, neatly labelled and organised.

All of which is a roundabout way of saying that I couldn't imagine life without `StringLists`. There isn't a day goes by that I don't write some code that makes extensive use of them. Several of my applications hinge entirely upon them. Once you've got the hang of them, they're so flexible and elegant and eager to please. If I were a poet, you'd now be reading 'Ode to a `StringList`.'

The first time I used a `StringList` to manage my objects, I must admit I was extremely sceptical. Up until then I'd just been using them as glorified arrays: a useful beastie, to be sure, but not one to change your programming life. Then I was browsing through the help, and found a reference to the `AddObject` method. It sounded extremely unlikely to me. I mean, this little glorified Memo-like object, swallowing down whatever you throw at it? Pur-leaze.

So I passed it a huge `StringGrid` populated with data, and it disappeared without a murmur. I asked it for it back, and it emerged like a rabbit out of a conjuror's hat. Impressed, I passed it an entire form. Same result. What about a live data table? Not a problem. And that was when I knew that life in Delphi was going to be different from now on. It was liking driving along in a car, and suddenly spotting there was a button marked 'Hyperspace.'

Now, obviously, even I know that it's all done with pointers and that no matter how much I load up my `StringList` there is absolutely nothing in memory which remotely resembles a bulging shrimping net (but part of me can't help wincing apologetically every time I bung a particularly bloated data structure down a `StringList`'s throat). And yes, I'm aware that they can be a bit slow and inefficient, and in Delphi 1 there were grumbles about memory leaks, but frankly, these are a small price to pay for such a useful tool. The only danger is that I become a bit too dependent on them. I use them to get me out of all kinds of programming cul-de-sacs, and if Inprise ever casually announce that they're discontinuing support for them, I will be seen with a large coil of rope looking for a strong beam.

I'm deeply impressed with whoever thought up `StringLists`. For all I know, they're not unique to Delphi (indeed, for all I know, many of you reading have long since moved on in favour of housing your entire application's data structure on a little known property of the `TLabel` component, but please do write and tell me if you are). What possessed that person, perhaps sitting slumped in the Inprise canteen one day, to muse that it would be really neat if alongside each modest string in a flexible array-type structure, you could also provide a home for just about any kind of junk the user cared to put there? Whoever it was, in whatever language, I salute you. The rock pools of Delphi are a far more productive place thanks to my magic shrimping net.