

# Delphi for .NET Developers Guide book by Xavier Pacheco (pub Sams)

a subjective review by Jack Birrell

There are very few books so far, on this precise topic. Other reviews of this book have been published so I am approaching this review from a very personal angle.

I am an experienced (ageing?!) Delphi Developer whose greatest achievement might have been devising a boot-strap routine that was half the size of the manufacturer's. At this stage, I don't know if .NET is a good thing. I know nothing about .NET, so I am looking to this book to tell me not only all about it, but also whether it is something that I want to get into. I expect to use this book rather than take courses to help me get started. I've had all previous versions of Delphi so I believe that I will find upgrading to Delphi 8 will be no harder than any of the other versions. The reader may scoff!

Technical books these days come from one of a group of publishers, each having a common house style. This is the first SAMS book I have seen, with 861 pages of which 71 are index. Within the index, alternate pages ask the reader to mail the publisher, suggesting ways to make the index more useful. I found the layout and typeface easy to read but wanted to see links to unfamiliar TLAs introduced within the text. Otherwise it seemed very thorough. The whole book though is heavy and I found just handling it (I tend to read these things in bed!) somewhat awkward. In fact, it dropped on my face a couple of times.

Vast quantities of example code appear within the book. Below each is a reference to where the code can be found on the accompanying CD. However, some chapters have over a dozen code files on the CD and it's not initially clear as to which file is which, causing some frustration. Couldn't the book actually show the file name? Some files bear the prefix D8DG, (I presume) Delphi 8 Developer's Guide but why the inconsistency?

The CD also contains the text of a previous book by the same author as well as samples and demonstrations of other software. This is common practice but I am wary of taking sample software off a book CD as it's often out of date. I haven't looked at these yet, however it is often useful to look at other people's code, when supplied.

The book is in five sections: [ See Table of Contents in our July/August issue. Ed]

- Introduction
- Programming
- Language
- Class Library Development
- Database Development with ADO.NET
- Internet Development with ASP.NET

The Introduction did not answer any of my questions. I am left cynical that the invention of .NET is an attempt to introduce a higher "common standard of software development", and presumably a means to increase employment within our industry. I used to joke that it was invented because Microsoft could not make Java work. Now I feel that it is a ploy to force the world in a single proprietary direction. While the Framework is said to be multi-platform, (and there is a chapter on the Mono project) I've seen little evidence of its use anywhere than under Windows. I've been in this business long enough to have been through many initiations of universal languages. With this one, consider how much machine power and resources will be consumed by the garbage collector alone and you'll realise that complete applications (including operating systems) a few years ago would use much less.

The author quotes that ".NET is the Microsoft solution for Web Services, the next generation of software that connects our world of information, devices and people in a unified, personalized way." The author mentions confusion over what .NET actually was, when it was announced. I am left concluding that .NET is a solution for problems which have already been solved, much more simply and elegantly. Of course, I could be wrong?

Back to the book - it explains clearly that we don't have *units* anymore as they are called *namespaces* now. The principal namespaces are listed along with their basic functions. Programs are now called *assemblies*. There are many new and lengthy words, needless to say. The language is explained well and some of it seems very different from the Delphi I know. I was surprised to read that Delphi is also known as Object Pascal as I recall the latter being dropped, was it in Delphi 7?

Many of the code listings contain comments in XML format, for which I've so far found no reason. It does take up loads of space, like this:

```
/// <summary>
    /// Required designer variable.
    /// </summary>
```

Many examples also contain the full code of a module (including interface section) when surely just the procedure being described would suffice? Having read through such lengthy listings, I was surprised to see a note on a following page saying words to the effect that the preceding code would not work if a specific file or directory was not present. A comment within the code at the right place would surely serve better. If I hadn't turned the page, I could have struggled for days, trying to get the code to work.

There are pages also, listing tables of methods, properties and events which I found rather superfluous. Surely the best place to read about these would be within a product's online help? I'd certainly like to see a few mentioned with some explanation. Within the section on string formatting, there is a table listing every one of Date and Time Format Specifiers, then a forty line code listing showing use of them (where each line consists of:

```
Console.WriteLine(System.String.Format(
```

and then there follows a table listing the output from each code line. Had the outputs been included next to the relevant code line as a comment (the normal way within Delphi online Help), it would have been much easier to follow.

As for typographical errors, I wonder how many newcomers will understand what is wrong with the Concatenate function when they spell it *Contact*?

I'm not quite sure why several chapters are from different authors. I'm not complaining. I found those by Steve Teixeira and Brian Long to be particularly clear, readable and useful.

I have not yet finished reading this book. It is a lot to take in. To fully understand the entire topic, I reckon will take at least one further reading of the whole book. There is a huge wealth of information here and, while the book annoyed me, it does form a useful source of information about Delphi for .NET, which it sets out to do. My deeper questions about .NET itself will hopefully be answered when and if I ever read a Microsoft version.

*Jack Birrell is a seasoned <G> developer, based in Surrey. You can contact him on [jackbirrell@btconnect.com](mailto:jackbirrell@btconnect.com).*