

# Preview: 32-Bit Delphi

by Chris Frizelle

Since before the roll-out of the 16-bit version of Delphi, the folks at Borland have been working hard on a 32-bit version targeted at Windows 95 and Windows NT. At last we can give you a preview of what you can expect in the new version when it ships.

The current official word on the ship date, by the way, is “approximately 90 days of the commercial availability of Windows 95”. We all hope it will be nearer the middle of that 90-day period than the end, of course, but how much it matters is a controversial point! (How many large companies do you know who plan to move all their users to Win95 immediately?).

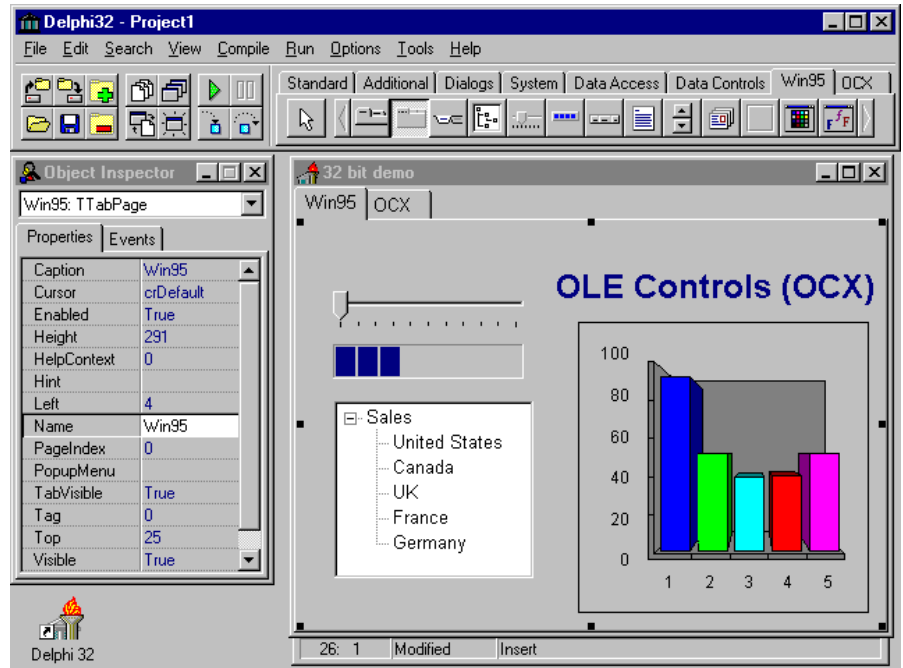
## The Old And The New

A basic 32-bit version of Delphi has been running internally at Borland for some considerable time. This was helped by writing the 16-bit version with the 32-bit version in mind – unlike some older software, there’s been no need to go right back to first principles and re-think things from scratch. What Anders and the others have been concentrating on over the months is adding new functionality, as well as listening to feedback from users of course.

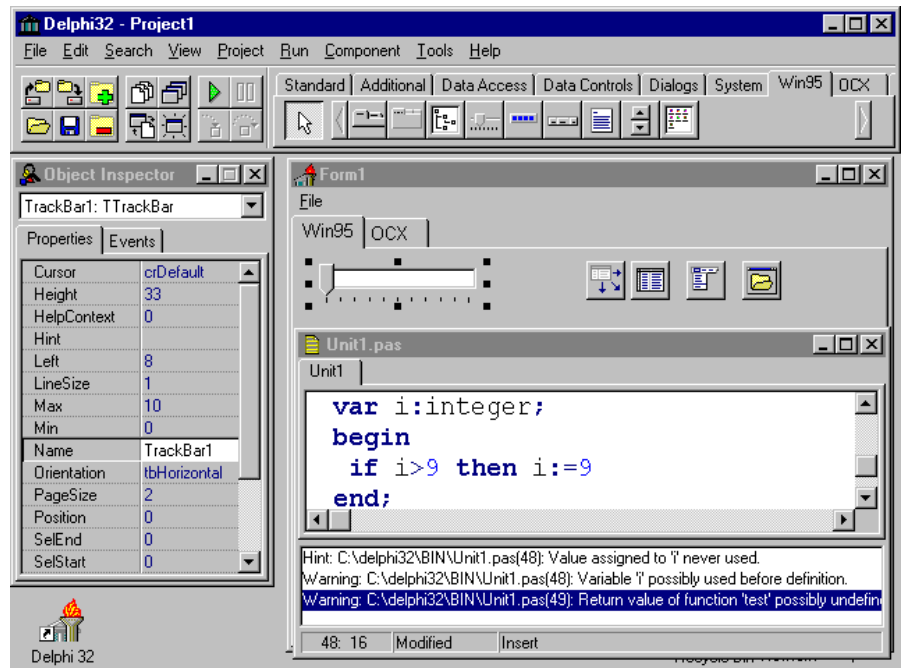
As you can see from the screenshots, visually things are pretty much the same, except the interface is a Windows 95 one. This means that the component palette and object inspector have the tabs on the top, not the bottom, and so on. You’re going to feel pretty much at home with this new version.

## First, The Pretty Bits

Apparently, the development team decided to do a full Win95 implementation, rather than a quick “yes it produces Win95 apps but don’t expect anything fancy” job. There will be complete support for the Win95 interface, with a set of



➤ Figure 1 Full support for Win95 and OCX controls



➤ Figure 2 Improved error messages and diagnostics

specific Win95 controls (see the component palette in Figure 1).

Whilst your 16-bit Delphi applications will indeed recompile under the new version and produce a Win95 application (providing you didn’t do anything

naughty like write masses of 16-bit assembler...), they won’t take advantage of the Win95 controls unless you specifically amend them. Hopefully Borland or someone else might come up with a clever utility to automatically

convert projects using Windows 3.x style controls to their Win95 equivalents...

As the world is moving away from VBX controls to OLE custom controls (OCXs to you and I), the new Delphi has full support for them and will include a selection in the standard installation (see the cover screenshot; what will be in the shipping product by way of add-ons is as yet a moveable feast). Native Delphi controls will of course remain the best choice in terms of power and flexibility.

As befits a full programming language, there's full unfettered access to the Win95 API, with the attendant goodies like multi-threading available for use (there's scope for some how-to articles on this in the future, methinks - any volunteers?). OLE automation is in there too.

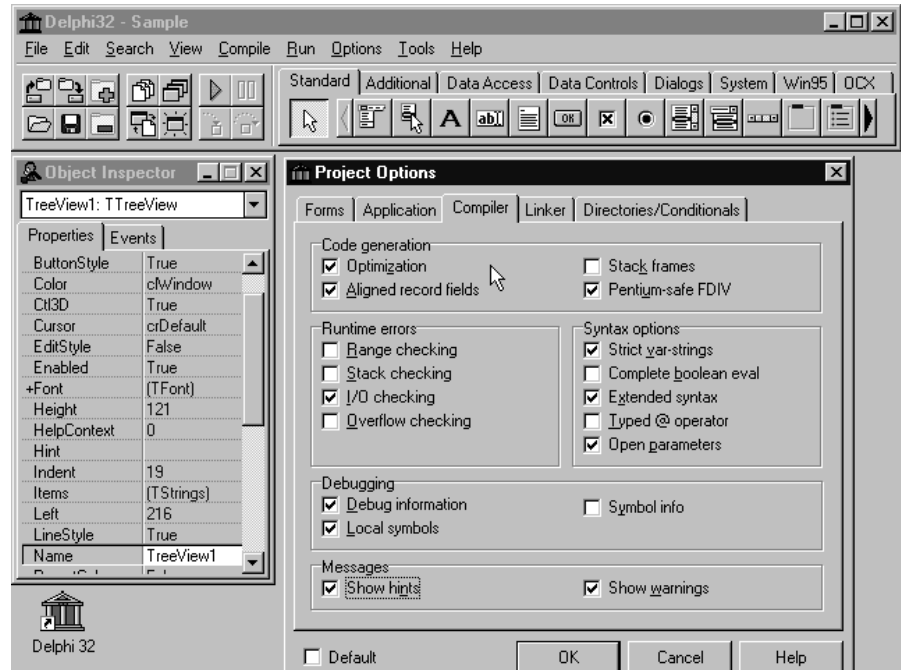
Applications developed with 32-bit Delphi will, we are assured, be able to meet the requirements to sport the Win95 logo. That'll keep your marketing department happy...

### The Power Inside

There's a lot more to 32-bit Delphi than meets the eye, however. The compiler features a common back-end with Borland's C++ product, with improved performance and new optimisation features.

Preliminary benchmark figures from Borland indicate dramatic speed improvements for applications: typically 3 times the performance of 16-bit Delphi for standard Sieve and Whetstone tests - OK these are not real-life applications, but they give an idea of the gains to be had and certainly leave VB and (interpreted) PowerBuilder way behind coughing and spluttering in the dust.

This power comes courtesy of a number of automatic optimisation features, which all sound very clever. The key thing is that they do their job, which they seem to from the benchmarks, and that it doesn't take a PhD in computer science to be able to use them. Looking at Figure 3, I reckon even I could work out how to check the "Optimization" box! You will also



➤ Figure 3 Compiler optimisations

notice the "Pentium-safe FDIV" check box, which allows your application to produce code which will run correctly even on the "flawed" Pentium chips.

The common back-end means that Delphi can use standard .OBJ files, which will help when linking in legacy C code and getting access to third party libraries. Also added are better error messages and 'hints' to help you understand why your new killer app wouldn't compile (Figure 2). As you can see, Delphi can now find *all* the errors in one pass, rather than just stopping at the first one.

If your background is C/C++ or VB, then you'll be glad to know that there are smarter warnings to help find those darned forgotten or superfluous semicolons which you so often curse!

Those who have been with Borland's Pascal-based products for a while have grown used to smart linking and blazingly fast compile times. You'll be pleased to know that you keep both these advantages and indeed they are built on in the new version. The new 32-bit smart linker eliminates unused functions, procedures and methods (static and virtual).

A 32-bit compiler and a 32-bit operating system will mean access to a "flat" memory model. No more

limitation of data structures to 64Kb, say goodbye to the 16-bit segmented architecture.

### Borland Database Engine

User feedback on the Borland Database Engine (BDE) seems to have had some effect, as a new 32-bit version will be included, with improved performance and a scalable query engine.

I for one hope that there will be more improvements on the database side before the release - the wealth of 3rd party commercial and shareware database-related products for Delphi which have come out seem to indicate there is a need here. We also have no information at present on any improvements or additions to the VCL (Win95 controls aside).

### Conclusion

The basis of 32-bit Delphi certainly seems sound and shows great promise. I'm sure that there will be more to hear over the next few months on less fundamental matters: a couple of extra swings here, a nice bell or three there.

We'll keep you posted as things develop and of course we're as itchy as you are to see the final availability, pricing and upgrade information announced. Don't forget to watch this space...