

---

---

# Introducing Borland Kylix

---

---

*What Microsoft did for Windows development with Visual Basic, Borland is hoping to do for Linux with Kylix.*

By Julian Moss  
Developer and Journalist

**L**inux took a step nearer becoming a mainstream platform for corporate use when Borland released Kylix, a tool for developing enterprise applications on the free operating system. Based on the company's popular product Delphi, Kylix allows developers to create database and Web server applications, as well as desktop GUI clients, all of which run as native object code binaries under Linux. Kylix's compatibility with Delphi will make it easy to develop cross-platform applications and to port Windows applications to the Linux platform. We take a look at Kylix in more detail, to help you decide if it has a role to play in your company's future development strategy.

## ***RADical Development***

If you've ever seen one of the popular Windows rapid application development (RAD) tools you'll be instantly familiar with Kylix. The development environment looks exactly like those of Borland's Delphi, C++Builder and JBuilder products, which in turn were closely modelled on Microsoft's pioneering visual development tool Visual Basic. The main window sits at the top of the screen and contains the usual menus and toolbars for loading and saving projects, running and debugging programs and so on, plus a tabbed palette of visual components that you drop onto a form to add functionality to your application. The form designer, code editor, project manager and other elements of the environment operate as free-floating windows, some of which hide themselves when not required to reduce screen clutter.

The editor, which for Linux offers the option of Emacs key bindings, boasts code completion (a tool that presents a list of choices to complete a line of code based on what you've already typed) and a browser that shows the object hierarchy as a tree structure and permits rapid navigation within a project. The debugger is integrated with the editor and has features like tool-tips that show the current contents of a program variable. This is all old hat to Windows developers, but it will be a revelation for many programmers on Unix and Linux platforms where visual development tools, such as there are, have lagged a few years behind the Windows state of the art.

## ***Pascal***

The programming language behind Kylix is Pascal. This again is a revolutionary step for Linux, under which application development has until now been almost exclusively C (and to a lesser extent C++) based. There's a good reason for this, though. Over the years Borland has developed its Object Pascal dialect (which differs quite considerably from "standard" ISO Pascal) to make it into a first-class object-oriented programming language. It's effectively a proprietary language, but no more so than Microsoft's dialect of Basic or any of the 4GLs that corporate developers often use.

Pascal lends itself to rapid conversion into object code using a one-pass compiler. Kylix's compiler - with a lineage stretching right back to Borland's first product, Turbo Pascal, in 1983 - is lightning-fast, so that it takes typically just a second or two between clicking the Run button and seeing the latest version of your program pop up on screen ready for testing. Such fast compilation isn't possible with C++ or most other programming languages, enabling the "Rapid" element of Kylix's RAD tag to be more than just copywriter's hyperbole.

The programming language of choice is a religious issue for many developers, and Pascal has always had fewer adherents than C/C++. This wasn't something Borland could afford to ignore, so Delphi eventually spawned a C++ based clone called

C++Builder using the same development environment, the same component model, the same debugger and the same compiler back-end. A C++ companion for Kylix is on the cards, too, though its chances of becoming a reality probably depend a lot on how well sales of Kylix itself go. Before that, Delphi 6.0 for Windows will make cross-platform development easier by adding support for the changes (from Delphi 5.0) that have been made in Kylix.

### Component Clicks

Chief among those changes is the introduction of an entirely new library of object-oriented components. Along with its visual interface design tools, point and click development methodology and a compiler that takes the wait out of the compile-link-test cycle, VCL (Visual Component Library) was what gave Delphi the edge over rival contenders for the RAD crown. But VCL was designed with nothing other than the task of creating Windows applications in mind. Many VCL components contain code that access the Windows API, or are derived from other components that access it. Consequently, porting the VCL to Linux wasn't a practical option for the Kylix developers. Instead, Borland created a new component library called CLX (pronounced "clicks"). The CL stands for Component Library while the X represents "cross-platform". Architecturally, VCL and CLX are similar, with functionally equivalent components, methods and so which for the most part share the same names. This makes porting programs written under Delphi with nary a thought for running under Linux a lot easier than you might imagine, although it's still by no means an automatic process.

Developers planning from the outset to create programs that will run on both Windows and Linux will create them using CLX components, avoiding calls to the API of either platform (or using conditional compilation if that is unavoidable). These programs will then compile under Kylix to produce a Linux binary, and under Delphi 6.0 to create a Windows executable. This opens up valuable opportunities for the enterprise. It makes it feasible for companies to consider deploying the free, open source Linux operating system as a platform for custom applications, with all the cost savings that implies, without burning their bridges and creating programs that won't run under Windows as well if necessary.

Besides a different component model, there's another subtle difference between a Delphi project and one created using Kylix. In Delphi, each form (window or dialog box) is represented by two files. One, the .pas file, holds Pascal source code. It

---

*“Developers planning from the outset to create programs that will run on both Windows and Linux will create them using CLX components, avoiding calls to the API of either platform.”*

---

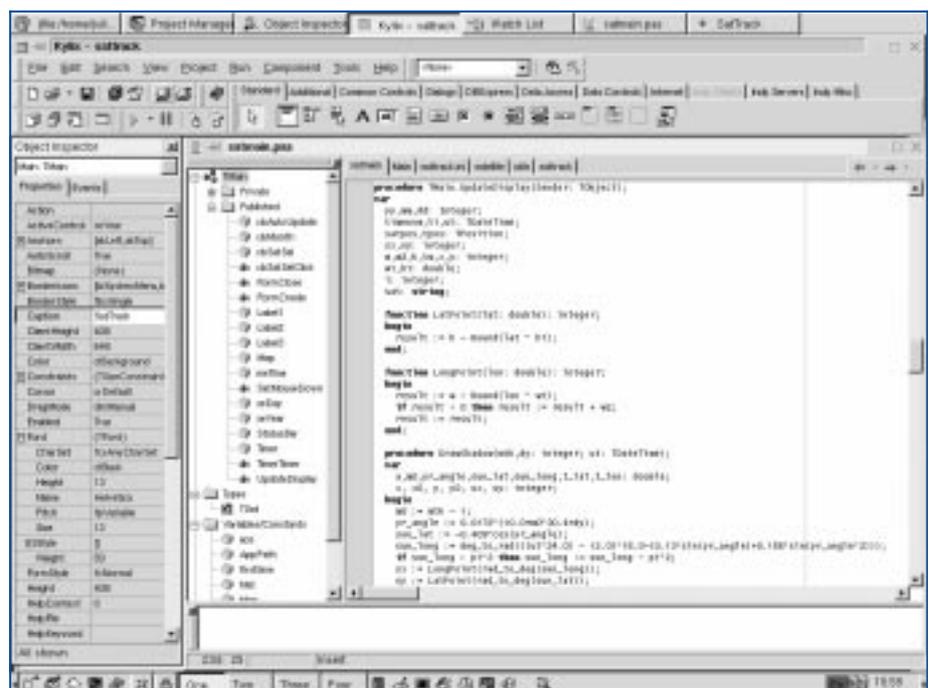


Figure 1 - The Kylix development environment is a leap forward in usability and productivity terms for Linux.

contains the declarations for the form and all the objects and variables belonging to it, and all the event-processing code that the programmer has written. The second file, the .dfm file, is a kind of resource file that holds the properties to be used for the form when it is created. In Kylix, the form resource file is an .xfm file. Although they both hold the same kind of information in the same kind of format, and most of the components share the same names, there are some important differences between the equivalent components which would cause errors if a Kylix project was opened in Delphi 5.0, or vice versa.

### **Qt Pie**

VCL is a set of objects that encapsulates Windows controls and the Windows API. However, the designers of CLX had no standard windowing API they could use. Graphical programs under Unix/Linux are displayed on-screen using X Window, but programs don't communicate directly with X. In the Linux display hierarchy, programs are built using a graphical toolkit that converts high-level function calls into calls to a window manager. The window manager determines the look, feel and behaviour of the windows but it isn't responsible for displaying the result. Instead, it sends a stream of commands to X, which does the actual rendering of the graphics on screen.

For the visual components of CLX, Borland chose to use the Qt graphics library from TrollTech. Qt is the library that was chosen by the developers of KDE, the most popular graphical desktop for Linux (though rival GNOME is gaining ground). With the formation last summer of the GNOME Foundation, which has some big-iron supporters like Sun and Hewlett Packard and grand plans for an object-based component architecture to rival Windows' COM, Borland must have wondered if it backed the wrong horse. Both Qt and gtk (the graphical toolkit used by the developers of GNOME) are available for Windows, which was an important consideration if CLX was to be made cross-platform. However, Qt was designed from the start to be a cross-platform toolkit, which probably made it a better fit with what Borland wanted to do.

Although the choice of Qt makes Kylix, and programs developed with it, run best under KDE, they run perfectly well under GNOME or any other of the popular Linux window managers. All you need to do is to install the Qt runtime libraries on the target system. A distribution license for them is included with the product.

Borland has tried hard to keep VCL and CLX as similar as possible, but there are important differences for developers to get used to. Visual elements under Windows are called "controls"; in Qt they are known as "widgets" and the base classes of CLX are named accordingly. Qt supports application-wide styles that determine the look and feel of a program's interface: you can either make it look like Motif or like Windows. Windows-specific features like dockable toolbars, MCI and COM aren't available in Kylix. Windows format bitmaps and icons are supported, but CLX - at least in the field test version I evaluated - omits some of the VCL's low-level graphics functions like the flood-fill operation and the ability to address rows of pixels: this is more likely to trouble games and multimedia programmers than corporate developers, however. Kylix even provides a component for storing and retrieving configuration settings in INI files, though unsurprisingly there's no emulation of the Windows Registry.

### **Availability**

Kylix, like Delphi, is pitched as a tool for database and Web application development. It comes in two versions: Desktop Developer costs around US\$1,000; and Server Developer is a hefty US\$2,000. Both versions are suitable for database development and include dbExpress, an interface to SQL drivers that provides access to any database engine. Drivers for Borland InterBase and MySQL are included in the Desktop Developer package. There's no Borland Database Engine for Linux, however, and the Paradox and dBASE file formats - still widely used in smaller database projects under Windows - are unsupported on this platform.

Kylix Server Developer adds dbExpress SQL drivers for Oracle and DB2, with an unlimited deployment license. Database access using MIDAS is also supported. The other major benefits are the inclusion of the NetCLX Internet and Apache web development components and the Apache WebBroker application framework.

### **The Linux Loader Bug**

If you decide to install Kylix or to deploy Kylix applications, one of the things you'll need to do is ensure that the systems the programs will be running on are using one of the latest Linux distributions. During development of Kylix, Borland discovered a bug in the Linux loader. It has now developed a fix, which ships with the product. Red Hat now has an official release of glibc 2.2 that incorporates the fix, and Borland provides patches for Linux Mandrake 7.2 and SuSE Linux 7.0. If you're using another version of Linux then an upgrade may be needed.

The bug relates to the reference count used to determine if a module can be unloaded from memory. Suppose an application loads module A, which has a dependency on module C. It then loads module B, which also has a dependency on module C. If the program now unloads module A or module B the reference count is updated incorrectly, allowing C to be unloaded even though a module that uses it is still running. This could result in a crash. However, this is far more likely to be a problem with Kylix applications - which dynamically load and unload modules called "packages" - than with programs built using traditional Linux development tools, which is probably why the bug went undetected for so long.

These are designed to facilitate the development of large-scale data-centric web applications that run on the popular Apache web server. The WebBroker CLX components are compatible with the VCL WebBroker components provided in Delphi 5.0 Enterprise, making it a relatively simple matter to port applications running on Microsoft Internet Information Server using ISAPI across to Apache running on Linux. If you're thinking about Microsoft's .NET instead, bear in mind that it's possible .NET will never be implemented on non-Microsoft platforms.

### ***Fans***

An important factor in the success of Delphi has been the number of enthusiast developers who have created VCL components and made them available on sites like the Delphi Super Page, usually complete with source code, either free or for a nominal cost. Using such components saves reinventing the wheel and can be an enormous time-saver for the developer. The cost of Kylix is probably too high for most Linux enthusiasts, so it remains to be seen whether CLX components will become freely available to the same extent.

Kylix Open Edition will be a free download and will also be offered on CD with hard copy manuals for \$99. This version will be only for open source and free software developers, however, and the differences between this and the Desktop Developer version are not known at this time. Check out the Kylix home page listed below for the latest news on this product.

### ***Conclusion***

If you are using Delphi in your organisation and also have an interest in Linux then you have a good reason to try out Kylix. Thanks to the similarity between Kylix and Delphi the learning curve will be almost non-existent. The reward will be the ability to deploy relatively low-cost Linux boxes in place of Windows 2000 servers or workstations and a consequent saving in hardware costs and software licence fees.

If your company already has Linux servers running database applications or Apache then the argument for Kylix is compelling. Until now, Linux has lacked a framework that encompassed both databases and Apache. Kylix provides this, making it much easier to develop large-scale high performance Web applications.

As a cross-platform development tool Kylix has some shortcomings, supporting as it does just Windows and Linux. Borland's JBuilder is better in that regard, since it runs on all the platforms that will run Sun's Java. It's also worth noting that at present Kylix only supports Linux on the Intel x86 platform. Itanium support is promised soon, but to realise the product's full potential as a tool for creating large-scale Linux server applications Borland will need to develop compilers for other processor architectures.

Borland has taken a bit of a gamble bringing Delphi to Linux. Whether it will emerge a winner is hard to guess. If the use of Linux by enterprises continues to grow, Kylix could emerge as the star of the show. But if Linux fails to take off on the desktop and Microsoft manages to snare web application developers with its .NET, Kylix could turn out to be Borland's Cinderella: all dressed up with nowhere to go.

### ***Resources***

Kylix Home Page  
<http://www.borland.com/kylix>

Delphi Super Page  
<http://community.borland.com/homepages/dsp>

Torry's Delphi Pages  
<http://homepages.borland.com/torry>

Dr. Bob's Kylix Kicks  
<http://www.drbob42.com/kylix>

**PCSA**

*Copyright ITP, 2001*

## New Reviews from [Tech Support Alert](#)

### [Anti-Trojan Software Reviews](#)

A detailed review of six of the best anti trojan software programs. Two products were impressive with a clear gap between these and other contenders in their ability to detect and remove dangerous modern trojans.

### [Inkjet Printer Cartridge Suppliers](#)

Everyone gets inundated by hundreds of ads for inkjet printer cartridges, all claiming to be the cheapest or best. But which vendor do you believe? Our editors decided to put them to the test by anonymously buying printer cartridges and testing them in our office inkjet printers. Many suppliers disappointed but we came up with several web sites that offer good quality cheap inkjet cartridges with impressive customer service.

### [Windows Backup Software](#)

In this review we looked at 18 different backup software products for home or SOHO use. In the end we could only recommend six though only two were good enough to get our "Editor's Choice" award

### [The 46 Best Freeware Programs](#)

There are many free utilities that perform as well or better than expensive commercial products. Our Editor Ian Richards picks out his selection of the very best freeware programs and he comes up with some real gems.