

# Open Source

by Rob Bracken

*The emergence of Linux as a competitor to Windows has put the open source movement firmly in the spotlight. But what is open source, and how will it affect developers like us? I hope this short article will help you answer those questions.*

So what is an open source project? Let's describe how you start one. First you need an idea for a useful program – something that a lot of people will want to use. Two well-known examples are Linux and Apache. Then you need to develop a first version of it. Now you can distribute it and get others interested. Finally, you must set up a way for other developers to submit bug fixes, feature requests, etc. The end result will be something that works reliably and is not tied to one company. If successful, it will develop a life of its own and will survive many changes to itself, to the environment it runs in, and to the group of developers maintaining it.

This process has a few implications. As mentioned, the program must be something that developers will want to contribute to. The development tools used must be readily available, so they don't present a barrier to anyone who wants to contribute. Delphi Personal and Kylix Open editions are freely downloadable, while it needs a considerable investment to use Delphi and Kylix Enterprise. A project developed with a free version is much more likely to gain momentum than one developed with Delphi Enterprise. A lot of the current open source projects are developed with the GNU tools, for this reason. Having got the development tools, then source must be freely available, otherwise how can anyone else fix the bugs? The best place is on a web site that everyone can access. Source Forge is host to many open source projects, but your project doesn't have to be on Source Forge to be open source.

## Open and Shut

What makes a good candidate for an open source project? The most obvious is the driver for a bit of hardware. Suppose you manufacture video cards. You're not really interested in writing the drivers for it, but if your video card won't work on Windows, you won't sell many. If, however, you produce a driver that works and then open source it, the card's users will maintain and bug fix it for you. You can then concentrate on the driver for the next model with lots of whizzy new features. This gives you an economic advantage over anyone who has closed source drivers – you don't have to produce a new driver for new versions of Windows, bugs are fixed quickly and the hardware will perform faultlessly with a reliable driver.

There are, however, only a limited number of projects that will lend themselves to the open source model. Most software will continue to be closed source, as it will not make sense to open it. After all, if you develop a project for a customer, it probably won't be relevant to anyone else, so opening the source will be a waste of time.

## Give It Away

Why does anyone develop open source software? Well, each has their own reasons for this, but there are several common threads. There is a certain kudos amongst fellow developers

if you're seen to be a major contributor to an open source project, and this is certainly a factor in most contributions. Another is the desire to give something back to the industry, and a third is dissatisfaction with commercial products. Then again, most developers enjoy just writing programs, and an open source project is an ideal way to leave your mark on the world by doing something you enjoy.

It is possible to make money from open source software. Just because you're planning to release a project as open source doesn't mean that a customer shouldn't pay you to write it. If the software will be of use to a number of customers, it may make sense to release the source, to improve the quality of the finished product. Someone has to pay you to write the software in the first place, but once it's written you get the benefits of a larger maintenance team and end up with a more reliable product. If your customer's business depends on this software, the increased reliability could be very attractive. Another way is to release your open source software to the world and then sell support services. e-smith have built a business around this model (see [www.e-smith.org](http://www.e-smith.org)). You can also add value to open source and sell the result (this is how the Linux distributions – e.g. Red Hat and SUSE – run their business, though they also have substantial sponsorship from interested parties such as IBM). Or you can sell accessories, as Tim O'Reilly does with his books.

Delphi add-ons are already close to the open source model. If you buy a component set, you generally expect to get the source as well. If you find a bug, you can fix it. What's missing, though, is the feedback to the original developers. Yes, it is possible to send the bug fix in an e-mail, but this isn't really encouraged, and there isn't usually any sense of being involved with the development. After all, it would be very unusual for the developers of a component set to allow just anyone to make a decision about which features will be included in the next version.

## Impact

Open source has already made a significant impact on the software world. Microsoft see it as a major threat to their business (see <http://www.opensource.org/halloween/links.html> for more information). Their response is to denounce it as evil and un-American on the one hand, and to think how they can use it on the other. From various rumours, I don't think there's much doubt that they were forced to make Windows 2000 more reliable to compete with Linux. Another open source project – Samba – has been a major influence in persuading IT departments to install Linux servers, BIND is used to convert domain names to addresses and Apache is the most popular web server on the Internet.

How can Delphi developers get involved with open source? There are quite a few open source Delphi projects, of which

the best-known is Indy – the Internet component set. DUnit – the test framework – is also open source. You can find a list of open source Delphi projects at [http://delphree.clexpert.com/pages/supported\\_projects.htm](http://delphree.clexpert.com/pages/supported_projects.htm). There is also the JEDI project which is best known for its conversions of Windows and Linux API header files for Delphi. That's not all they do, though. They're developing a library of useful utilities, a library of VCL components and a UML-based component builder.

## Frightened?

I don't believe we should be frightened of open source. It'll continue to create waves in the software industry, but it won't mean that we all have to work for nothing. On the contrary, it's producing high quality Delphi add-ons that make our life easier, and I recommend everyone to get involved in some way, however small.

I've only touched briefly on the issues surrounding open source, but I hope I've answered some of your questions. For more information, see:

- <http://www.opensource.org> for information and discussions of projects, business models, etc.
- <http://delphree.clexpert.com> for information about Delphi open source projects
- <http://delphi-jedi.org> for the Delphi JEDI project (Joint Endeavor (sic) of Delphi Innovators)
- <http://homepages.borland.com/ccalvert/Opinion/JediProjects.html> for Charlie Calvert's discussion of the JEDI project.
- "The Cathedral and the Bazaar" by Eric S. Raymond (published by O'Reilly books, ISBN 1-56592-724-9) for discussions of Open Source and some historical information.



*Rob Bracken is a Delphi developer and managing director of Bracken Software. His spare time is currently occupied with making a Blue Peter Tracy Island for his son (see [http://www.bbc.co.uk/cbbc/bluepeter/makes/m\\_tracyisland1.shtml](http://www.bbc.co.uk/cbbc/bluepeter/makes/m_tracyisland1.shtml)). You can contact him at [bug@brackensoftware.co.uk](mailto:bug@brackensoftware.co.uk).*