

Windows-to-Linux Migration with Qt

Combine the power of Qt with the ability to write one application that runs on Windows, Linux and Macintosh, and you get a bright future for Linux on the desktop.

Qt (pronounced “cu-tee” or “cute”) is the C++ application development framework that forms the basis of KDE—the K Desktop Environment—and it is used by Linux applications such as Scribus and Skype. It is developed by Trolltech, a Norway-based software company whose product line is centered around Qt. Although Qt mainly focuses on GUI (graphical user interface) functionality, it also provides excellent support for various programming domains, such as internationalization, networking, multithreading, SQL and XML.

From day one, Qt was a fully object-oriented cross-platform toolkit, with support for both Linux/UNIX and Windows. In 2000, Trolltech released Qt/Embedded, which was designed to run on embedded Linux devices and provided its own window system as a lightweight alternative to X11. And with the release of Qt 3.0 in 2001, support was added for Mac OS X.

Who can use Qt?

Trolltech’s business model, based on the idea of dual-licensing, works as follows:

- Developers who want to give away their source code to the Open Source community can use the Open Source (GPL) edition.
- Customers who make money on their software or use it in a commercial context must purchase a commercial Qt license.

The open source edition is first and foremost a gift to the open source community. Many developers at Trolltech have their roots in that community, and most of them use open-source and free software every day to do their work.

From a Linux perspective, it means that a high-quality professional tool, developed by a dedicated team of paid developers and used by thousands of paying customers, is available for open-source software development. What’s maybe less obvious is that Qt’s single source approach to cross-platform development and



its dual-licensing model accelerate the migration of Qt-based commercial Windows applications to Linux.

A better MFC than MFC

Although Qt is cross-platform, many companies use it for single-platform development. These companies use Qt because they find the Qt API (application program interface) superior to that of the Windows-specific toolkits. They also find that Qt insulates them from differences in the Windows APIs. For example, with Qt, the same executable works on Windows 95 to XP; Qt performs a check at runtime and uses the most advanced capabilities available. Microsoft's library of tools for the Windows API is called Microsoft Foundation Classes (MFC). With MFC, this is not possible without creating two executables.

Rowley Associates based their CrossStudio IDE on Qt/Windows. They have now ported their application to Linux at the request of some of their customers. The port was done by a developer who had no previous Linux/UNIX experience and took only a couple of days, showing how easy it is to transform a well-written Windows Qt application into a native Linux application.

Toolkit confusion on Windows

Microsoft has practically given up MFC and is preparing to replace its successor, Windows Forms, with the Avalon framework. The uncertainty surrounding these three APIs has alienated many Windows developers, who are now looking at Qt as a more stable alternative.

JMP—a division of SAS—who previously used MFC in statistical data visualization software, found themselves in that situation. JMP software is widely used in industrial companies—notably Dow Chemical, Honeywell International and Aventis Pasteur.

Richard C. Potter, Senior Software Manager at JMP, summarizes the search for an alternative to MFC as follows: "We needed to find a class library that would be compatible with our existing architecture. We briefly considered using GTK+, but we quickly realized that it would not be acceptable. Qt was a better fit with the existing architecture of our product. Since Qt classes match closely with those in MFC, converting our existing MFC-based source code into Qt-based source code was straightforward."

Another example is Firstlogic, which provides customer relationship management (CRM) solutions to more than 6,000 businesses around the world. When they set out to develop IQ8, their next-generation data quality solution, they decided to move away from MFC to Qt because they needed a more modern and flexible toolkit. Now, IQ8 is available for Linux and commercial UNIX systems.

How MAC OS is helping Linux

Although the uncertainty surrounding Windows Forms and Avalon is an important factor playing in Linux's favor, another factor is that software companies increasingly

need to target both Windows and Mac OS X. They choose Qt because of its cross-platform capabilities.

BMPi, the interactive version of "Birds of the Western Palearctic", developed by Skylark Associates, is such an application. Other examples include PerfectTablePlan by Oryx Digital and Mindawn by theKompany.com. As Linux becomes more popular as a desktop platform, the providers of these cross-platform applications are likely to port their software to Linux as well as Mac OS X.

The open source factor


Interestingly enough, many companies consider the existence of an Open Source (GPL) edition of Qt as an advantage over other toolkits.

For them, this means that the product is widely tested by hoards of open-source software users, including KDE users. Mailing lists, on-line forums, books and certain components (for example, the Qwt plotting library) are available on the Internet to both open-source and commercial developers. Finally, when it comes to hiring, there is a large talent pool of open-source developers with Qt skills out there.

About this topic, Rainer Goebel, who is Chief Software Designer at Brain Innovation, a company that migrated from MFC to Qt, says, "We ended up choosing Qt due to its consistent object-oriented approach combined with its elegant signal-slot mechanism. The knowledge that Qt was used to develop KDE, one of the two major Linux/UNIX desktops, also helped in the decision process."

The future

Even though Qt was originally designed as a cross-platform GUI toolkit, there is an increasing number of companies who adopt Qt for single-platform development on Windows, rather than Microsoft's half-baked solutions. With the emergence of Linux on the desktop, these companies will be in a position where they can port their applications to Linux in very little time, which in turn will help user adoption of Linux. It also may be noted in passing that Qt is helping Linux adoption on embedded devices. Qtopia, Trolltech's application suite for smart phones and PDAs, is based on Qt/Embedded, which in turn is based on Linux.

With the growing popularity of Linux and Mac OS X, companies are becoming increasingly concerned about portability, even when they have no immediate plans to port their applications. They don't want to be locked in to an unsupported technology, such as MFC or Windows Forms, and prefer to turn to a widely used and actively developed cross-platform toolkit. For many of them, Qt is that toolkit. 

By: Jasmin Blanchette. The author is documentation manager at Trolltech in Oslo, Norway. You can reach him at jasmin@trolltech.com.

Courtesy: The Tux Magazine