



Developing Portable Applications with C++



software that makes sense



www.troll-tech.com

T R O L L
T · E · C · H

A GUI Framework for

Qt is a C++ GUI application framework which provides a uniform Application Programming Interface across different operating and window systems. It gives you the power to develop applications with state-of-the-art graphical user interfaces, and easily port them to new platforms. The screenshots you see on this page show the same Qt-based application running on Microsoft Windows and Unix / X11.

Designed for Real Work.

Qt is designed by programmers with firsthand experience in using the product for real work. At Troll Tech, we use our own toolkit for making real, commercial applications, thus learning the hard way exactly how the API should be designed.

We know that no design is ever complete after the first try. The Qt API is the result of continuous design iterations, refine-

“Qt has a lot to offer . . . fully object-oriented, easily extensible and much faster than Motif for important operations. Its architecture is cleaner than Motif’s. Qt is exceptionally well documented . . .”

– Cameron Laird, SunWorld,
March 1998.

applications. After the first prototype is made, and the real, demanding customer requirements start to come in, then you don’t want to be locked in with a tool that limits your creativity. And you definitely do not want to maintain the “write-once, read-never” sort of code typically produced by some tools.

What you want is an open application framework that gives you real control: Qt.

Open Source and Peer Review.

We invite you to look under the hood: Qt is provided with full source code. We’re not shy about our code; we’re proud of it! Having the source code available also means that you can modify it to suit your needs, and you can trace any problem all the way down.

The Qt source code has been scrutinized by hordes of programmers in the free software community for years – giving us valuable feedback, which we use to continually tune our toolkit for better usability and performance.

Real Extensibility.

No toolkit can offer everything you will ever need. A real application will always have special requirements for data display and user interaction; it will need custom-made widgets. Have you ever tried to design your own widget using Motif? Or MFC? Then you will know why we have designed Qt with extensibility built in from the start.

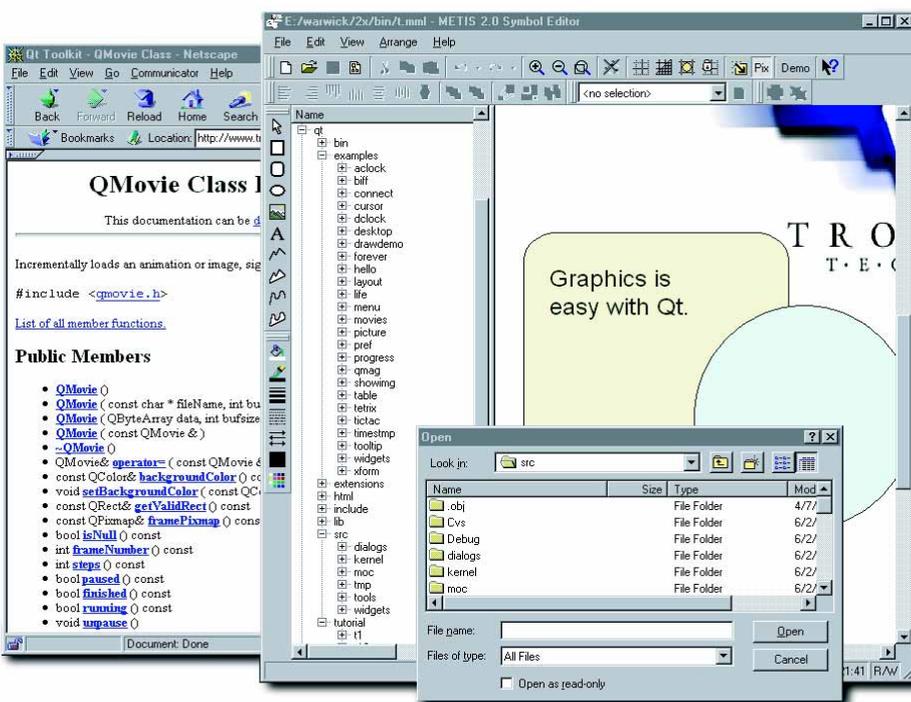
Making your own widgets with Qt is a simple matter of using C++ inheritance; there are no hidden details. The standard widgets in Qt are designed to be inherited, so you can

ment, and enhancement of the classes. We know that this is the only way to achieve the standard that both you and we want.

Revolution? No Thanks.

For decades, new, “revolutionary” Rapid Application Development products have been released. No revolution has happened. And you know why: It simply doesn’t work.

We have seen how these tools work for only the simplest



or Real Development.

easily extend their functionality to cover your specific needs. All the standard widgets that are included with Qt are made in this way, so you can study their source code to learn for yourself how the various features can be implemented.

but without the sender and receiver having to know about each other, and with the added power of type safety. The signal/slot mechanism overcomes the problem of fitting C++ to the demands of event-driven programming.

On-Line Documentation.

The Qt Reference Documentation is provided on-line in HTML format, and is fully cross-referenced. It is automatically generated from the documentation-augmented Qt source code, so it is always up-to-date with respect to the API. In addition there are Qt programming guide books available from O'Reilly and others.

No Empty Promises.

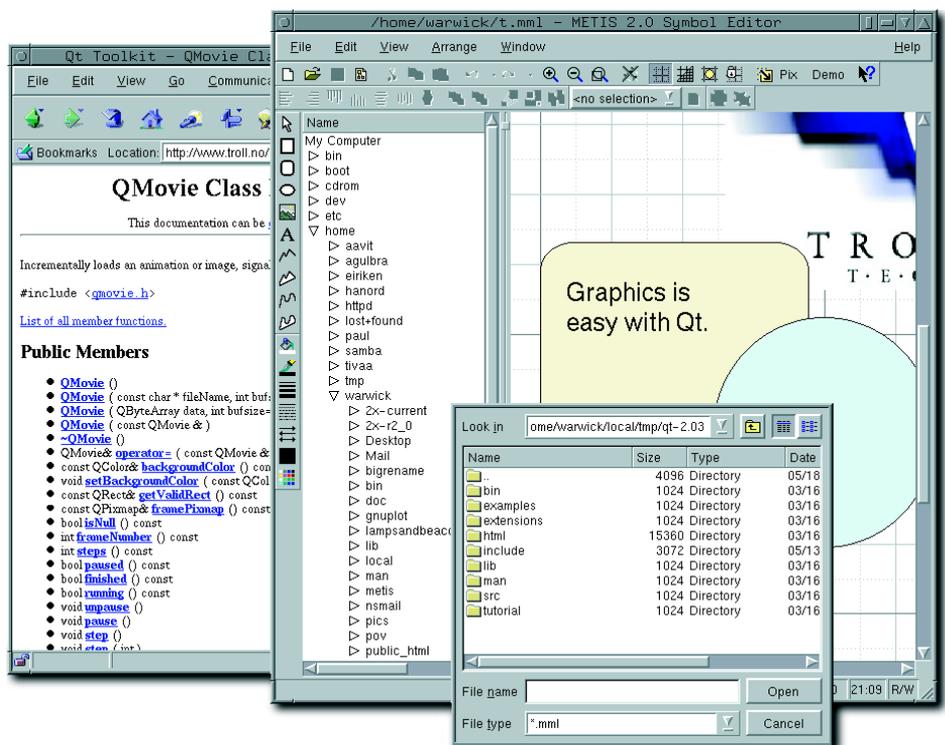
We will not promise you that by using Qt, you will increase your productivity 10 times, or that you will be able to make your application in a day, or any similar presumptuous assertions. Our industry is riddled with software development

Technical Support.

At Troll Tech, we take pride in providing our customers with the best technical support. Inquiries are handled by the Qt developers themselves, so you can be assured of getting real help.

Component Programming.

Qt is fully object-oriented and is designed to allow you to develop your code as independent, reusable components. We achieve this by the unique signal/slot mechanism in Qt, which gives you all the power of callbacks,



"It's not difficult to see why Qt is gaining such a strong following . . . I'd strongly encourage you to download the code, look at the comprehensive HTML-based documentation, work through the tutorials and have a play. I think you'll be impressed . . . Because KDE, and the numerous 'K' utilities that come with it are all built using Qt, you've essentially got a glowing testimony to the power of the Qt library right in front of you as you work! . . . I'm deeply in love with Qt."

— Dave Jewell, Developer's Review, August 1999.

products that make such claims, but experienced developers know that it doesn't work that way. Making real applications takes real work — there is no easy way out.

Instead, let us give you a real example of productivity. A team of seven programmers were able to port the front end of Netscape Navigator to Qt in only five days. The front end, consisting of several hundred thousand lines Motif and MFC based C/C++ code, was replaced with only twenty thousand lines of Qt-based code. The result is a browser called QtMozilla. You can see it in the screenshots, and you can download it from our web site.

We believe that the only thing that can really improve your productivity is to use a toolkit that enables you to give your applications a clean, efficient, readable and maintainable design.

Qt is such a toolkit.

Qt. Software That Makes Sense.

Qt is used all over the world for creating the GUI of successful applications. Qt provides total platform independence, turning the job of porting your application to new operating systems into a simple matter of recompiling.

Mature

Since 1996, companies like Alcatel, HP, IBM, Intel, Lockheed Martin, ESA, NASA, and Siemens have based applications on Qt. We continuously improve and extend Qt with new features, GUI elements, and extensions.

The Linux Solution

Qt is the obvious choice for your Linux project. Qt is the basis of the KDE, the default desktop of most major Linux distributions. The KOffice office suite, the Caldera Linux Installation Wizard, and a host of other Linux software are all built with Qt. It allows you to bring your existing Windows software to Linux while avoiding the maintenance of two codebases: using Qt, the same application code can be built on both Windows and Linux.

Versatile

Qt is employed in everything from web browsers to satellite simulation systems. For 3D graphics applications, we provide full integration with OpenGL as well as Tech

Soft America's high-level HOOPS graphics subsystem. Qt has powerful image handling capabilities, and can easily be extended to handle custom image formats through the Image I/O extension. With the LiveConnect Extension, it is easy to build Netscape plug-in applications which can make the browser display your custom data integrated in normal web pages.

International

Qt text handling operations use the 16 bit Unicode standard throughout, so using character sets such as Japanese or Cyrillic is straightforward. Qt supports application localization, where the same Qt-based executable can choose at runtime from any number of different languages.

Portable

Qt provides complete portability across Microsoft Windows 95/98/NT, Linux, Solaris, HP-UX, Digital UNIX, IBM AIX, BSD, SCO, SGI Irix, and more. On Windows, the C++ compilers from Borland and Microsoft are supported. On the Unix

systems, we support the native compilers as well as g++/egcs. Qt gives C++ programmers the same cross-platform benefits as Java, while still allowing direct access to the underlying operating system.

Efficient

Qt is highly optimized for execution speed and is memory conservative. While most other cross-platform libraries are layered on top of Motif and MFC (or MFC or Motif emulation layers), Qt sits directly on top of the native window system. This makes Qt-based applications far leaner and more responsive. Qt also avoids the performance penalty of Java's virtual machine environment.

Easy to learn

The Qt programming interface is coherent, elegant and intuitive. It is our experience that new developers typically come up to speed far sooner with Qt than with e.g. Motif or MFC. Our users say that Qt gives them the feeling that *"this is the way programming was always meant to be."*

Download your evaluation version today!



T R O L L
T • E • C • H

www.troll-tech.com

General inquiries: info@troll.no

Pricing and availability: sales@troll.no

Phone: **+47 22 80 63 90**

Fax: **+47 22 80 63 80**