

# Qt Creator 2.0

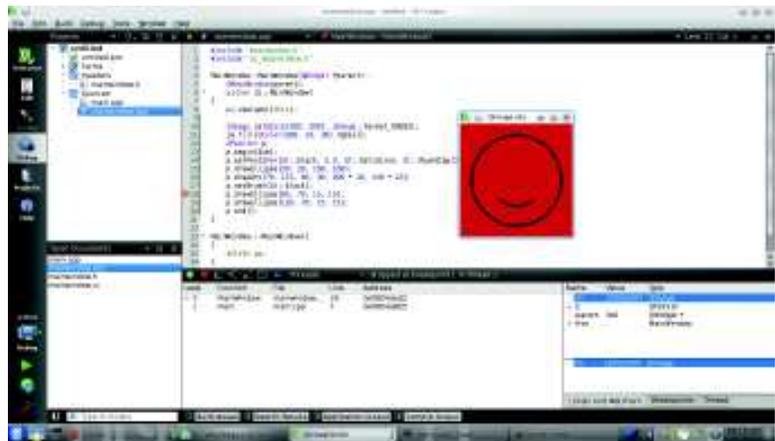
A month after the release of KDevelop 4, **Graham Morrison** reviews its arch nemesis. But which one's best? There's only one way to find out...

## In brief...

» A development environment for C++ programmers who use the Qt framework. Also consider Eclipse, KDevelop 4 or plain old Kate.

**H**ot on the heels of *KDevelop 4*, and several months before we anticipated, Nokia released version 2 of its groundbreaking integrated Qt development environment. It had been expected to be part of *Qt 4.7*, which is still in alpha testing. Instead, it has been released as part of Nokia's *Qt SDK 1.0*, a slightly modified version of Qt that includes a device emulator and support for MeeGo. You can also grab it as part of the *Qt 4.6.3* package and as a standalone project.

Fortunately, installation is easy. You grab a large binary, make this executable, and run it from the command line. If you do this from your user account, all the required files install automatically, including a link to *Creator* in your Desktop folder. Clicking on this will launch the IDE. You won't even need the Qt development libraries, as these are included by default. Ubuntu users,



» *Qt Creator's* debugging capabilities are greatly improved – it can even debug image types by rendering their output to a temporary window.

## “Qt Creator can now make a lot more sense of your Qt code.”

for instance, will only need to install the **build-essentials** package, followed by Qt, to start developing.

A lot has changed since version 1.0, which now feels more like a prototype than a major release. The new launch wizard, for example, is one of the best

we've seen, and for the first time, can genuinely help the newcomer. You can start by exploring examples that give an overview of the UI, then move on to some simple template tools before working your way through the official Qt docs. All of this is done through the embedded help system, so you never need to leave the main window.

## Bugtastic

*Creator 2.0* loaded all of our 1.x projects without problem, and you can now work with more than one project at a time, switching between their code trees using the panel on the left-hand side. The only problem with this is that it's never clear which project is going to build and run. The solution is to use the new target selector tool, just below the build icon. This enables you to switch between projects to build, displaying their name beneath the play icon, as well as letting you choose between debug, release and architecture.

The syntax highlighting is just as good as in the previous version, with small additions like symbol names being emphasised when you use them within a line. You can even use this feature to refactor your code. Right-click on a symbol, select 'Rename Symbol Under Cursor', and as you type a new name for it, all other occurrences are updated in real time. This function even stretches to other files, such as the header, when you change the name of an object. And

if you can't find the object, you can now search using regular expressions.

Another area of rapid expansion is *Creator's* debugging. It's now fully compatible with *GDB 7*, and can take advantage of the new reverse direction feature that lets you step back through an application's execution. What's more important for Qt developers is that *Creator* can now make a lot more sense of your Qt code using a series of Python helpers that automatically inspect certain Qt and standard datatypes.

There are still a number of annoying design decisions, such as the watch window only appearing in debug mode, but *Creator 2.0* is a massive step up. Add true cross-platform capabilities and MeeGo development, and *Creator* becomes a killer app. Roll on *Qt 4.7*. **LXF**

LINUX Verdict  
FORMAT

**Qt Creator 2.0**  
**Developer:** Nokia  
**Web:** <http://qt.nokia.com>  
**Price:** Free under LGPL

<b>Features</b>	9/10
<b>Performance</b>	9/10
<b>Ease of use</b>	7/10
<b>Documentation</b>	10/10

» *If you like Qt, this is the best development environment you can use. If not, you should try Qt again.*

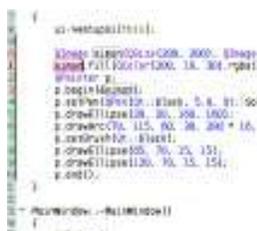
Rating 9/10



## Features at a glance



**GDB 7 support**  
Reverse debugging and excellent Qt object support helps fix problems fast.



**Refactoring**  
Change a symbol's name and it changes in all other instances.