

# The New QNX Hybrid Software Model

Combining open source and proprietary benefits for embedded systems

By Lawrence Rosen

Lawrence Rosen is both an attorney and a computer specialist with deep roots in the open source community. He is founding partner of Rosenlaw & Einschlag, a technology law firm that specializes in intellectual property protection, licensing, and business transactions for technology companies. In addition to this law practice, Larry also served for many years as general counsel and secretary of the non-profit Open Source Initiative (OSI). He currently advises many commercial open source companies and non-profit open source projects, including the Apache Software Foundation. His book, *Open Source Licensing: Software Freedom and Intellectual Property Law*, was published by Prentice Hall in 2004.

QNX Software Systems has asked me to review their new software model in the context of open source, and I'm pleased to do so. The new model integrates open source and proprietary software products in new ways. It is a step forward in the embedded systems market toward openness and freedom of software development, and it gives QNX customers significantly greater flexibility to extend and adapt QNX technology for their own purposes.

The new QNX model is an effort to address fundamental problems in the way proprietary embedded software is traditionally developed and distributed. Today the rate of change to software and hardware is so rapid, and software so complex, that vendors and customers alike struggle to keep up. Often, software vendors are their own worst bottleneck, as they work to fix or extend their existing products while also attempting to satisfy new, and often divergent, customer needs. Meanwhile, the sophisticated users and customers in the embedded market often know exactly what features and functions they want; many would make the modifications themselves if allowed to do so. And many of them would welcome opportunities to cooperate and share the results of their collective development efforts — just as they would in an open source project.

A pure open source approach doesn't work in all cases, and it doesn't work for QNX. They do not believe that relinquishing all control over their intellectual property and giving it away for free would best serve the interests of their customers.

Technology companies implement their fundamental business strategies through licensing their intellectual property. It is a subtle task. If a company gives too much away through overly generous grants of copyrights or patents, then its competitors and customers get a "free ride" on its products; the company loses its incentive to invest in research and development. If the company makes restrictions on use too tight and complicated, it discourages customers from taking advantage of what its products have to offer. This is where QNX is looking to innovate, with a new blend of transparent development and accessible licenses for the embedded development community.

For me, open source is usually part of the answer. I have been involved for a long time in the open source community, and have served as a lawyer for both for-profit companies around the world and non-profit organizations like the Open Source Initiative and the Apache Foundation. The goals of open source — built upon licenses that promise freedom to use, copy, modify, and distribute the software that people receive — are part of my nature. They are also becoming part of the nature of the entire software and technology industry. Customers and vendors alike demand open source advantages to be included in their software-based products.

That is what QNX is doing: Offering their development community the freedom to proceed without the company being the bottleneck. It is an enablement strategy that combines the benefits of an open source development model with the sustainability of a royalty-based business model for commercial projects. It isn't entirely open source; rather, it's a mixture of open source and proprietary software and rules. I'll explain why.

## What's already open and what isn't?

There is already much open source software inside QNX runtime technology and associated development tools. The varied and growing list of the open source components of QNX software is published at <http://licensing.qnx.com/license-guide/>. Many of those components are available for free use. QNX licensees are encouraged to take those open source components and do with them whatever their open source licenses allow.

In addition to incorporating this open source software into its products, QNX also serves as a major contributor to open source software widely used by other companies around the world. QNX was a founding member of Eclipse and continues to manage one of its most popular projects, the Eclipse C/C++ Developer Tools (CDT), which is based on code that QNX donated to Eclipse.

QNX has released major components of its own software under open source licenses and will continue to release even more over time. Most important, at least initially, many of the QNX board support packages (BSPs) are now available under the Apache License, Version 2.0 ("Apache 2.0"). This is key to extending the amount of usable hardware available for QNX applications and will enable users to build their own BSPs to satisfy their own needs or those of other QNX users. QNX deliberately chose Apache 2.0 as its open source license for this code in order to give developers the option to offer their derivative works — of BSPs in this case — for free or for a fee. Apache 2.0 doesn't force developers to publish their derivative source code, yet it provides a framework for open cooperative development, if that is their preference.

Meanwhile, key proprietary components of the copyrighted and patented technology at the heart of QNX runtime software remain available only to QNX licensees, as are certain value-added features of the QNX developer tools. Nothing in this new QNX business model changes that basic rule. The public cannot freely copy, modify, or distribute QNX software, except for the specific open source components within it. QNX software as a whole — meaning the QNX<sup>®</sup> Momentics<sup>®</sup> development tools, the QNX Neutrino<sup>®</sup> RTOS, and a variety of middleware — is available for use only by QNX licensees and cannot be redistributed to third parties without QNX permission. That stays the same.

However — and this is what's new — QNX is offering much more visibility into its development process and is granting developers much more freedom to modify, enhance, and share licensed copies of QNX software and to create new applications around QNX software for their own purposes. Building on its Eclipse experience, QNX has started to publish the source code for key parts of its runtime products and will conduct the ongoing product development for those products in the open, for anyone to follow. Non-commercial development licenses for the full-blown commercial version of the QNX development suite, which includes the QNX Momentics development tools and the QNX Neutrino RTOS, are available for free. Partner licenses are also available at no charge for anyone looking to add their products to the QNX ecosystem.

QNX is, in effect, creating an open source community within its existing and growing community of RTOS, middleware, and development tool licensees. As a result, *anyone interested in QNX technology can now cooperate on development for the benefit of the community as a whole*. At the same time, by publishing its QNX Neutrino RTOS source code and by licensing its BSPs under Apache 2.0, QNX is inviting others to take the powerful QNX technology platform down new open or commercial development paths. QNX has even created opportunities so that commercial developers can implement and promote the use of QNX technologies for use with target operating systems other than the QNX Neutrino RTOS, and is prepared to license its proprietary technology for those purposes.

To enable these activities, QNX intends to publish all of its runtime component source code (some source code won't be published immediately because of third-party licensing or confidentiality restrictions) and to let developers use that code to create derivative works. As this source code is published, the associated product development activity will also be moved into the public arena.

Traditional open source communities are, in some sense, open to anyone who wants to participate and to follow community rules of behavior and licensing. This QNX community is very similar, but the laws of intellectual property — and the limitations that QNX places on the use and distribution of its copyrighted and patented software products — gives this community more of a commercial feel and practice. Anyone can join, and they can become QNX licensees (for free!) as long as they promise not to license their QNX or derivative work software to third parties who aren't also QNX licensees, or unless they get a commercial distribution license from QNX.

This community consists only of QNX licensees; nobody else. That isn't open source, but it is a realistic modification of open source rules to create an open development community for QNX software, which is used at the heart of commercial products built and sold by commercial companies around the world. Outside of the community of QNX licensees, QNX proprietary software is published but it isn't open. Within the QNX community, developers enjoy the benefits they would find in an open source development environment while at the same time still being able to leverage the advantages available to those who use proprietary products.

## **Building the QNX development community**

Open source software thrives when a community of users and developers cooperate to develop new solutions for the entire community to share. There are many successful open source projects that work on common goals, exchanging ideas and code, mentoring and motivating each other, building product expertise, forming partnerships, and profiting from their collective work.

QNX wants its software to grow through that kind of community effort.

Perhaps the most important aspect of the new QNX strategy, then, is the creation of a user and developer community that is internally open and sharing, even though parts of it remain closed to those who don't license QNX software. Within the community, developers can find all of the beneficial aspects of open source development, including transparency of the contribution process, visibility to priorities and projects, merit-based community collaboration, and freely available development tools and resources.

To provide access to developer and customer resources relating to the QNX Neutrino RTOS and the QNX Momentics development tools, QNX is launching a community-oriented web portal called Foundry27. Anyone can access information from QNX and from others in the community about QNX products and services (including all published source code). As with most open source development projects, free registration is required to get write privileges for wikis and forums.

As the commercial entity that will provide the resources and website upon which the community will work and share, QNX will also help the community with the basic maintenance and coordination needed to sustain a healthy environment. QNX will publish a development roadmap, take steps to evaluate contributions, verify the provenance of contributions provided for adoption into the head branch, and provide infrastructure support through the portal for the benefit of the entire community.

By downloading the QNX development suite and applying for a license key, developers will gain access to a free copy of most QNX development tool and runtime software — including repository download rights for all published source code. Depending upon the software license(s) that they qualify for, developers can use that software for many purposes, such as experimenting with and prototyping target systems, extending hardware and peripheral support for the QNX Neutrino RTOS, and developing new applications for or porting applications to the QNX Neutrino RTOS. Members of the QNX community can exchange their software solutions with any other QNX licensees, as specified in the set of new licenses that QNX is introducing.

## **Sharing within the community**

Within the QNX community, developers are encouraged to share their modifications to QNX code with one another for experimental and commercial uses. If their derivative works are based on proprietary QNX code, then they can do so provided their code is offered only in substitution for the original form of the work licensed directly from QNX Software Systems and only as long as that original work remains licensed from QNX. This substitution concept, expressed in the QNX licenses outlined later in this paper, enables others to share modifications to what remains an underlying QNX proprietary code base.

QNX also encourages the sharing of modifications to code that QNX has published under Apache 2.0. This well-known open source license created by the Apache Software Foundation ([www.apache.org](http://www.apache.org)) allows licensees to make copies of Apache-licensed software, to create

derivative works by themselves or with help from others, and to distribute their software to other licensees — it also licenses the necessary copyright and patent rights to do these things and to use the software as it was designed to operate.

### *Three classes*

QNX has created an innovative way to enable free sharing of derivative works based on either open source or proprietary QNX code. Their new hybrid software model divides QNX products into three classes, all of which are available under one or more of the licenses described in the next section or under one or more open source licenses:

1. The first class of software is a small (and growing smaller) set of patented or copyrighted proprietary QNX software that is based on unpublished source code. Soon this will be limited to certain QNX value-added tools and some QNX middleware products.
2. The second class is a large (and growing larger) set of published source code for proprietary components of QNX software that is available for the creation and sharing of derivative works within the QNX community.
3. The third class is a large (and hopefully growing much larger) collection of published source code that is available from QNX under open source license terms, or that has been made available for free from other members of the QNX community to satisfy customer and community needs.

Derivative works from community members and executables built for target systems based on the QNX Neutrino RTOS will often depend on software in the first or second classes, and to that extent companies will need a commercial patent and/or copyright license from QNX to distribute such products.

Deciding what software goes into what class is a balancing act. Claiming too many intellectual property rights for QNX will limit the ultimate success of the community that QNX hopes to empower. The balance is maintained by the company's commitment to publish more and more of its software over time, and by its promise to cooperate to allow its customers and its development community greater creative and licensing freedom with QNX software.

QNX Neutrino RTOS runtime technologies and the QNX Momentics development tools aren't "open source" in the way that the Open Source Definition requires, and don't claim to be. But the QNX approach to enabling the sharing of derivative works within the community *is* open source, and is familiar to anyone who has received and used open source software. This shared source code will help developers to create new and varied applications for the QNX Neutrino RTOS and to share them, in turn, with others. It will also help companies that are building target systems, and companies that are creating new tools for the QNX development suite, to make even more capable products.

This new, more transparent development model serves the needs of QNX customers without giving away QNX's valuable copyrighted and patented technology for free. It incorporates the advantages of the open source development and distribution model, but strictly within the community of QNX Neutrino RTOS licensees who are themselves licensed to benefit from shared advances in the QNX software technology.

## New licenses to set community rules of sharing

When developers download the QNX Momentics development suite, they can choose from one of three QNX licenses, the first two of which are free of charge:

- **Non-commercial end users** — Licensees may receive the QNX development suite, which includes the QNX Momentics development tools and the QNX Neutrino RTOS software, under a royalty-free *QNX Non-Commercial End User License Agreement (EULA)*, for certain evaluation and limited development purposes.

This *EULA* is intended for individuals or companies to experiment with QNX software and to prototype target systems. A licensee can even demonstrate their new RTOS-based products to others, *provided that the licensee doesn't leave copies with third parties who aren't themselves licensed by QNX*. The *EULA* is also designed for academic faculty to train their students.

- **QNX community partners** — Licensees may become QNX technology partners to offer their own products and services to QNX customers. The expectation is that many independent developers will want to take advantage of the opportunity to market their derivative works or add-ons commercially through the QNX portal or elsewhere, or to use their licensed QNX development suite to support QNX customers.

QNX now offers its technology partners the *QNX Partner Software License Agreement (PSLA)* at no charge. The *PSLA* permits developers to use QNX software for commercial activities associated with their hardware or software businesses in support of QNX products.

- **QNX commercial customers** — The most important participants in the QNX development community, of course, are the companies that create products with QNX Neutrino RTOS software. For these companies, QNX software is available for the development of commercial applications under the *QNX Commercial Software License Agreement (CSLA)*.

This development license isn't free. It includes certain important warranties and indemnities by QNX that are appropriate for commercial software. The *CSLA* can also provide, through the *QNX Middleware Addendum*, access to certain QNX middleware products, which are additional (often patented) features available for specialized applications. A *Standard Support Addendum*, under which QNX will provide product updates and support licensing, is included in the *CSLA*, but commercial customers can also benefit from enhanced QNX

support with an optional *Priority Support Agreement*. Licensees will need to execute a separate *QNX OEM License Agreement* or *QNX Runtime License Agreement* in order to manufacture and distribute target systems that embed the QNX Neutrino RTOS software.

All of these licenses authorize participation in the QNX development community. All of them allow developers to develop derivative works of QNX software that can be distributed to other QNX licensees, and they enable a community where each developer can benefit from the efforts of other developers.

## License keys and sharing of software

Access to QNX software, and access within the QNX web portal, is controlled by license certificates and license keys, just as it was previously; certificates and keys will be assigned to companies and individuals who expressly accept the software under the terms of the *EULA*, *PSLA*, or *CSLA*.

None of the QNX licenses allow developers to share their license keys with others. Only those who accept the *EULA*, *PSLA*, or *CSLA* are allowed to share software that the community develops, and only a separate *OEM Agreement* authorizes the distribution of QNX Neutrino RTOS software in target systems to end users.

QNX doesn't mandate that its community portal be the only development and distribution vehicle for QNX-related products. Licensees may participate in other academic development labs or commercial and non-commercial projects, as long as all the participants are themselves licensed by QNX — under either the free *EULA* and *PSLA* or the royalty-bearing *CSLA* — to use and develop with the QNX development suite. Coordination at the QNX development portal is encouraged but not required.

It is this unique combination of open development and strict licensing controls that excites me about the new QNX business model. I'm looking forward to seeing the embedded systems community adopt and enhance their QNX software to meet their own needs — in an open way — while protecting the proprietary components that are at the heart of the QNX embedded systems business.

*QNX, Momentics, and Neutrino are trademarks of QNX Software Systems GmbH & Co. KG, registered in certain jurisdictions and are used under license. All other trademarks and trade names belong to their respective owners.*