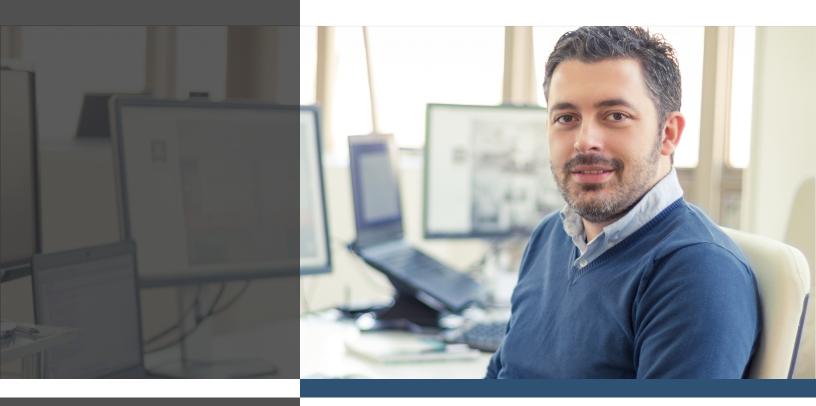# Photon to Qt
MIGRATION
Service

KDAB

Customers invariably put constant pressure on companies to innovate and improve. They want products that incorporate the latest technology to help them grow their business. They want the slick user interfaces that they've come to expect from consumer products. And they want products that are faster, safer, and cheaper.

All of this spells trouble for a company whose products are based on old, unsupported technology. But we can help.

There are always better ways for engineers to improve a company's bottom-line than porting legacy code.

Companies that still use the QNX Photon microGUI in their products — a deprecated and unsupported technology — are trapped using older hardware that's increasingly hard to acquire, unable to update libraries for the latest bug fixes or security patches, and stuck on old versions of the QNX OS that lack new features and performance improvements. They're finding it more and more difficult to get engineers who are trained on Photon, leading to engineering-resource shortages. And they're left with products that have a significantly dated look-and-feel. While a decades-old UX might imply product stability — a benefit in some industries — it also connotes dated functionality, a lack of innovation, vulnerability to cybersecurity attacks, and so on.

WHEN IS IT TIME TO MIGRATE?

If you answer yes to any of the following questions, the answer is now.

- Do you need to update your product to hardware that is no longer supported by earlier versions of QNX?

- Are bug fixes or security patches required in libraries that aren't back-ported?

- Is certifying for security/safety problematic with older, uncertified components?

- Do you need to modernize a UX to match customer expectations?

- Are you redesigning or re-architecting a product and consider it risky to start on an unsupported platform?

- Are you considering moving to a later version of QNX to take advantage of new OS features?

- Do you need to adapt your codebase to handle both real-time systems and mobile/tablet controllers?

- Are you unable to find experienced Photon developers?

KDAB is an expert in migrating Photon software to a modern UX framework based on Qt, and the only QNX-recommended supplier for this process.

We've moved many customers from Photon to Qt, helping them transform their product with a cleanly designed, fully modernized UX – even when dealing with unique Photon features like Ditto, Phindows, or stroke fonts. We are big believers in creating well-architected, quality code that can pass the test of time.

KDAB has three key ingredients that make the porting process straightforward and trouble-free: we have engineers who specialize in Photon, proprietary automation tools specifically to assist in Photon migrations, and

deep Qt expertise in many areas – we are, in fact, very active Qt contributors.

Photon and Qt have enough significant differences to make a Photon to Qt migration complicated and tricky. At KDAB, we know the migration pitfalls and understand the techniques needed to make a port successful – in other words, small, fast, bug-free, and quickly delivered. A KDAB port saves time, experimentation, re-engineering, and allows your engineers to focus on things that add value.

For customers that want to do a Photon to Qt migration themselves, we happily offer workshops to educate staff on migration best practices and provide ongoing mentoring to help when necessary.

And for those customers with a stalled or half-complete migration, we are happy to step in, digest what work remains to be done, clean everything up, and complete the effort.

SIX STEPS FOR A CLEAN PHOTON TO QT MIGRATION

1. **Evaluation**: We start with a free migration evaluation to determine the scope, timeline, complexities, and cost of the project, ensuring that the porting process meets both needs and expectations
2. **Discussion and design**: We discuss any architectural, refactoring, or UX changes, designing the port with these in mind
3. **Porting**: We port over much of the legacy code using a proprietary Photon-to-Qt transformation tool that automates much of the mindless work to save on overall development cost
4. **Revisions**: We hand off the remainder of the port to Qt and Photon experts, incorporating any agreed-to refactoring and design changes, and collaborating with your engineering staff as desired
5. **Testing**: We test the port against our own test scaffolding as well as any additionally required testing
6. **Training**: We hold specialized workshops and/or training as needed to ramp up your engineering staff on the new Qt frameworks and tools

Migration work is never fun – it's boring, detailed, and time consuming. Considering the number of hours it takes engineers to learn through trial-and-error how to move a product from Photon to Qt, using KDAB is a great way to save time, money, and engineers' hair.

About the KDAB Group

The KDAB Group is the world's leading software consultancy for architecture, development and design of Qt, C++ and OpenGL applications across desktop, embedded and mobile platforms. KDAB is the biggest independent contributor to Qt. Our experts build run-times, mix native and web technologies, solve hardware stack performance issues and porting problems for hundreds of customers, many among the Fortune 500. KDAB's tools and extensive experience in creating, debugging, profiling and porting complex applications help developers worldwide to deliver successful projects. KDAB's trainers, all full-time developers, provide market leading, hands-on, training for Qt, OpenGL and modern C++ in multiple languages. Founded in 1999, KDAB has offices throughout North America and Europe.

www.kdab.com

Moving to Qt will open up your product to the benefits of a modern, actively supported, and huge community framework:

- Qt is continually updated through a regular release cadence with feature enhancements and bug fixes
- Qt uses modern development methodologies and embedded-friendly C++, supporting C++11 features (C++14 coming soon)
- Qt 3D provides full 3D support with optional physics-based rendering (PBR)
- Qt has components for graphics, IoT devices, Bluetooth, Sensors, and other peripherals
- Qt supports comprehensive multimedia standards (including DRM)
- Qt Quick and QML provide a powerful but simple scriptable UX
- Qt Creator IDE allows closer collaboration between designers and developers
- Qt offers multi-platform support for QNX SDP 6.6, QNX OS for Safety, QNX OS for Medical, as well as portability to other operating systems like Linux, Windows, Windows Embedded, or Android