myQNX License Manager and QNX Software Center User's Guide

BlackBerry | QNX

BlackBerry Limited
1001 Farrar Road
Ottawa, Ontario
K2K 0B3
Canada

Voice: +1 613 591-0931
Fax: +1 613 591-3579
Email: info@qnx.com
Web: https://www.qnx.com/

Electronic edition published: September 12, 2023

Contents

About This Guide

This guide explains how to use the myQNX License Manager and the QNX Software Center. It is intended for corporate buyers, development managers, and IT administrators who need to manage QNX product licenses, and for developers who need to install and maintain QNX software on their development hosts.

To:	See:
Get an overview of the myQNX License Manager and QNX Software Center	Introduction
Manage and track licensing activity	Using the myQNX License Manager
Add, modify, or remove product installations on your development host	Using the QNX Software Center
Perform seldom-required tasks, or address connectivity or product activation issues	QNX Software Center Advanced Topics
Use the command line to add, modify, or remove product installations	Using the QNX Software Center Command Line
Generate reports that list the properties of files found in an installation or a target image	Manifest Reports & File Lookup

If you're a developer, this guide tells you how to use the QNX Software Center to install the QNX Software Development Platform (SDP) 7.* and related products. Once you've installed QNX SDP, see the *QuickStart Guide* on the QNX website, which provides an introduction to developing QNX-based applications.

Typographical conventions

Throughout this manual, we use certain typographical conventions to distinguish technical terms. In general, the conventions we use conform to those found in IEEE POSIX publications.

The following table summarizes our conventions:

Reference	Example
Code examples	`if( stream == NULL)`
Command options	`-lR`
Commands	`make`
Constants	`NULL`
Data types	`unsigned short`
Environment variables	*PATH*
File and pathnames	**/dev/null**
Function names	*exit()*
Keyboard chords	**Ctrl–Alt–Delete**
Keyboard input	`Username`
Keyboard keys	**Enter**
Program output	`login:`
Variable names	*stdin*
Parameters	*parm1*
User-interface components	**Navigator**
Window title	**Options**

We use an arrow in directions for accessing menu items, like this:

• You'll find the Other... menu item under **Perspective → Show View**.

We use notes, cautions, and warnings to highlight important messages:

Notes point out something important or useful.

CAUTION: Cautions tell you about commands or procedures that may have unwanted or undesirable side effects.

DANGER: Warnings tell you about commands or procedures that could be dangerous to your files, your hardware, or even yourself.

ALERT: Alerts tell you about commands or procedures that could expose private information or otherwise compromise security.

Note to Windows users

In our documentation, we typically use a forward slash (/) as a delimiter in pathnames, including those pointing to Windows files. We also generally follow POSIX/UNIX filesystem conventions.

Technical support

Technical assistance is available for all supported products.

To obtain technical support for any QNX product, visit the Support area on our website (https://blackberry.qnx.com/en/support). You'll find a wide range of support options.

Chapter 1
Introduction

Together, the myQNX License Manager and the QNX Software Center make it easy to license, download, install, and maintain QNX software products. It doesn't matter whether your company is a small development shop with a handful of developers or a large OEM with globally distributed software teams. Either way, these tools will maximize the efficiency of your license administrators and enable all of your developers to stay up to date with the latest QNX software tools and components.

Traditional approaches to the licensing and deployment of embedded software platforms rely on specialized knowledge and manual, error-prone procedures. The myQNX License Manager and QNX Software Center replace these approaches with graphical tools that take the guesswork out of managing large license pools and complex software dependencies. They also make it easier to cascade new software products, updates, or patches throughout an organization—and to install those products quickly and securely.

Centralized management of product licenses

The myQNX License Manager provides a single, integrated view into the QNX licensing activity in your organization, enabling you to take a proactive approach to license management and renewal. You can:

- see at a glance the QNX products that your company has licensed
- dynamically monitor the deployment, activation, utilization, and deactivation of products
- maximize the use of purchased licenses and thereby reduce the need for overlicensing
- anticipate when support plans are set to expire, to ensure timely renewals and uninterrupted productivity

When you log in to the myQNX License Manager, you can immediately take control of common licensing activities. Perform any of the following actions with a few mouse clicks:

- allocate administrator privileges to other managers
- deploy licenses to developers
- transfer licenses between developers
- view an interactive timeline of product activations
- view the license certificate for any product
- check the status of support plans and their associated products

Simple, secure installation of software products

Think of the QNX Software Center as your one-stop shop for discovering, installing, and managing all of the QNX software that you are entitled to. Besides making it easy to install QNX products, the center tells you when new software packages or updates to existing packages become available. From there, it takes just a few clicks to install the packages you want.

The QNX Software Center automatically manages product dependencies. You don't have to install software packages by hand, in a specific order or location. Everything works together, out of the box.

The QNX Software Center lets you create detailed reports of the QNX files and subsystems used in your target images. These reports make it easier to manage your runtime deployments, accurately and efficiently.

The QNX Software Center is designed with security in mind. Every package is digitally signed, to guarantee its integrity and to protect your existing installation.

The QNX Software Center is multiplatform, with support for Linux, Microsoft Windows, and macOS hosts. So all of the developers in your organization can use the same efficient tool to manage their QNX installations and to stay current with the latest patches and updates.

What's more, the QNX Software Center lets you create *patch sets*: specific configurations of QNX products, patches, and updates that can be shared by some or all members of a development team. This feature makes it much easier to reproduce issues and enables teams to standardize on the same software base.

Chapter 2
Create Your myQNX Account

To use the myQNX License Manager or the QNX Software Center, you must first create a myQNX account. Setup is easy and takes only a few minutes.

To create your account:

1. Go to **www.qnx.com** and click **Sign In** at the top of the page.

2. Click **New Member**.

3. Fill in your email address and choose a password, then click **Create Account**. (Your password should contain only alphanumeric characters.)

4. Fill in your profile information, then click **Submit**.

5. Check your email for a confirmation key, and then use one of the following methods to activate your account:

   • Click the activation link in the confirmation email.

   • On the account creation page, enter the key and your email address, and then click **Continue**.

You are redirected to the login screen where you can log in to your myQNX account profile.

Chapter 3
Important Terms and Definitions

This section covers important definitions that can help you better understand how to use QNX licenses.

Terminology

Development seat

Refers to any license type meant for development

Host/Host machine

Refers to a developer's machine that has QNX SDP installed on it

Build server

Refers to any machine which has components of QNX SDP installed and is used exclusively for automated QNX builds

License model types

Below is a quick overview of each license type. To get an in-depth explanation, see the *QNX Product Portfolio Guide* (http://www.qnx.com/qnx_sdp_product_guide_71).

License model types for development

Named User Developer License

This is a standard development license seat. Issued to a user who is named and identified by their email address.

Floating Developer License

Requires a floating license server. Floating license seats may float between developers (multi-user mode) provided each floating license is tied to a single developer at any given time.

For instruction on how to deploy and set up Floating Developer Licenses, see the *Floating License and Continuous Integration Build Server License Management Guide*.

License model types for automated builds on build servers

Floating Build Server License

This license allows for QNX automated builds on build servers using floating licenses. It uses the same floating license server for both development and build servers. Each floating license is tied to a single developer or build server at any given time. Multiple builds on a build server require multiple floating licenses.

For instruction on how to deploy and set up Floating Build Server Licenses, see the *Floating License and Continuous Integration Build Server License Management Guide*.

Continuous Integration Build Server License

Allows for QNX automated builds on build servers using floating license technology. It requires its own floating license server. Each license is tied to a single build on the build server at any given time. Multiple builds on a build server require multiple licenses.

For instruction on how to deploy and set up Continuous Integration Build Server Licenses, see the *Floating License and Continuous Integration Build Server License Management Guide*.

Standard Build Server License

Uses a Named User Developer License on a build server. Developers require a Named User Developer License.

Chapter 4
Using the myQNX License Manager

The myQNX License Manager lets you take centralized control of the QNX licensing for your organization. It reduces the hard work of managing licenses for large software teams to a few simple tasks that you can perform from a single, easy-to-use control panel:
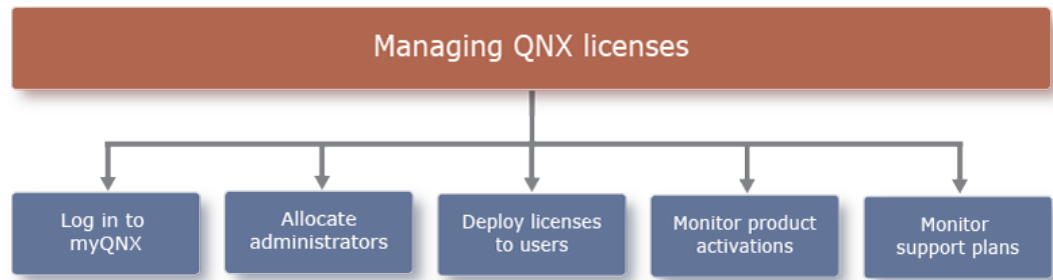
Accessing the license manager

To access the myQNX License Manager, visit https://www.qnx.com/account/dashboard/ or go to your myQNX profile and click the Manage Registered Products link.

If you are logged out of your myQNX account when you attempt to access the myQNX License Manager, you'll be prompted to enter your myQNX credentials. If you don't have an account, see "Create Your myQNX Account."

Accept licenses

The myQNX License Manager enables a flexible approach to license management by allowing you to easily view licenses and accept them.

Corporate buyers with administrator rights or development managers can accept licenses to deploy products. Any new product deployments are gated until the license is accepted. Upgrades are generally included in the agreement and do not require additional license acceptance.

Accept procedure

To accept a license for a product:

1. Open the myQNX License Manager: https://www.qnx.com/account/dashboard/
2. Under **Accept License** field, click **View License Areement**.
3. Review the **License Manager Product Acceptance Agreement**.
4. Click **Access BlackBerry QNX Licensing**.
5. From the **BlackBerry QNX Licensing** page, click the "document archive" hyperlink.
6. Review the table of Development License Agreements.

Back in myQNX License Manager, you can now move forward with accepting the license.

1. Click **Accept License**. Message Center pops up.
2. Click **I accept**.
3. Click **Acknowledge**.

Allocate licenses to administrators

The myQNX License Manager enables a flexible approach to license management by supporting up to 20 administrators for each product serial number. For instance, administrators A, B, and C can manage product 1; administrators A, E, and F can manage product 2; administrators A, D, and G can manage product 3, and so on.

If you're a corporate buyer with administrator rights, this flexibility makes it easy to delegate day-to-day license management to development managers or IT staff. If you're a development manager, this same flexibility lets you share your administrator rights with colleagues who can serve as your backup.

> We recommend that at least one person have administrator rights to all of the products that your company licenses. This approach ensures a global view of all licensing activities. You can subsequently transfer this responsibility to another individual, should the need arise.

Allocation procedure

To allocate administrator rights for a product:

1. Open the myQNX License Manager: https://www.qnx.com/account/dashboard/
2. Click the product's **Managers** field, then click **Allocate**.
3. Enter the email addresses of up to 20 managers, then click **Save**.

   Each address must be associated with a valid myQNX account. You can enter the addresses in multi-line or comma-delimited format.

To allocate administrator rights for multiple products all at once:

1. In the first column on the left, click the checkbox of each product that you want to allocate. To select all products, click the checkbox at the top of the column.
2. Click the dropdown icon at the top of the column, then select **Allocate Selected Products to Managers**.
3. Enter the email addresses of up to 20 managers, then click **Save**.

When you allocate administrator rights for a product:

- The manager will see the product when they log in to the myQNX License Manager. The manager can then begin to deploy licenses to users.
- The manager will, within a few minutes, receive an email notification.

> **ALERT:** When you allocate administrator rights to a myQNX account, the owner of that account can subsequently allocate administrator rights to any other myQNX account. Also, any administrator for a product can deallocate the rights of any other administrator for that product.

Deallocation procedure

To deallocate administrator rights for a product, whether your own or those of another manager:

1. Click the product's **Managers** field.

2. Click **Deallocate** next to the email of the manager you want to deallocate.

You can deallocate administrator rights for a product only if you are a manager for that product.

Deploy licenses to users

The task of deploying licenses takes only a few simple steps, even when you are deploying multiple licenses all at once.

In this section, we define package status and discuss how to deploy:

- named user licenses
- build server licenses
- floating multiuser licenses
- project licenses

Named user licenses

To deploy a named user license to a user:

1. Look at the product's **Status** field to make sure the license is Available.
2. Click the product's **Users** field, then click **Deploy**.
3. Enter the user's email address, then click **Save**. The address must be associated with a valid myQNX account.

To deploy multiple named user licenses to a user all at once:

1. In the first column on the left, click the checkbox of each product that you want to deploy. To select all products, click the checkbox at the top of the column.
2. Click the dropdown icon at the top of the column, then select **Deploy Selected Products to Users**.
3. Enter the user's email address, then click **Save**.

When you deploy a named user product license:

- The product's **Status** field changes to Deployed, indicating that the user can now install the product through the QNX Software Center.
- The user's myQNX profile will display the product serial number and its associated license key.
- The user will, within a few minutes, receive an email notification.

> 💡 You can deploy a named user license to only one user.

Build server licenses

The procedure for deploying a build server license is similar to that for deploying a named user license: click the product's **Users** field, click **Deploy**, and then enter the user's email address.

Make sure that you deploy the build server license to the same user who will create the build server installation and run the builds.

To write an automated deployment script that creates a build server installation, see the *Creating and Deploying Installations of QNX SDP in Large Organizations* guide.

Floating multiuser licenses

To deploy a floating multiuser license to one or more users:

1. Click the product's **Users** field, then click **Deploy**.

2. Enter the users' email addresses, then click **Save**. Each address must be associated with a valid myQNX account. The addresses can be in multi-line or comma-delimited format.

To deploy multiple floating licenses all at once:

1. In the first column, click the checkbox of each product. To select all products, click the checkbox at the top of the column.

2. Click the dropdown icon at the top of the column, then select **Deploy Selected Products to Users**.

3. Enter the users' email addresses, then click **Save**.

When you deploy a floating license:

- The **Users** field changes to show the number of users to which the license is now deployed.
- Each user's myQNX profile will display the product serial number and its associated license key.
- Each user will, within a few minutes, receive an email notification.

For more information on administering floating multiuser licenses, see the *Floating License Management Guide* (http://www.qnx.com/download/group.html?programid=30391). The guide provides an introduction to floating licenses; it also explains how to configure a license server and how to connect development hosts to that server.

Project licenses

The procedure for deploying a project license to multiple users is similar to that for deploying a floating license to multiple users: click the product's **Users** field, click **Deploy**, and then enter the users' email addresses.

You can deploy a project license to all developers working on the project.

Package status

The Status (See status) field identifies the maturity of the software at a file level, although it is usually defined at package level so that all included files have the same maturity.

Possible values of this field can include **Experimental**, **Alpha**, **Beta**, **Validated**, **Official**, **Stable** and **Experimental-Pending Stable**.

The identified software maturity level may have specific meanings and ramifications detailed in the terms and conditions of your software license. For details please consult the license agreement(s) pertaining to your use of the relevant product.

See also "Inspect package and file properties" and "Manifest Reports & File Lookup".

Experimental packages can be excluded from the installation or from the calculation of the dependencies by modifying the installation preferences. See "Edit installation properties". All packages are considered experimental unless marked Stable, Experimental - Pending Stable, or Official.

Deploy licenses offline

If you need to deploy or allocate licenses to many users or administrators, you can use the Export feature to specify the deployments offline, in an easy-to-use spreadsheet. You can then use the Import feature to apply all of the deployments at once, with just a few mouse clicks. The deployments take effect only when you import them.

To deploy or allocate licenses offline:

1. Click the **Export** button, then click **Excel**. The myQNX License Manager will create a Microsoft Excel file that contains several columns (Serial Number, Users, Managers, etc.) for every product.

2. To deploy or allocate a license, find its serial number, then enter the email address in the corresponding **Users** or **Managers** column.

   If you need to enter multiple addresses in a cell, separate them with commas.

3. Save the file when you're done.

Note that the Excel file contains a record for every product that matches the current contents of the license manager's **Search** field. If you want the file to include all of the products you manage, make sure the field is blank before you click **Export**.

⚠ **CAUTION:** Don't change the column structure of the Excel file. Otherwise, importing the file will fail.

⚠ **CAUTION:** The exported file doesn't contain product collection information, so we recommend that you avoid using it to deploy or allocate products that belong to product collections. If you do wish to use the Export feature for product collections, make sure that your assignments follow the product-collection rules described in "Access multiple versions of a product."

To import and apply the deployments or allocations you specified in the Excel file:

1. Click **Import**.

2. Click **Choose File**, select the Excel file, then click **Next**. You'll see a **Confirm Data** window.

3. Review your deployments, then click **Next**.

4. Click **Import** to apply your changes. You'll see a summary of your changes, along with any error messages.

5. Click the **X** in the upper-right corner of the Import window to return to the myQNX License Manager.

The Import feature enforces all the rules that apply to license deployments and allocations. For instance, it:

- deploys a named user license to a user only if that license isn't already deployed
- accepts an email address only if the address is associated with a valid myQNX account
- deploys or allocates a license to a user or administrator only if you have administrator rights to that license at the time of import

If a deployment breaks any of these rules, the Import feature ignores it and displays a corresponding error message.

Currently, you can't use the Export and Import features to revoke licenses or deallocate administrators.

Access multiple versions of a product

The myQNX License Manager has a *product collection* feature that enables users to access multiple versions of certain products.

Using this feature, you can request access to the newer or previous version of a product. In response, the license manager will create a collection that contains the requested version as well as your current version.

The user of a product collection gains access to all product versions in the collection. For example, if you are managing a QNX SDP 7.0 product and request an upgrade to release 7.1, the two versions will form a product collection, and the user can install, activate, and use both versions.

The product collection feature helps simplify renewal of support plans. For example, in the above scenario, renewing the support plan for the 7.1 product version will also renew the support plan for the 7.0 product version.

You can issue a product collection request for a product either before or after you deploy the product to a user. In fact, you can issue the request regardless of whether the product status is Available, Deployed, or Activated.

To determine whether a product is supported by the product collection feature, look at the product's **Available Actions** field. The field will contain an **Upgrade** button or a **Get Previous Version** button.

Before you begin...

Before you issue a product collection request for a product, note the following:

- *You can make product collection requests for several products all at once.*

  If you have several eligible products, we recommend you select all of them and then issue a single product collection request, as described below. Upon approval, each product will form its own collection.

- *The product collection feature isn't available for all products.*

  The product collection feature is available only to certain products covered by named user, floating, or enterprise development seat licenses. Also, having the entitlement to access a newer version of a product doesn't necessarily mean you also have the entitlement to a previous version. Contact your sales representative if you'd like more information on which products are eligible.

- *Products within a collection function as a group.*

  The user of a product collection gains access to all product versions in the collection. Thus, you cannot split a collection across users. For example, if a collection contains two product versions, QNX SDP 7.0 and QNX SDP 7.1, you cannot deploy the 7.0 version to one user and the 7.1 version to another user. Also, if any product in a collection has been deployed to multiple users, those same users have access to all products in the collection.

- *You can create a product collection only if the product has an active support plan.*

  If the support plan is inactive (for instance, it has expired), the product's **Available Actions** field will display a button that prompts you to contact a QNX sales representative. Once the plan has been renewed, an **Upgrade** or **Get Previous Version** button appears in the **Available Actions** field;

you can then issue your product collection request. The button typically appears within one business day after the renewal has been processed.

- *Creating a collection for a product governed by floating licenses doesn't change your total number of floating license keys.*

  For example, if you have 10 QNX SDP products governed by floating licenses and upgrade all 10 products, the number of floating licenses will still be 10.

  If required, you can increase the size of your floating license key pool by purchasing more floating licenses. See the *Floating License Management Guide* for more information on floating licenses (http://www.qnx.com/download/group.html?programid=30391).

- *If a product collection governed by a named user license contains activated products, the developer must deactivate all of those products before the collection can be revoked.*

  Suppose, for example, that a collection governed by a named user license contains QNX SDP 7.0 and QNX SDP 7.1. If the user has installed and activated the 7.0 version on one machine and the 7.1 version on another machine, the user must explicitly deactivate the installation on each machine before the product collection can be revoked. For more information, see "Revoke products," "Transfer a named user license from one user to another," and "Transfer a named user license to another user or workstation."

- Perpetual and Subscription development seats for QNX SDP versions 7 and later do not require registration of the support plan.

  The developers or license managers to whom a license is deployed or allocated, receive access to the support plan associated with the license. Both the types of development seats behave in the same way for the support portal and allow access to it, provided the support is active.

  - **Perpetual development seat**: The license for QNX SDP 7.1 is automatically appended to the existing support plan.

    Expiration- When the standard support plan expires, it is no longer associated to that license. The technical support portal and updates are no longer accessible.

    Renewal- In case of Perpetual products, the renewal allows you to retain the same support plan provided it was renewed under that same plan.

    If your Perpetual development seat is renewed under a different support plan, then it is through this plan that you will access the technical support portal.

  - **Subscription development seat**: The QNX SDP 7.1 license expires at the same time as the QNX SDP 7.0 license.

    Expiration- When the subscription expires, the support plan expires too, and the subscription becomes unusable. The subscription is removed from the **Registered Products** and placed in the **Expired Products** of a myQNX account.

    Renewal- The license manager will receive a new subscription license along with a new support plan.

Making a product collection request

> 💡 Before you begin, we recommend you select "Show All entries" so you can see all of your eligible products at a glance.

If a product is eligible, its **Available Actions** field will display an **Upgrade** or **Get Previous Version** button.

To submit a request for a single product, simply click the button.

To submit a request for multiple products all at once:

1. In the first column on the left, click the checkbox of each desired product. To select all products, click the checkbox at the top of the column.

2. Click the dropdown icon at the top of the column, then select **Upgrade** or **Get Previous Version**.

In response, the **Available Actions** field displays an "in progress" message and the request is issued to BlackBerry QNX. When BlackBerry QNX processes the request—typically within two business days—the following changes will occur:

• The "in progress" message changes to "Upgrade approved" or "Previous version approved". You may need to refresh the myQNX License Manager page to see this change.

• The multiple versions of the product will appear together in a collection.

• Any user and license administrator for the product will receive an email notification that includes the serial number of the added product version. In addition, the Registered Products section of the user's myQNX account profile will change to show information such as the product collection number and the serial number of the added product version.

• The user can now install either product version, using the QNX Software Center. To see the package for the added product version, the user may need to relaunch or refresh the QNX Software Center.

> 💡 Remember that the processing of a product collection request can take two business days. A user can't access the requested product version until the process is completed.

Monitor product activations and support plans

You can view the activation history of any product, including when it was deployed, activated, deactivated, or revoked. You can also keep tabs on the renewal dates for all of your product support plans.

Product activations

To monitor which products have been activated:

For:	Do this:
Named user products	Look at the **Status** field. If the status has moved from Deployed to Activated, the user has installed the product through the QNX Software Center.
Multiuser products	Look at the **Users** field. It displays the number of deployed users (in gray) and activations (in orange).

To view the activation history of a product:

1. Click on the product's serial number.
2. Scroll down to the **History Log** section and select one of the following options:

Select:	To:
List View	View a history log for the product, sorted by date.
Timeline View	View an interactive timeline of the product's history. You can: • Change the date range of the timeline by using the **Show Now**, **Default**, **+**, **-**, **<**, and **>** controls. • Use the scroll wheel on your mouse to zoom in or out of the timeline. Just hover over the timeline and turn the wheel. • View information about any event by clicking the circle that represents the event.

Support plans

To monitor support plans:

• Look at the **Available Actions** column—shows you at a glance when the support plan for any product needs renewal. You can click the column title to sort products by their renewal dates.

• Select the **Support Plans** tab—displays all of your support plans, which you can sort according to plan type, plan number, serial number, and renewal date.

Revoke products

When you revoke a user's license for a QNX product, the user loses their entitlement to use the product, and the myQNX server removes the product from the list of registered products in the user's myQNX account.

Named user products

> You cannot revoke a product governed by a named user license if the product is currently activated. See "Deactivate products governed by a named user license" to learn about product deactivation.

To revoke a named user product:

1. Look at the product's **Status** field to make sure the license status is Deployed.
2. Click the product's **Users** field.
3. Click **Revoke** next to the user's email address, then click **Yes** to confirm.

   The **Status** field changes to Available. (If the license is already Available, there's no need to revoke it.)

Multiuser products

To revoke a product governed by a multiuser license:

1. Click the product's **Users** field.
2. Click **Revoke** next to the user's email address, then click **Yes** to confirm.

To revoke a multiuser product for several users all at once:

1. Click the product's **Users** field.
2. Use the checkboxes to select the users, then click the dropdown icon and click **Revoke Selected**.
3. Click **Yes** to confirm.

Deactivate products governed by a named user license

If a product is governed by a named user license, the user may, in some circumstances, need to deactivate it. For instance, the user must perform deactivation when upgrading to a new workstation or when the license administrator needs to transfer the license to another user.

Standard deactivation method

To deactivate a product governed by a named user license, the user should always try to use the QNX Software Center. Here is the deactivation procedure:

1. Launch the QNX Software Center.
2. From the Welcome screen, select **Manage License Keys**.
3. Right-click the product's license key and select **Deactivate**, then click **OK** to confirm.

The deactivation disables product features and removes the user's entitlement to use the product on that workstation.

If the user has installed components of a product collection across multiple machines, they must explicitly deactivate the component residing on each machine. Otherwise, the product collection will remain activated.

Suppose, for example, that the user has a product collection consisting of QNX SDP 7.0 and QNX SDP 7.1. If the user installed the 7.0 product version on one machine and the 7.1 product version on a different machine, they need to deactivate the product version on each machine.

This requirement applies only to collections governed by a named user license. See "Access multiple versions of a product" for more information on product collections.

If, for some reason, the user can't use the QNX Software Center (for instance, the user's workstation has been damaged or lost), the user can perform the deactivation from their myQNX account profile. The profile page will display a Manual Product Deactivation link if any of the user's products are currently activated.

If deactivation from myQNX isn't possible (for instance, the user no longer works for your company), you must submit a deactivation request to BlackBerry QNX. See the instructions, below.

Submitting a deactivation request

To submit a deactivation request:

Follow these steps only if the user can't deactivate the product through the QNX Software Center or through their myQNX account.

---

1. Open the myQNX License Manager.
2. Look at the product's **Status** field to make sure that the status is Activated.
3. Click the product's **Users** field, then click **Deactivate** next to the user's email address. A popup dialog will ask you to provide the reason for the deactivation request.

4. Enter the requested information and submit your request. The myQNX License Manager will send the request to BlackBerry QNX and cc you via email.

Only products covered by named user licenses support deactivation.

> If a product is part of a product collection, you will need to submit a separate deactivation request for every activated product in the collection.

Transfer a named user license from one user to another

Transferring a named user license consists of a few simple steps.

To transfer a name user license from one user to another:

1. Make sure that the existing user has deactivated the license, using the QNX Software Center. The deactivation procedure is described in "Transfer a named user license to another user or workstation."

   A deactivated named user license will show as Available in the myQNX License Manager.

2. Revoke the license, following the steps described in "Revoke products."

3. Deploy the license to the new user, following the steps described in "Deploy licenses to users."

License transfers aren't required for products covered by a multiuser license.

Use live search to locate items

Speed up item searches by using the **Search** field of the myQNX License Manager. As you type, the table of products or support plans will dynamically refresh to display any records that match your query.

You can search for products, serial numbers, user names, manager names, ship dates, renewal dates, and so on.

Clear the **Search** field to revert to the default display.

Understand development licensing

We recommend that you download the *QNX Product Portfolio Guide* (http://www.qnx.com/qnx_sdp_product_guide_71) and review the section on development licensing, which provides an overview of the licensing models that apply to software development. To access this document, you need a myQNX account and QNX SDP 7.* product registration.

If you need to administer floating licenses, see also the *Floating License Management Guide* (http://www.qnx.com/download/group.html?programid=30391). The guide provides an introduction to floating licenses; it also explains how to configure a license server and how to connect development hosts to that server.

Chapter 5
Using the QNX Software Center

The QNX Software Center provides a secure, efficient way to discover, install, and manage software products distributed by BlackBerry QNX. It replaces traditional command-line procedures with a graphical user interface that takes the guesswork out of managing product dependencies. It also lets you create *patch sets*: specific configurations of QNX products, patches, and updates that enable development teams to reproduce issues and standardize on the same software base.

With the QNX Software Center, managing product installations consists of a few, easy-to-perform tasks:
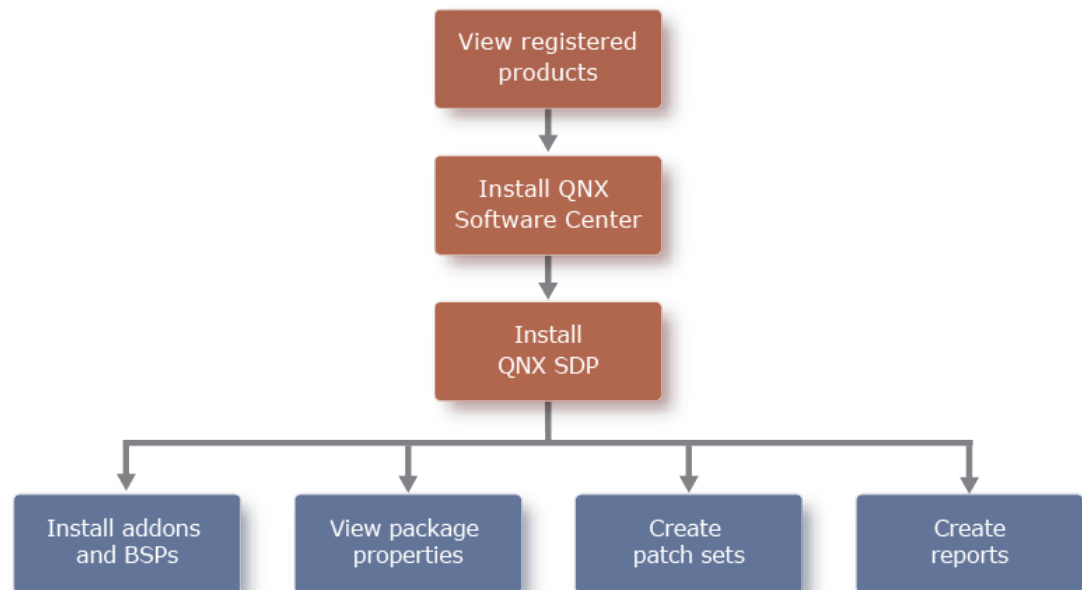
View your registered products

Before you begin using the QNX Software Center, visit your myQNX account profile to view the products that have been deployed to you. The profile includes a Registered Products section, which shows at a glance the license key, serial number, license certificate, and license administrators for each deployed product. If the products you need to install don't appear in your profile, contact your license administrator.

To access your myQNX profile, visit https://www.qnx.com/account/index.html.

Using QNX SDP 6.6 or earlier?

The QNX Software Center lets you download and install the QNX Software Development Platform (SDP) 7.* and related QNX products. If you have registered products based on QNX SDP 6.6 or earlier, you can access those products from the download center on the QNX website:
https://www.qnx.com/download/

Install and launch the QNX Software Center

To install the QNX Software Development Platform (SDP) 7.* and any of its associated packages, you must first install the QNX Software Center onto your workstation.

💡 You must be a registered QNX SDP 7.* user to download the QNX Software Center.

To install and launch the QNX Software Center:

1. Log in to your myQNX account: https://www.qnx.com/account/index.html
2. In your myQNX account profile, click the QNX Software Center link:
   http://www.qnx.com/download/group.html?programid=29178
3. Scroll down and click the installer link for your host OS.
4. Scroll down to **Related Documents** and open the installation note, which describes how to run the QNX Software Center installer. (The **Related Documents** section also includes release notes for the QNX Software Center.)
5. Install the QNX Software Center according to the installation note.

💡 Upon launching, the QNX Software Center may display a warning that it cannot contact the myQNX server. This warning typically appears when your workstation sits behind a proxy server. To update your proxy settings and gain access to the myQNX server, see "Configure your proxy settings."

Log in with your myQNX credentials

When the QNX Software Center connects to the myQNX server, it prompts you to enter your myQNX credentials:

Choose **Save password to disk** if you want to log in automatically when you subsequently launch the QNX Software Center. See "Manage secure storage of your passwords" for more information.

Install the QNX Software Development Platform

After you log in, the QNX Software Center displays a Welcome screen that lets you install the QNX Software Development Platform (SDP) and other QNX products.
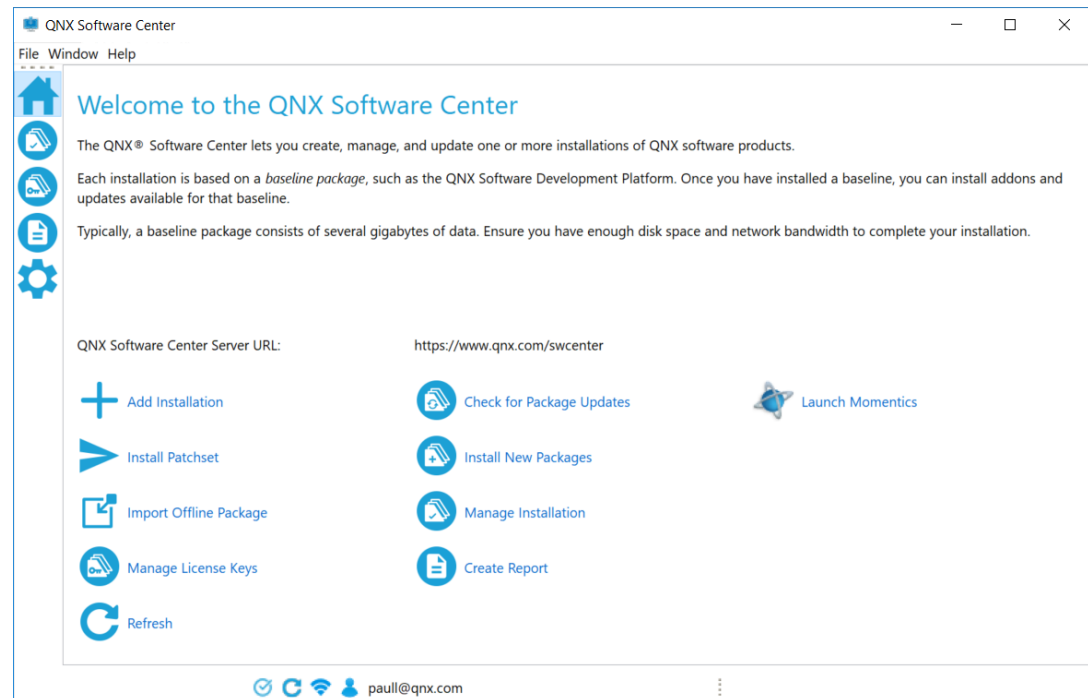
You can use the following installation methods:

Standard installation

Simply click **Add Installation** and follow the instructions in "Add an installation."

Offline installation

If you have to use the QNX Software Center offline, you can obtain packages for offline installation. See the instructions in "Import packages or package archives from your local filesystem."

Patch sets

You can install QNX SDP by using patch sets, which determine not only which software packages to install, but also which version of each package to install. Patch sets are useful, for example, when all members of a development team need to use the same versioned installation.

A project administrator or other team member may have defined a patch set for you to import. If so, see the instructions in "Import a patch set to add an installation."

> You can return to the Welcome screen at any time by clicking the Home icon in the shortcut bar or by selecting **Help → Welcome**.

Licenses

You need a valid license key to install QNX SDP. In most cases, the QNX Software Center automatically downloads the key that your license administrator has deployed to you.

If your license for using QNX SDP is a floating multiuser license, you need to set up a connection between your workstation and the license server. See the Floating License Management Guide for details. (The Registered Products section of your myQNX profile indicates the license type of each of your deployed products.)

Add, choose, or remove installations

The QNX Software Center lets you create and maintain multiple installations of the QNX Software Development Platform (SDP) on your workstation. Each installation serves as a container for a set of QNX software packages, and each lives in its own directory, where you can modify it independently of any other installation. For instance, you can create one installation for production purposes, another for experimental purposes, and so on. You can install copies of the same addon package in multiple installations.

Add an installation

The following steps assume that the QNX Software Center is online and able to communicate with the myQNX server. If you need to perform offline installation, see "Import packages or package archives from your local filesystem."

💡 Before adding an installation, make sure that your system has at least 4 GB of memory.

To add an installation:

1. Select **Add Installation** from the Welcome screen or the **Advanced** tab.
2. Select the product that you want to install.

   For instance, to install QNX SDP 7.0, you would expand the **QNX Software Development Platform 7.0** group, then click the **QNX Software Development Platform 7.0** package.

   💡 To read the product release notes before proceeding with the installation, right-click the package, then select **Properties → Release Notes**.

   ---

3. Click **Next** when you're ready to proceed with the installation.
4. Choose a license key for the product being installed, then click **Next**.

   If you need help with choosing the appropriate license, consult with your license administrator or visit the licensing information page.

5. Specify the installation folder, name, and other properties of your installation, then click **Next**.

   For help, see "Edit installation properties" in this guide.

   The QNX Software Center may take a few minutes to calculate installation requirements.

6. Make sure that the item you want to install is checked, then click **Next**.
7. Review the list of packages that the QNX Software Center will install, then click **Finish** to start the installation process. The process may take a few minutes to complete.

   Too little disk space?

   If the **Review Packages** dialog warns that you have too little disk space, don't click **Finish**. Instead, click **Back**, free up some disk space, and then click **Next** to return to the **Review Packages** dialog. If necessary, repeat this procedure until the dialog no longer issues the warning.

> When installation is complete, the QNX Software Center may display a warning that it couldn't activate your product. If so, you may need to activate the product manually. For help, see "Perform manual product activation."

Viewing the progress of your installation

By default, the QNX Software Center displays a dialog that allows you to track the progress of your installation or to cancel the installation, if required. If you dismiss the dialog (e.g. by clicking **Run in Background**), you can still access these functions, as follows:

- To view the installation progress, click the Progress button in the lower-right corner of the main window:

- To cancel the installation, click the progress bar's Cancel button:

> **CAUTION:** Wait for the installation process to complete before attempting to add another installation. Otherwise, the installation wizard may freeze and the installation in progress may not complete successfully.

Viewing your completed installation

Your new installation will appear in the **Installed** tab. It will also appear in the QNX Momentics IDE, if the IDE is installed, as a new SDK. To select the SDK, launch the IDE, click **Window → Preferences** in the IDE, and then expand the **QNX** item. (You may need to restart the IDE to see the SDK.)

Choose an installation

To choose an installation on a development host that has multiple installations:

1. Go to the main window of the QNX Software Center (for instance, by choosing **Manage Installation** from the Welcome screen).

2. Click the installation selector, which appears in the top-right area of the window. Here's an example of the selector:

   ⬥ qnx700   ⌄

3. Click the installation you want to view, modify, or remove.

Remove an installation

> **CAUTION:** When you remove an installation, the QNX Software Center deletes all of the files in the installation.

To remove an installation:

1. From the installation selector, select the installation you want to remove.

2. Click the **Advanced** tab, click **Remove Installation**, and then click **OK** to proceed with the removal.

3. To confirm that the removal succeeded, click the installation selector to see the dropdown list of current installations. The installation you removed should no longer appear in the list.

Note the following:

- If you remove the last installation on your workstation and exit the QNX Software Center before removal is complete, a large amount of content may be left uninstalled on your disk. If this issue occurs, you'll need to manually remove the installation directory and its contents. To avoid the problem, wait until the progress indicator and the "Remove Profiles" message disappear. Also, avoid exiting until you confirm that the installation directory is either empty or deleted.

- If you remove the only installation of a product on your machine, and the product is governed by a named user license, the QNX Software Center prompts you to deactivate the product. If the machine contains other installations of the product, the product remains activated, unless you explicitly deactivate it in the License Key perspective (you can access this perspective from the Welcome screen).

If you need to uninstall the QNX Software Center and all of your installed products, see the QNX Software Center *Installation Note*.

Advanced installations

Empty installations

Empty installations allow the installation of individual packages in a separate directory, without requiring performing a complete installation.

To create an empty installation, simply click **Add Installation** on the Welcome screen, and select the **Empty installation** checkbox option in the **New Installation Wizard**. This will allow you to create an empty installation based on the selected baseline within the chosen directory. Afterwards, individual packages may be chosen to be installed.

> Any dependencies related to the selected baseline installation will be resolved during an empty installation, which may result in more packages being installed than expected.

Phantom installations

To avoid having to install all the dependencies that come along with a selected package, you may choose to install Phantom packages. These are packages that are considered to be installed but are not actually on disk. This method also provides a way to perform quick installation checks without having to actually install anything.

To perform a baseline Phantom installation, initiate a regular installation but check the **Phantom Installation** check box after selecting your installation directory. Once a baseline Phantom installation is complete, you may install individual packages on disk by right-clicking on the package and selecting the **Repair** action, or by installing new packages from the **Available** tab.

Download only

Another way to avoid dependency resolution is to download individual packages by right-clicking on an available package and selecting the **Download Only** action.

Once the package has been downloaded this way, QNX Software Center does not manage it anymore. To install the package, unpack it using the `tar` command.

Install the QNX Momentics IDE

If you intend to develop applications for the QNX Neutrino RTOS, we recommend that you install the QNX Momentics IDE. The IDE extends the capabilities of the QNX Software Development Platform (SDP) by providing a graphical environment for developing, debugging, and profiling applications.

Installing the IDE

The procedure for installing the QNX Momentics IDE is similar to that for installing QNX SDP.

To install a completely new Momemtics application or a second copy in a different location:

1. Select **Add Installation** from the Welcome screen or the **Advanced** tab.

2. Expand the **QNX Momentics IDE** group, then click **QNX Momentics IDE**.

> To read the product release notes before proceeding with the installation, right-click the package, then select **Properties** → **Release Notes**.

3. Click **Next**.

4. Accept the default installation folder, name, and description, or modify them according to your requirements.

5. Ignore the settings for the update policy, debug symbols, experimental packages, and target architectures—they don't apply to the IDE.

6. Click **Next**.

7. Make sure that the **QNX Momentics IDE** item is checked, then click **Next**.

8. Review the list of packages that the QNX Software Center will install, then click **Finish** to start the installation process.

Upon completion of the installation process, the new installation of the IDE becomes your current installation and appears in the **Installed** tab.

Updating the IDE

If you install anything (e.g., third party plugins) into Momentics Installation, it is modified. To preserve your installation, use the update method below:

- To update a base installation to the latest one, use the Momentics (Update Only) package which appears in the **Available** tab of QSC.

- To update the existing installation which is already an update, use the Momentics (Update Only) package which appears in the **Updates** tab of QSC.

> If you did not modify your installation, use the **New Installation** method instead of the **Update** method. Then, remove the old installation.

Installing Momentics plugins into another Eclipse instance

Use the Update Only package to install Momentics plugins in a Non-Momentics Eclipse based IDE.

Launching the IDE

To learn how to start and use the IDE, download the *Integrated Development Environment User's Guide*. To access the guide, go to the QNX Product Documentation page (**http://www.qnx.com/developers/docs/index.html**) and follow the links for your version of QNX SDP. You can also find documentation for the IDE and for QNX SDP in the IDE's integrated help system.

To use the IDE, you need an installation of QNX SDP.

Uninstalling the IDE

If you need to uninstall the IDE, use the standard procedure for removing an installation. See "Remove an installation" for details.

Get to know the QNX Software Center's user interface

After you install the QNX Software Development Platform (SDP), you see the main screen of the QNX Software Center.

Here's an example of the main screen:
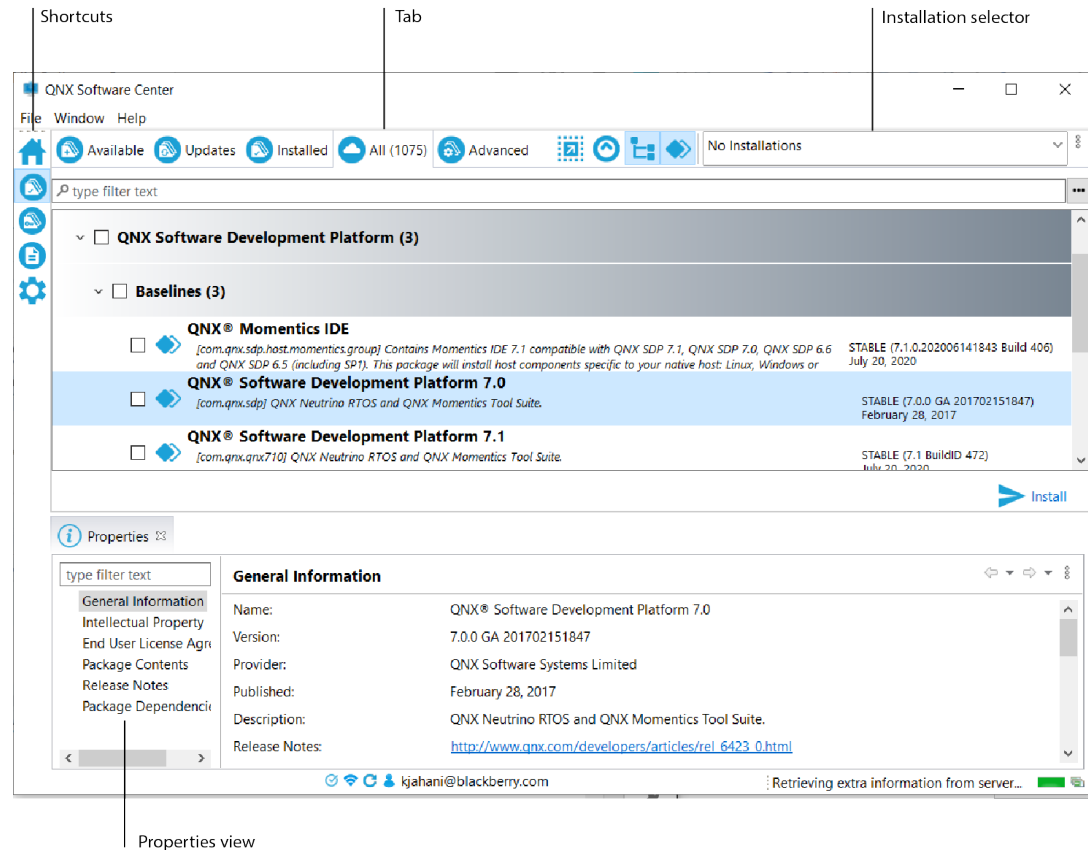
Tabs

The main screen contains several tabs, each of which displays information for the currently selected product installation—in this case, an installation of the QNX Software Development Platform.

Each installation typically contains one *baseline package*, such as QNX SDP, and other added packages, such as board support packages, middleware products, and reference images.

Use this tab:	To:
Updates	Group by, View, Inspect, or Install updates for packages in a product installation.
Available	Group by, View, Inspect, or Install packages that you can add to a product installation.

Use this tab:	To:
All	Group by, View, Inspect, or Install (if compatible) all packages listed in the **Updates**, **Available**, and **Installed** tabs, inlcuding **Inaccessible** packages.
Installed	Group by, View, Inspect, or Uninstall packages you have added to a product installation and patch sets you have applied to the installation.
Advanced	Perform any of the following tasks: • Add or remove a product installation • Edit the properties of an installation • Verify an installation • Inspect the properties of individual files in an installation

> The QNX Software Center may not display every software package offered by BlackBerry QNX. The packages you can view or access depend on the QNX products that your company has licensed and that your QNX license administrator has deployed to you.

Finding specific packages or files

Many views in the QNX Software Center provide a "type filter text" field that can help you find specific items quickly. When you enter a string in this field, the QNX Software Center will filter out any items in the current view that don't match the string.

If a view (for instance, the **Available** tab) displays packages, the QNX Software Center will search the following properties of each package for the string you enter:

• Name

• Package Id

• Status

• Description

• Published

• Package Version

• Package Internal Version

In addition, the QNX Software Center will search through the names of any APIs, libraries, or binaries contained in each package.

You can view the above package properties in the **General Information** pane of the **Properties** view. See "Inspect package and file properties" for more information.

> See the "Filter packages" section to learn how you can create a variety of custom filters.

Identifying recently posted packages

To see which packages have been posted since you last launched or refreshed the QNX Software Center, simply look for the asterisk icon: ✳

This icon may appear in any of the following tabs: **Updates**, **Available**, **All**.

Inspecting properties

Most tabs have an associated **Properties** view that lets you inspect various attributes of packages and files. Simply click a package displayed in the tab, then click the desired option (for instance, **Package Contents**) in the **Properties** view. See "Inspect package and file properties" for more information.

Restoring the window layout

If you ever need to restore the user interface to its default settings, simply select **File → Upkeep → Restore Window Layout**. This action is useful, for example, if you have unintentionally dragged the view panel (which displays the tabs) away from the main window.

Terminology

To make best use of the QNX Software Center, read the glossary, which defines several terms used throughout this guide and the user interface.

Inspect package and file properties

Using the QNX Software Center, you can inspect the properties of any package or file available for, or deployed in, an installation.

> To view all of the links displayed in the property dialogs, you must, in your web browser, be logged in to your myQNX account: https://www.qnx.com/account/login.html

Inspecting the properties of a package

To inspect the properties of a package displayed in any tab of the main screen:

1. Click the package.

2. In the **Properties** view, select one of the following categories:

Select:	To view:
General Information	Name, version, provider, status (Package status), publication date, runtime subsystem, links to license and product information, and other general properties of the package.
Intellectual Property	Copyright and patent information for the package.
End User License Agreement	License agreement for the package.
Package Contents	Files contained in the package and detailed properties of each file. See "Inspecting the properties of a file," below, for details.
Release notes	Release notes for the package, if applicable.
Package Dependencies	Raw dependencies (for instance, libraries) of the package as well as packages that the QNX Software Center will install to satisfy those dependencies. On occasion, this pane may indicate that any one of several packages can satisfy a raw dependency. Suppose, for example, that either of two packages can satisfy a raw dependency for a package that we'll call Package A. In that case, you can choose one of the following approaches: • Do nothing and allow the QNX Software Center to decide which of the two dependency packages to install when you install Package A. OR • Determine which dependency package you prefer and then install it before installing Package A.

Inspecting the properties of a file

To inspect the properties of a file, you can use the **Package Contents** pane of the **Properties** view (see table above) or the **Advanced** tab:

Package Contents pane

The **Package Contents** pane lets you inspect the properties of any file in the currently selected package.

Simply right-click any file in the pane to perform the following actions:

- inspect the file's package asset properties or file properties (see "Package asset properties versus file properties," below)
- open the file, if it's installed
- open the directory that contains the file, if the file is installed
- copy the file path to the clipboard

You can also configure the pane's column headings (Name, Subsystem, Status, etc.) to inspect a variety of package-asset properties for each file:

- Right-click any column heading to hide, show, sort, or group properties.
- Drag the heading of any column left or right to rearrange the data.

Advanced tab

The **Advanced** tab lets you inspect the properties of any file in an installation, without having to first locate or select the file's package.

Simply click the tab and do the following:

1. Search or browse for the file:
   - To search, enter a string in the "type filter text" field.
   - To browse, right-click in the pane, then choose **Group by Package** or **Group by Directory**.

2. Click the file, then go to the **Properties** view (or simply double-click the file) to inspect the file's package asset properties or file properties.

> You can also inspect the properties of files on your host or target by generating manifest reports. See the "Manifest Reports & File Lookup" chapter for details.

Understanding properties

For descriptions of the various properties displayed for files, see "Manifest report fields" in the "Manifest Reports & File Lookup" chapter.

Package asset properties versus file properties

For each file, the QNX Software Center can display the following property types:

Package asset properties

The properties of the file, as declared in the package to which the file belongs. Normally, the QNX Software Center retrieves this information from the myQNX server. If can't connect to the server, it will retrieve the information from local package data instead.

These properties are displayed whenever you choose **Package Asset Properties**; they are also displayed in the **Package Contents** pane of the **Properties** view.

The myQNX server may define newer, more accurate properties than local package data. Thus, we recommend you always use the QNX Software Center online.

If you have to use the QNX Software Center offline, you can still inspect the server-defined values of a file by using the QNX Software Center Portal on the QNX website: https://www.qnx.com/swcenter/

File properties

The properties of the actual file on disk. File properties are displayed only for installed files.

In most cases, the file properties of a file are identical to its package asset properties. The two property types may differ if, for example, you override the file provided by QNX Software Systems with your own file.

To generate a report that flags modified, missing, or unexpected files in an installation, you can use the **Verify Installation** feature. See "Verify and Repair" for details.

Dismissing and restoring the Properties view

To dismiss the **Properties** view, click the **X** in the upper-right corner of the view. To restore the view, select **Window** → **Show Properties View** or select **File** → **Upkeep** → **Reset Window Layout**.

To inspect the properties of any package or file when the view is dismissed, right-click the item and select **Properties**.

Edit installation properties

The **Edit Installation Properties** dialog helps you control which packages will be added to your installations. You can change these properties at any time; you can also assign a different set of properties to each installation. For instance, you can assign one set of properties to an installation used for production purposes, and another set of properties to an installation used for experimental purposes.

A version of the **Edit Installation Properties** dialog appears automatically when you add an installation.

To access the dialog for an existing installation, follow these steps:

1. Click the installation selector at the top-right of the main window, then select the desired installation.

2. In the **Advanced** tab, click **Edit Installation Properties**.

3. Set any of the properties in the following table, then click **Apply** to make sure they take effect.

Use this control:	To:
Name	Modify the name of the installation.
Variant	View the installation variant (for example, development seat or build server). The installation variant is typically determined by the license key you selected when creating the installation.
Description	Modify the description of the installation.
Update Policy	Apply one of the following update policies to the installation: **Liberal** *Install all updates during the initial installation. When subsequently installing a package, install the latest version of each dependent package.* Suppose, for example, that you are installing a filesystem update that requires **libc**, and that an update to **libc** is available. If you choose this policy, the QNX Software Center automatically installs the **libc** update. **Conservative** *Install all updates during the initial installation. When subsequently installing a package, apply the following rules to dependent packages:* • *If a dependent package isn't currently installed, install the latest version.* • *If a dependent package is already installed, use the existing version unless the package being added requires a newer version.* Let's return to above example, where you are installing a filesystem update and an update to **libc** is available. If you choose this policy, the QNX Software Center installs the update to **libc** only if the filesystem update specifically requires the **libc** update.

Use this control:	To:
	Ultraconservative
	Don't install updates during the initial installation. When subsequently installing a package, install the earliest version of each dependent package that isn't already installed.
	Suppose, for example, that you are installing a new package, and that it depends on other packages that aren't currently installed. If you choose this policy, the QNX Software Center chooses the oldest versions of those packages, even if later versions are available.
	This option provides fine-grained control over your installation by letting you update individual packages to the versions you prefer.
	If you choose this policy when adding an installation, the QNX Software Center installs original versions of all packages, without the latest updates.
Package Cache Location	Change the package cache location.
	The package cache improves installation performance by storing copies of packages you have downloaded from the remote myQNX server. If you need to repair a package or install some packages more than once (for instance, to create another installation), the QNX Software Center uses the cached versions instead of downloading the packages again from the remote server.
	The location defined here is used by all installations. Avoid changing it, unless you have a restriction on disk size in your home directory.
	If you need to free up disk space, click the **Clear Cache** button. (You can also use **File → Upkeep → Clear Package Download Cache**). Note, however, that the cache will grow again if you download more packages.
	The **Package Cache Location** and **Clear Cache** controls don't appear when you are adding an installation. They become available for the installation only after it has been added.
Automatically install patches	When installing a component, automatically install any patches or updates that the component requires.
	If you uncheck this control, the installation of any component that requires patches or updates will fail, and you will need to add the patches or updates manually. You will, however, gain greater control over the contents of your installation.
	This control doesn't appear when you are adding an installation. It becomes available for the installation only after the installation has been added.
Install debug symbols	View or install packages that contain only debug symbols, in addition to any other packages you are entitled to.
	If you activate this option, the QNX Software Center also installs any associated debug symbols, when you subsequently install packages.
	This checkbox turns on and remains activated if you select a package or group of packages in the **Installed** tab and then click the tab's **Install Debug Symbols** control. (Clicking the control tells the QNX Software Center to install

Use this control:	To:
	the debug symbols for the selected package and for any updates or addons subsequently added to the installation.)
Install experimental packages	View or install experimental packages, in addition to any other packages you are entitled to.

You can install experimental packages only if this option is activated, even if you have the packages in hand and intend to install them offline.

This property automatically resets to on after the baseline installation to allow experimental packages in new installations. To install only stable packages, reset it after the baseline installation is complete.

If you want to disable the Install experimental packages flag, make sure that the current installation does not include any experimental packages which are already installed to avoid internal inconsistency and help with the update; otherwise, it will be internally inconsistent and you will not be able to update it. |
| **Allow foreign host packages** | Disable the host filter. You can install host packages from another host platform (i.e., Windows host packages on Linux).

If the host filter is disabled, the current host (installation host) type is considered as all type (meaning host type agnostic). If you choose this option, the QNX Software Center:

• installs packages which have specific host requirements such as Windows-only packages on Linux host or the other way around.

• does not automatically install any host toolchain for other hosts. You need to manually select the **Install** option after the original installation is finished.

If this option is selected, shortcuts for Linux or Windows specifically are not created. |
| **Target Architectures** | Choose the architectures for which the QNX Software Center installs target packages.

The target architecture options let you hide packages for architectures you don't need to support. The options apply only to packages that are 100% target-specific, such as BSPs and reference images. Most other packages contain binaries for all architectures and, as such, aren't affected by these options. |

Group by

The **Group by** action allows users to group **Available** packages by Download Center Folder or **Release**.

This option makes it easier to determine which packages are CSP patches and which are from the GA stream.

A new dropdown action option is introduced under the top level for **Updates**, **Available**, **Installed** and **All** tabs.

The selection of actions available for the **Group By** button are **None**, **Product/Category**, **Release**, **Release Folder**, **Status** and **Custom**. The **Custom** option supports various customizations including **None** and current default (in **Product/Category**).

Copyright © 2023, BlackBerry Limited
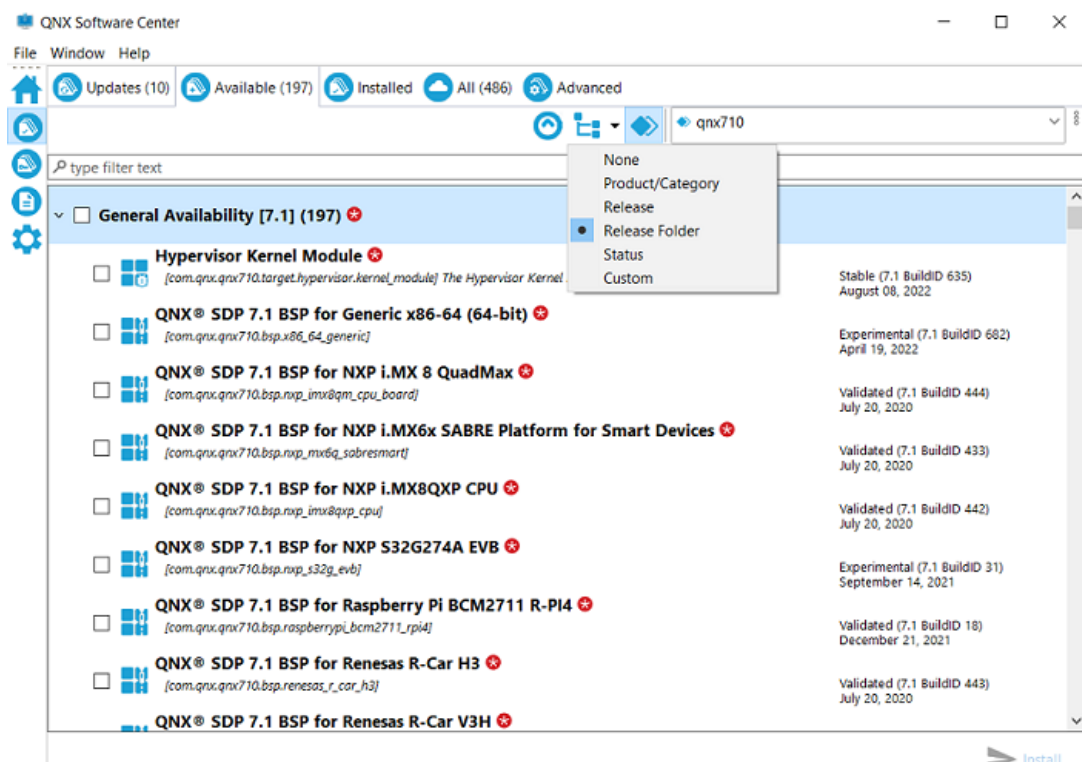
Read the package release notes

If a software package has release notes, you can access the notes through the QNX Software Center, either before or after you install the package.

To view the release notes for a package displayed in any tab of the main screen:

1. Click the package.
2. In the **Properties** view, select **Release Notes**.

Alternatively, you can right-click the package and select **Properties** → **Release Notes**.

To view all of the QNX SDP documentation, launch the QNX Momentics IDE and bring up its integrated help system. You can launch the IDE from the Welcome screen of the QNX Software Center.

See "Install the QNX Momentics IDE" for instructions on installing the IDE.

Email notifications

Subscribe to receive email notifications regarding new package updates or customer notifications.

Subscribing and unsubscribing to email notifications

There are two ways to subscribe and unsubscribe from email notifications:

QNX Software Center

1. Log into the QNX Software Center.
2. Click on **Window** in the top-left corner of the screen, then select the **Preferences** option to navigate to a new window containing the QNX Software Center preference settings.
3. On the left panel of the new window, select the **Package Notifications** option.
4. From here you can manage your software package mailing subscriptions.

QNX Software Center Portal

1. Log into the QNX Software Center Portal on the QNX website: https://www.qnx.com/swcenter/.
2. Click on the email address displayed in the top-right corner of the webpage to open up a drop-down menu, then select the **Account** option to navigate to your account details and email notification settings.
3. From here you can manage your software package mailing subscriptions.

Once subscribed or unsubscribed to the email notifications, an email will be sent to confirm the change.

Install addon packages

Once you've installed the QNX Software Development Platform (SDP), you have a complete environment for developing programs that run on the QNX Neutrino RTOS. The environment includes the QNX Momentics IDE and a rich suite of command-line tools. To run programs on QNX Neutrino, you need to install addon packages, such as a reference image (for instance, a virtual machine for VMware) or a board support package (BSP) for your target system.

Selecting an addon package for installation

The **Available** tab (accessed by selecting **Install New Packages** from the Welcome screen) displays addon packages available for your current installation. By default, the packages are grouped by product, then by package category.

Selecting packages is easy. For instance, to select all of the packages in a package group, click the group's checkbox. To select (or deselect) one or more individual packages within a group, expand the group, then click the checkbox for each desired package.

When you select a package or package group, the QNX Software Center automatically selects all other packages (for instance, dependencies) that it will install with your selection. Likewise, if you deselect that package, the QNX Software Center deselects any other packages that it would have installed with the package.

Installing an addon package

To install a reference image, BSP, or other addon package, such as a middleware product:

1. In the **Available** tab, select the package, then right-click and select **Install**. The QNX Software Center will compute installation requirements and display an **Install** wizard.

2. Make sure that the item you want to install is checked, then click **Next**.

3. Review the list of packages that the QNX Software Center will install, then click **Finish**.

   When installation is complete, you'll see the newly installed item in the **Installed** tab.

You don't have to worry about installing packages in a specific order. The QNX Software Center manages dependencies for you.

> 💡 For an introduction to developing and running QNX Neutrino programs, see the *QuickStart Guide* on the QNX website.

Controlling which addon packages you can view or install

The following controls let you sort the contents of the **Available** tab and control which packages you can view or install:

To:	Do this:
Sort packages by publication date or by name	Toggle the Sort By icon: 🔵

To:	Do this:
View packages by category	Click the Group by Category icon:
View or install only the latest versions of available packages	Click the Show Only Latest icon:
View or install debug symbols, patches, experimental packages, or packages for specific architectures	Use the corresponding checkboxes in the **Edit Installation Properties** dialog.
Change the policy that controls which versions of dependent packages the center will install	Use the **Update Policy** selector in the **Edit Installation Properties** dialog.

To access the **Edit Installation Properties** dialog, go to the **Advanced** tab. For more information, see "Edit installation properties."

Can't find a package?

If the above controls don't help you find the package you want to install, try the following:

- Use the troubleshooting table in "Access inaccessible packages."
- Go to the QNX Software Center *Technotes* (http://www.qnx.com/qsc_technotes) and read the section on "Cannot access packages."

Determining where your package has been installed

Often, you need to know where a package and its files have been installed. For instance, when importing a BSP into the QNX Momentics IDE, you may need to supply the BSP's absolute path name.

To determine the location of an installed package or of any file within an installed package:

1. Make sure you are in the **Installed** tab.
2. Click the package, then select **Package Contents** in the **Properties** view.
3. Look in the **Name** field to see the relative path of each file in the package (i.e. relative to the directory that contains the installation to which the package has been added). Note that a BSP consists of a single zip file.
4. To get the absolute path of a file, right-click the file, then select **Copy Full Path**. The QNX Software Center copies the absolute path to your clipboard.

Install debug symbols

Many packages include debug symbols. To install the debug symbols for a package, you install the package first. You can then install some or all of its debug symbols.

> To install debug symbols, you must first activate the **Install debug symbols** control in the **Edit Installation Properties** dialog.
>
> If **Install debug symbols** in the **Installation Properties** dialog is deactivated, this only applies to packages that provide symbols in a separate package. The debug symbols for all other packages will installed nonetheless.

To install debug symbols for an installed package:

1. Make sure you're in the **Installed** tab, which displays installed packages.

2. Look at the package's icon. If the package has debug symbols, the icon looks like this:

3. Right-click the package and select **Install Debug Symbols**. The QNX Software Center displays a list of the package's debug symbols.

    To install symbols for multiple packages all at once, use Shift+Click or Ctrl+Click (Command+Click on macOS) to select the packages. To install symbols for all of the packages in a package group (the QNX Software Development Platform 7.0 package group, for example), simply right-click the package group.

4. Make sure that the debug symbols you want to install are checked, then click **Next**.

5. Review the list of debug symbols that the QNX Software Center will install, then click **Finish**.

The QNX Software Center displays an Installation Complete dialog when it has finished installing the debug symbols.

Install updates

The QNX Software Center automatically displays product updates that you are entitled to.

Installing one or more updates

The **Updates** tab supports the same package-selection and sorting methods as the **Available** tab. See "Install addon packages" for details.

To install an update for installed package:

1. In the Welcome screen, click **Check for Package Updates**.

2. In the **Updates** tab, select the update or updates you wish to install, then right-click and select **Install**.

   Alternatively, go to the **Installed** tab and find the package you want to update. Right-click the package and select **Update** to see a list of updates for the package, then pick a specific version.

3. Make sure that the update you want to install is checked, then click **Next**.

4. Review the list of packages that the QNX Software Center will install, then click **Finish**.

   When installation is complete, you'll see the update in the **Installed** tab.

You don't have to worry about installing updates in a specific order. The QNX Software Center manages dependencies for you.

Installing all updates

To install all available updates, click **Select All** in the **Updates** tab, then click **Install**.

You can also downgrade a package to a previous version. See "Downgrade a package to a previous version" for more information.

Installation Reporting

Package summary

Click **Show Package Summary** before an installation to view a detailed summary of the package or packages being installed or uninstalled, including links to the release notes.

File summary

Click **Show File Summary** before an installation to view a report of files being added, removed or changed.

The following types of changes may occur:

ADDED

   The file has been added.

REMOVE

The file has been removed.

REPLACED_CHANGED

The file has been changed.

REPLACED_UNCHANGED

The file has been updated but it is exactly the same as before.

REPLACED_SAMECODE

The checksum is different but code is the the same - this is done by checking the checksum of the binary.

This report can also be exported as text or in a CSV file.

Downgrade a package to a previous version

The QNX Software Center allows you to downgrade an installed package to a previous version.

Downgrading a package removes all files that belong to the package and replaces them with files from a previous version of package. It also removes any files added to the package.

To downgrade an installed package:

1. Go to the **Installed** tab, then right-click the package and select **Downgrade**.

   The QNX Software Center calculates system requirements and displays an **Available Updates** dialog.

2. Make sure that the package version you want to install is checked, then click **Next**.

3. Review the list of packages that the QNX Software Center will install, then click **Finish**.

   When installation is complete, you'll see the downgraded package in the **Installed** tab.

You cannot a downgrade a package to a previous version if other installed packages depend on the existing and more recent version.

Uninstall packages

The QNX Software Center lets you uninstall packages that you have added to an installation.

To uninstall a package:

1. In the **Installed** tab, right-click the package, then click **Uninstall**.
2. Review the items to be uninstalled, then click **Finish** to complete the uninstallation.

   The QNX Software Center will remove the uninstalled package from the **Installed** tab and display it in the **Available** tab or **Updates** tab.

You don't have to worry about uninstalling packages in a specific order. The QNX Software Center manages dependencies for you.

Which packages can you uninstall?

Root packages (i.e. packages you've explicitly added to an installation) are the only packages you can completely uninstall. But even then, you can uninstall a root package only if other packages aren't using it. If any other packages use a root package, the package will become a non-root package and remain in the system until the other packages are uninstalled. The QNX Software Center can then uninstall the package.

Uninstalling all packages from an installation

If you uninstall the last package from an installation that isn't a QNX SDP baseline installation, the installation isn't removed, even though it's now empty. To remove the installation itself, go to the **Advanced** tab and select **Remove Installation**. (Make sure that the installation you want to remove is the installation displayed in the installation selector.)

Filter packages

Sometimes you need to inspect, install, or uninstall a specific package or subset of packages. To help you do this, every view that displays packages provides a "type filter text" field. This field performs a fuzzy search to help you narrow down which packages are displayed. In addition to this field, the **All** tab provides a **Package Filter** dialog that lets you use a variety of filtering methods.

Accessing the Package Filter dialog

To access the **Package Filter** dialog, select the **All** tab, then select the menu button (**...**) that appears next to the tab's "type filter text" field.

Choosing a query method

The **Package Filter** dialog provides the following options:

Use this option:	To:
Package List	View one or more specific packages.
	For each desired package, enter the Package Id and, optionally, the Package Internal Version. Make sure to separate the ID and the version by a slash (/). See the dialog for examples.
	For example, you can cut and paste a set of packages from a text-based patch set into the list of packages.
	For definitions of Package Id and Package Internal Version, see "Manifest report fields" in the "Manifest Reports & File Lookup" chapter.
Custom Query	Create a custom query, using the p2ql query language.
	The p2ql language is highly sophisticated, allowing you to create filters that match specific parameters. For instance, you can create a filter to display all packages that contain a given library. Visit **https://wiki.eclipse.org/Equinox/p2/Query_Language_for_p2** to learn more about p2ql.
	See the **Package Filter** dialog for some helpful examples.
Fuzzy Search	Perform a fuzzy search of the specified string.
	This option performs the same function as the "type filter text" field. It searches the following properties of each package for the string you specify:
	• Name
	• Package Id
	• Status
	• Description
	• Published
	• Package Version
	• Package Internal Version

Use this option:	To:
	This option also searches through the names of any APIs, libraries, or binaries contained in each package. You can view the above package properties in the **General Information** pane of the **Properties** view. See "Inspect package and file properties" for more information.
Clear Filter	Clear any filter you may have specified in this dialog or in the "type filter text" field.

Work offline

If you are developing on a workstation that has no Internet connection, you can still use the QNX Software Center in offline mode. You can, for example, import a new installation from a locally mounted package archive. You can also install addons and updates that were included in your import, and work with patch sets. However, the QNX Software Center won't be able to notify you of newer updates from BlackBerry QNX.

For more information, see "Import packages or package archives from your local filesystem" and "Perform manual product activation."

Chapter 6
QNX Software Center Advanced Topics

This chapter guides you through tasks that you may have to perform only on occasion. It also discusses how to address potential connectivity or license activation issues.

This chapter describes how you can:

- import packages or package archives from your local filesystem
- perform manual product activation
- export and import patch sets
- migrate from an evaluation license to a commercial license
- transfer a named user license to another workstation or user
- manage secure storage of your myQNX credentials
- configure your proxy settings
- trust certificates from your proxy server
- downgrade a package to a previous version
- verify and repair
- delete imported package data that you no longer need
- access inaccessible packages
- perform unattended installation of the QNX Software Center and QNX SDP
- deploy QNX SDP for use by large development teams
- make the most of technical support
- configure the default location of the license directory

Import packages or package archives from your local filesystem

You may need to use the QNX Software Center on a workstation that has no Internet connection or that has restricted access to external websites. If so, you can obtain packages or package archives for offline installation, store them on your local filesystem, and then import them into the QNX Software Center to add or modify QNX Software Development Platform (SDP) installations.

A package is the minimal unit of software you can install. It contains metadata (for instance, name, version, dependencies) and, in most cases, a payload of files. The filename for a package typically includes the **.qpkg** extension. A package archive, in comparison, consists of multiple packages packed into a single container such as **.zip**, **.tar**, **.tar.gz**, or **.tar.xz**.

Importing packages to add an installation

To add an installation:

1. Select **File → Import**.
2. Click **Import and Install Package(s)**, then click **Next**.
3. Click **Browse** to select your baseline package or package archive.
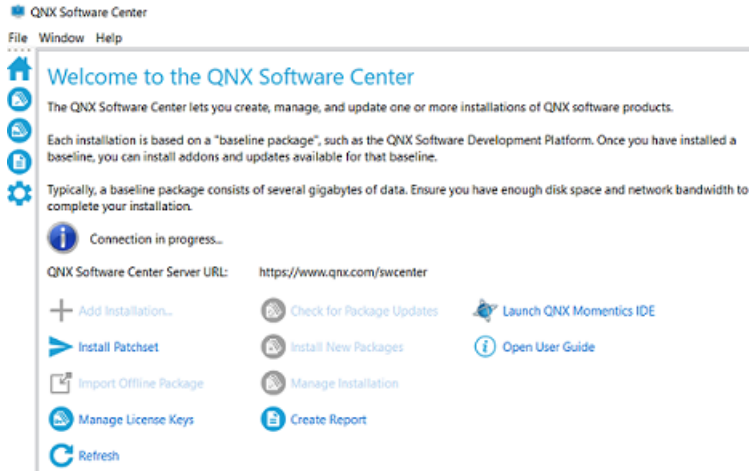
   Alternatively, you can use the **Select package file(s)** field to specify the baseline package for your installation. To specify the desired package, enter the Package Id and, optionally, the Package Internal Version. Make sure to separate the ID and the version by a slash (/). For definitions of Package Id and Package Internal Version, see "Manifest report fields" in the "Manifest Reports & File Lookup" chapter.
4. Click **Create new installation**, then click **Next**.
5. If you haven't already added your license key, click **Add License Key** and enter the key listed in your license certificate or myQNX account. Click **OK**, then click **Next**.

   If your product requires activation, you can activate it now or later. See "Perform manual product activation" to learn how to activate a product offline.
6. Specify the installation folder, name, and other properties of your installation, then click **Next**. For help, see "Edit installation properties."
7. Make sure that the item you want to install is checked, then click **Next**.
8. Review the packages that QNX Software Center will install, then click **Finish** to begin the installation process. The process may take a few minutes to complete.
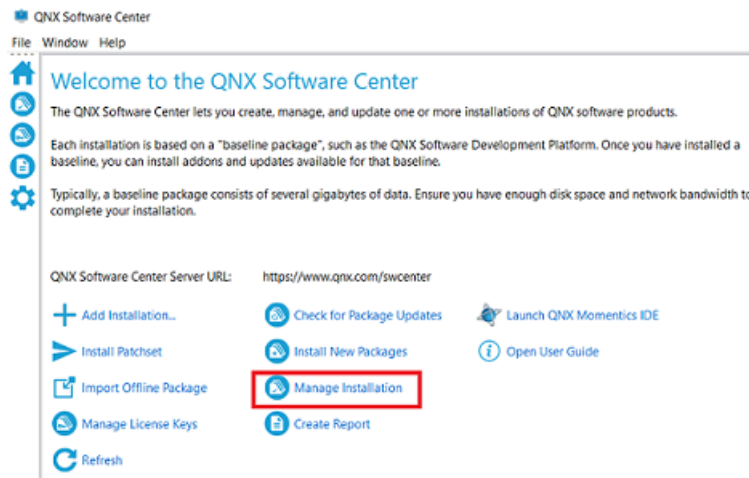
Importing packages to modify an installation
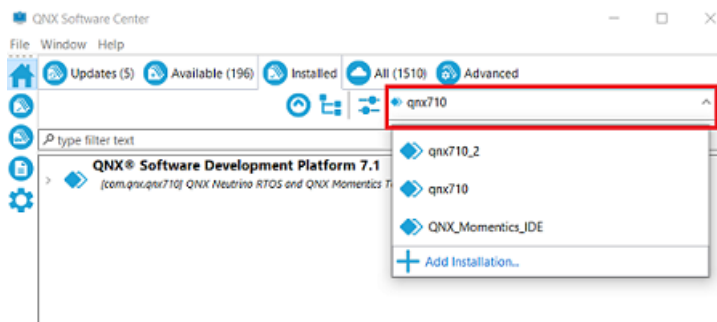
To modify an existing installation:

1. Sign into QNX Software Center and at the home page, "Welcome to QNX Software Center" page is displayed.
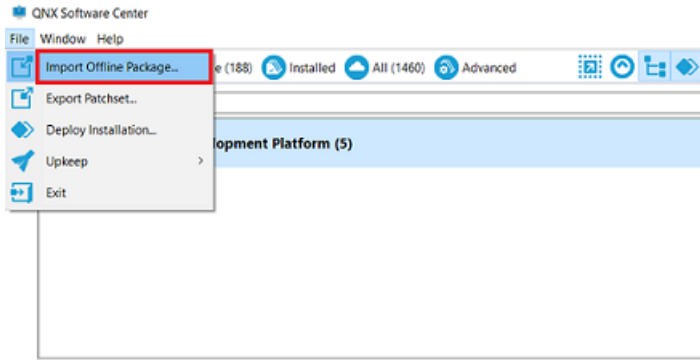
2. Navigate to **Manage Installation** and click it.

3. Click the installation selector in the top-right area of the window, then select the installation you wish to modify.

4. Select **File → Import Offline Package...**

5. Click **Import and Install Package(s)**, then click **Next**.

6. Click **Browse** to select your package or package archive.

   Alternatively, you can use the **Select package file(s)** field to do one of the following:

   - *Specify a list of packages to install* — For each desired package, enter the Package Id and, optionally, the Package Internal Version. Make sure to separate the ID and the version by a slash (/). For definitions of Package Id and Package Internal Version, see "Manifest report fields" in the "Manifest Reports & File Lookup" chapter.

   - *Specify a custom query, using p2ql, to select a set of matching packages to install* — See the **Package Filter** dialog for examples of using the p2ql language and visit **https://wiki.eclipse.org/Equinox/p2/Query_Language_for_p2** to learn more about p2ql.

7. Select **Use current installation**, then click **Next**.

8. Check the items you want to install, then click **Next**.

   You can install all of the items right away, or you can install some now and others later. When installing later, use the standard installation method described in "Install addon packages."

9. Review the packages that will be installed, then click **Finish** to complete the installation process.

Deleting imported packages

When you import a local package for offline installation, the QNX Software Center stores the imported data in the *offline package repository* (also known as the dropins repository). You can clear the contents of this repository if you are running short on disk space. See "Delete imported package data that you no longer need" for details.

Perform manual product activation

If you install the QNX Software Development Platform or other QNX products on a development host without an Internet connection, you must activate the products manually.

> To complete the following steps, you'll need a separate system that is connected to the Internet.
>
> The following steps assume that you are either in the Manage License Keys perspective of the QNX Software Center (you can access this perspective from the Welcome screen) or in the License Key Selection dialog of an installation wizard.

To perform manual product activation:

1. Add a license key for the product you wish to activate:

   a. Click **Add License Key**.

   b. Enter the key in the **License key** field, then click **OK**. If you're prompted to log in to your myQNX account, click **Cancel**. You'll see an error dialog.

   c. Click **Manual Activation**, then go to step 2.

   If you've already added the key (i.e. it appears in the Manage License Keys perspective), do the following:

   a. Right-click the key, then select **Activate**. If you're prompted to log in to your myQNX account, click **Cancel**. You'll see an error dialog.

   b. Click **Manual Activation**, then go to step 2.

2. In the **Activate License Key** dialog, select **Manual activation**. The dialog displays a prompt key.

   Leave the dialog open so you can complete the following steps.

3. From a system connected to the Internet, go to your myQNX profile (https://www.qnx.com/account/index.html) and click the **Manual Product Activation** link.

4. In the **Activate prompt key** field, enter the prompt key generated by the QNX Software Center, then verify that the named email address is correct.

5. Click **Generate Response Key** to generate your activation key.

6. Go back to the QNX Software Center and copy your activation key into the **Activation key** field.

7. Click **OK** to complete the activation.

Export and import patch sets

The QNX Software Center lets you export and import *patch sets*.

A patch set is like a recipe for installing packages. It tells the QNX Software Center not only which packages to install, but also which version of each package to install.

Patch sets enable all members of a team to standardize on the same software base. They also make it easier to reproduce issues across systems.

You can export a patch set from one installation and import it into a different installation, either on the same development host or on a different host. When importing a patch set, you can choose whether the patch set creates a new installation or modifies an existing one.

A patch set doesn't contain a payload of files; it contains only a list of required packages and their versions. So, to import a patch set, you must have access to the packages that the patch set specifies, either from the myQNX server (if you use the QNX Software Center online) or through packages you have imported from your local filesystem.

You must also be entitled to use the packages specified in the patch set, or the QNX Software Center won't complete the installation. If this issue occurs, contact your license administrator to determine if they can deploy the appropriate licenses.

Export a patch set

To create and export a patch set:

1. From the Welcome screen, select **Manage Installation**, click the installation selector, and then select the installation that will form the basis of your patch set.

   For example, if you select the installation named **qnx700**, the selector will look like this:

◈ qnx700	⌄

2. Select **File → Export Patchset**. You'll see the **Edit Patch Set** dialog.

3. Use the following guidelines to set the patch set properties:

 | Field | Description |
   |---|---|
   | **Author Id** | The reverse DNS address of your organization. For example: `com.mycompany` |
   | **Author Name** | Typically, the name of your organization. |
   | **Package Id** | An alphanumeric ID, prefixed by the **Author ID**. For example: `com.mycompany.projectA.patchset` |
   | **Package Name** | A meaningful description of the patch set. Can contain spaces. |
   | **Package Version** | The version of the patch set. |

Field	Description
	This field must contain four segments, separated by periods. The first three segments are numeric; the last segment is a string and typically specifies a date.

> ⚠️ **CAUTION:** The QNX Software Center uses the first three segments for numeric comparison. For instance, it sees 1.0.2 as greater than 1.0.100. However, it uses the last segment for *alphanumeric* comparison. Thus, it sees 1.0.0.2 as later than 1.0.0.10 because 2 as a string is greater than 10.

4. Review the contents of the **Required Packages** field if you wish to see which packages the patch set requires.

5. Look at the **Properties** field to see whether the patch set specifies experimental packages.

   A stable status indicates that no experimental packages are specified, while an experimental status indicates that one or more experimental packages are specified.

6. Ignore the **Filter** field, as it used primarily for internal purposes.

7. Enable the option **Include host packages** which allows you to include host packages in the patchset. QSC automatically exports the patchset with conditional dependencies on host packages. (That is, it depends on the Linux package only if you are installing on Linux).
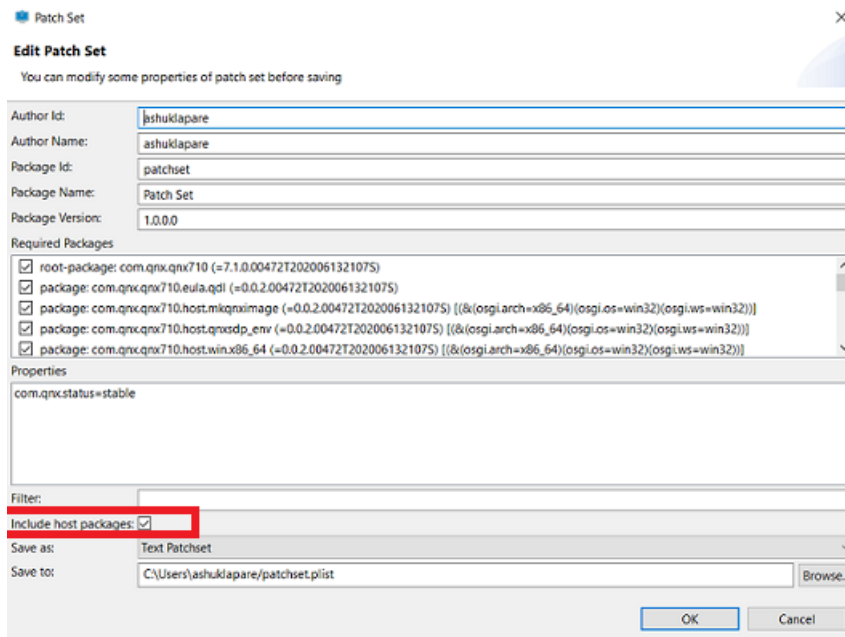
> 💡 To create a universal patchset which can be installed on all the hosts (with package IDs locked), you need to install all host packages first (e.g., Linux and Windows on Linux host).

8. Disable the option **Include host packages** to exclude the host packages from the patchset (i.e., Linux host packages are not in the patchset, only the target packages are).
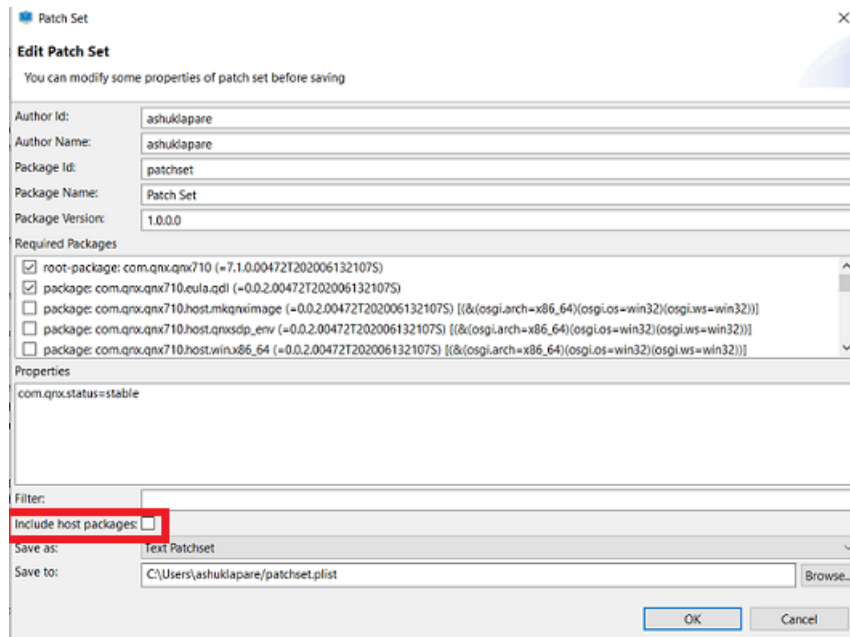
9. Click the **Save as** field and choose one of the following options:

- **Text Patchset** — Exports an editable, text-based patch set that can install all packages specified in the patch set, without installing itself. We recommend this option for most cases.

- **Unsigned Package Patchset** — Exports a non-editable patch set in the form of a QNX Software Center package. This option is for more advanced scenarios, where, for example, you may wish to version the patch set, add metadata to the patch set, or create a locked patch set.

10. In the **Save to** field, assign a name to the patch set.

11. Click **OK** to complete the export.

Note that, when you export a patch set, you export all items in the installation.

Import a patch set to add an installation

⚠ **CAUTION:** If another installation is in progress, wait for it to complete before before attempting to import a patch set.

To add an installation:

1. Go to the Welcome screen and select **Install Patchset**.

2. Click **Browse** to locate and select the patch set.

3. If you wish to lock the packages specified by the patch set (i.e. prevent them from being updated), uncheck the box labeled **Install patchset contents, do not lock installation**. Otherwise, leave the box checked.

4. Click **Create new installation**, then click **Next**.

5. Choose a license key for the product being installed, then click **Next**.

Note the following:

- If you need help with choosing the appropriate license, consult with your license administrator or visit the licensing information page.
- If you have to add a license key manually (for instance, you use the QNX Software Center offline), click **Add License Key** and enter the key listed in your license certificate or myQNX account.
- If your product requires activation, you can activate it now or later. See "Perform manual product activation" if you need to activate offline.

6. Specify the installation folder, name, and other properties of your installation, then click **Next**. For help, see "Edit installation properties."

7. Make sure that the baseline package (for example, QNX Software Development Platform) you want to install is checked, then click **Next**.

8. Review the list of packages that the QNX Software Center will install, then click **Finish** to start the installation process. The process may take a few minutes to complete.

   Too little disk space?

   If the **Review Packages** dialog warns that you have too little disk space, don't click **Finish**. Instead, click **Back**, free up some disk space, and then click **Next** to return to the **Review Packages** dialog. If necessary, repeat this procedure until the dialog no longer issues the warning.

Upon completion, the new installation becomes your current installation and appears in the **Installed** tab.

Import a patch set to modify an installation

To modify an existing installation:

1. Click the installation selector in the top-right area of the main window, then select the installation you wish to modify.

2. Go to the Welcome screen and select **Install Patchset**.

3. Click **Browse** to locate and select the patch set.

4. If you wish to lock the packages specified by the patch set (i.e. prevent them from being updated), uncheck the box labeled **Install patchset contents, do not lock installation**. Otherwise, leave the box checked.

5. Select **Use current installation**, then click **Next**.

6. Makes sure that all the packages you wish to install are checked, then click **Next**.

7. Review the list of packages that the QNX Software Center will install, then click **Finish**.

Upon completion of the import, the packages specified in the patch set appear in the **Installed** tab.

Updating packages specified in a patch set

Patch sets help ensure that developers use specific versions of packages. Thus, when you import a patch set, the QNX Software Center gives you the option of locking all of the packages that the patch set specifies. This locking prevents the packages from being updated. (The locking applies only to packages specified by the patch set. You can still add new packages to an installation based on, or modified by, a patch set.)

If you need to update any packages specified in a locked patch set, use one of the following methods:

- Import a new patch set that specifies the updated versions of the packages. The new patch set should have the same Package Id but a higher Package Version than the original patch set.

  or

- Uninstall the patch set and install the desired package versions manually.

If a patch set isn't locked, you can simply use the standard procedure for updating its packages. See "Install updates" for details.

Access your installation from the IDE

As with any other installation, an installation based on, or modified by, a patch set will appear in the QNX Momentics IDE as an SDK. To select the SDK, launch the IDE and click **Window** → **Preferences** → **QNX**. (You may need to restart the IDE to see the SDK.)

Creating or editing a text-based patch set

Each line in a text-based patchset specifies a package, in this form:

PackageId ['/' *PackageVersion* ['/' *Filter* ['/' *PackageType*]]] | *PackageId* '=' *PackageVersion*

When editing a patchset or creating your own patchset, we recommend that you use only the two first fields, which specify the unique alphanumeric ID and version of the package. Here are some examples:

```
com.qnx.sdp.target.connectivity.io_char/7.0.4284.S201908021626
com.qnx.sdp.target.connectivity.pci.compat/7.0.78.S201906201118
com.qnx.bsp.x86_64_generic/7.0.4185.E201907280054
```

Avoid editing any existing *Filter* and *PackageVersion* fields.

Migrate from an evaluation license to a commercial license

When you migrate from an evaluation license to a commercial license, activation of your commercial license key typically occurs automatically. Follow these steps only if auto-activation fails, in which case the QNX Software Center will display an activation error message.

To activate your product if auto-activation fails:

1. Go to the Welcome screen of the QNX Software Center and click **Manage License Keys**.

   If your license administrator has deployed a commercial license to you, its key should appear in the License Key perspective. If you don't see the key, visit your myQNX account profile (https://www.qnx.com/account/index.html) to check whether the license has, in fact, been deployed to you.

2. Right-click the commercial license key and select **Activate**. The status of the product will, within a few seconds, change to Activated.

To add license keys or activate products on a system without an Internet connection, see "Perform manual product activation."

If possible, obtain your commercial license before the end of your evaluation period. That way, you can ensure continuous access to product features.

Transfer a named user license to another user or workstation

If you have a named user license that must be deployed to another user, you must deactivate the corresponding product on your workstation. The license administrator can then deploy the license to the other user. Likewise, if you need to change workstations (for instance, you are upgrading to a new machine), you must, on the old machine, deactivate any product covered by a named user license. You can then reactivate the product on the new machine.

To deactivate a product on your workstation that has a named user license:

1. Launch the QNX Software Center.

2. From the Welcome screen, click **Manage License Keys**.

3. Right-click the product's license key and select **Deactivate**, then click **OK** to confirm.

The deactivation disables product features and removes your entitlement to use the product on that workstation.

If you have installed components of a product collection across multiple machines, you must explicitly deactivate the component residing on each machine. Otherwise, the product collection will remain activated.

Suppose, for example, that you have a product collection consisting of QNX SDP 7.0 and QNX SDP 7.1. If you installed the 7.0 product version on one machine and the 7.1 product version on a different machine, you need to deactivate each version on the machine where it is installed.

This requirement applies only to collections governed by a named user license. See "Access multiple versions of a product" for more information on product collections.

You should always use the QNX Software Center to deactivate a product. If you can't use the QNX Software Center (for instance, your workstation has been damaged or lost), access your myQNX account profile (https://www.qnx.com/account/index.html) to deactivate the product from there. The profile page will display a Manual Product Deactivation link if any of your products are currently activated.

If for some reason deactivation from myQNX doesn't work, your administrator will need to submit a deactivation request through the myQNX License Manager. See "Deactivate products governed by a named user license" for details.

If you are deactivating the product so your license administrator can deploy the license to another user, your job is done.

If, however, you need to transfer the license to your other workstation, follow these steps:

1. On the new workstation, start the QNX Software Center, using the same login credentials as on your old machine.

2. From the Welcome screen, select **Manage License Keys**. If your workstation is online, the product's license key should appear automatically.

3. Right-click the license key and select **Activate**.

When you deactivate a product on one machine, wait a few minutes before activating it on another machine. Otherwise, the QNX Software Center may display a message saying that the request is still being processed.

Manage secure storage of your myQNX credentials

When it launches, the QNX Software Center displays a login dialog that prompts you for your myQNX credentials. The dialog includes a **Save password to disk** option, which saves your credentials in an encrypted form, using Eclipse secure storage. This secure storage feature lets you log in automatically whenever you subsequently launch the QNX Software Center.

Resolving login issues related to secure storage

If you use the secure storage feature, you may, on occasion, need to delete and reset the storage record, either because your myQNX credentials have changed or because the storage mechanism remembers the wrong myQNX credentials.

To reset the secure storage record, you can use one of the following methods.

Clear Saved Credentials method

1. Select **File → Upkeep → Clear Saved Credentials**.
2. When prompted, click **Yes** to reset the storage record and restart the QNX Software Center.
3. When prompted, enter your myQNX credentials, then click **Save password to disk** if you wish to create a new storage record for the credentials.

### Secure Storage dialog method

To use the Secure Storage dialog method:

1. Select **Window → Preferences → Security → Secure Storage**.
2. Select **Contents → Default Secure Storage**, then click **Delete**.
3. When prompted, click **Yes** to restart the QNX Software Center.
4. When prompted, enter your myQNX credentials, then click **Save password to disk** if you wish to create a new storage record for the credentials.

The **Secure Storage** dialog contains a variety of other settings, but you rarely need to use them. They're provided mostly for troubleshooting and, to a lesser degree, for use by system administrators and power users.

For more information on the **Secure Storage** dialog, see "Secure storage preference page" in the *Eclipse Workbench User Guide*.

Configure your proxy settings

If you launch the QNX Software Center and see a popup dialog like the one below, your workstation probably sits behind a proxy server.
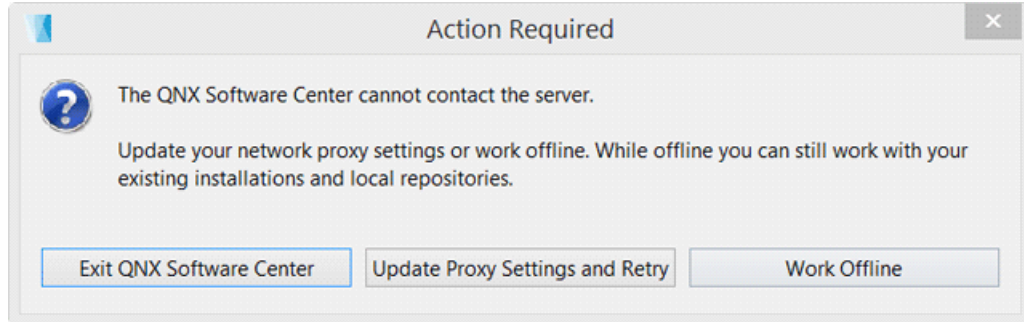
If the dialog appears, follow these steps:

1. Make sure your workstation has a live Internet connection. If it does, you need to configure the QNX Software Center's proxy settings so it can use the connection to access the myQNX server.

2. Click **Update Proxy Settings and Retry** to open the **Proxy Connections** dialog. (You can also access this dialog by selecting **Window → Preferences → Proxy Connections**.)

3. Under **Proxy type**, click **HTTP/HTTPS**.

> The **Proxy Connections** dialog also contains an option for the SOCKS proxy type, but the option isn't supported.

---

4. Under **HTTP/HTTPS settings**, uncheck **Use system settings** to configure the proxy manually. (When **Use system settings** is checked, the QNX Software Center tries to use the default proxy setting of your workstation.)

5. If your proxy server handles both protocols, enter its hostname (or IP address) and port number into the **HTTPS Proxy** field.

   If, however, you have separate servers for HTTPS and HTTP, uncheck **Proxy handles both HTTPS and HTTP**, then enter their respective hostnames and ports into the **HTTPS Proxy** and **HTTP Proxy** fields.

   If you're specifying an IPv6 address for a proxy server, you need to enclose the hostname in brackets to separate it from the port number: [*hostname*] **:** *port*

6. Click the **Test connection** button. If prompted, enter your proxy credentials and click **OK**. (The QNX Software Center doesn't save the credentials at this point.)

> In many corporate networks, the proxy server is integrated with the corporate network login mechanism. So, when prompted for your proxy credentials, try using your normal network user name and password. If that doesn't work, ask your IT department for the correct credentials.

---

7. Wait for a notice stating that the connection succeeded. If the connection doesn't succeed, you may need to enter different proxy settings or credentials.

8. If the connection succeeds, click **Apply**, then click **OK**.

9. If prompted, enter your proxy credentials, then click **OK**.

The QNX Software Center should now connect to the myQNX server and allow you to log in to the myQNX server with your myQNX credentials. If not, see "Trust certificates from your proxy server" and see also the *QNX Software Center Technotes*: http://www.qnx.com/qsc_technotes.

Trust certificates from your proxy server

Sometimes, the QNX Software Center won't access the myQNX server, even after you've configured your proxy settings correctly. This issue may occur if your proxy server rewrites SSL certificates for secure HTTPS remote hosts, in which case the QNX Software Center won't trust the certificate that returns from the proxy.

To determine if this issue is occurring, find the **.qnx/swupdate/.metadata/.log** file under your home directory, then look for an error about being unable to find a "valid certification path to the requested target".

If you see this error, follow the steps below to make the QNX Software Center trust the certificate from your proxy server:

1. In your installation of the QNX Software Center, look for the Java Runtime Environment (JRE) under the **/features/com.qnx.tools.jre.*version*** directory. In this directory, you'll see `keytool`, the Java certificate management utility.

2. From the command line, run `keytool` as follows:

    ```
    keytool.exe -J-Dhttps.proxyHost=proxy_hostname -J-Dhttps.proxyPort=proxy_port
    -printcert -rfc -sslserver qnx.com:443
    ```

    where *proxy_hostname* and *proxy_port* are the hostname and port of your proxy server.

3. The command may output several certificates. Copy the contents of the last certificate into a file and name the file **proxy.pem**. (The file should start with the BEGIN CERTIFICATE line and end with the END CERTIFICATE line.)

4. You now need to import the certificate into the Java trust store that the QNX Software Center uses. Using a version of OpenSSL recommended by your IT department, convert the certificate to `der` format:

    ```
    openssl x509 -outform der -in proxy.pem -out proxy.der
    ```

    Next, import the certificate into the trust store, which is located in the **cacerts** directory. For instance, in Windows you would enter:

    ```
    keytool -import -alias proxykey
    -keystore \features\com.qnx.tools.jre.win32.x86_64_1.8.121\jre1.8.0_121\libs\security\cacerts
    -file proxy.der
    ```

    If the command asks for a password, enter `changeit`, which is the default Java trust store password.

Once the certificate is imported you should be able to use QNX Software Center with your proxy server. If you continue to encounter connection issues, see the *QNX Software Center Technotes*: http://www.qnx.com/qsc_technotes.

Verify and Repair

The QNX Software Center's Verify Installation feature provides options to diagnose and repair installation issues. For instance, you can remove unexpected files. You can also return a modified installed package to its original state, restoring any files that have been changed or deleted. The QNX Software Center's Repair option simply un-installs and re-installs the selected package.

How to use the QNX Software Center's Verify Installation feature:

1. Click the installation selector in the top-right area of the main window, then select the installation you wish to verify.

   For example, if you select the installation named **qnx700**, the selector will look like this:

◆ qnx700	∨

2. In the **Advanced** tab, click **Verify Installation**.

   The QNX Software Center generates a report that indicates which, if any, packages have issues. It also displays a set of possible actions.

3. At this point, you can apply fixes on a global or per-package basis:

   • To apply fixes globally, simply click the desired action. For instance, clicking **Remove Extra Files** will remove all extra files from the installation.

   • To apply fixes only to a specific package, right-click the package, then choose the desired action from the displayed menu.

The QNX Software Center displays an Installation Complete dialog when it has completed the repair.

> 💡 To repair packages, the QNX Software Center uses the contents of the package cache. If you have cleared the cache, the QNX Software Center must be online so it can re-download the package.
>
> If you add files to a package and then repair the package, the QNX Software Center doesn't remove the new files. It only restores files explicitly listed in the package.

How to use the QNX Software Center's Repair option:

1. In the **Installed** tab, type the name or Id of the package you wish to repair into the search filter.

2. Right-click on the package and select the **Repair** option.

The QNX Software Center then re-installs the package.

Delete imported package data that you no longer need

When you import a local package for offline installation, the QNX Software Center stores the imported data in the *offline package repository*, which consists of the **$HOME/.qnx/swupdate/dropins** and **$HOME/.qnx/swupdate/dropins-repo** directories. You may need to delete the contents of these directories on occasion, either to free up space on your hard drive or to improve the performance of the QNX Software Center. (The QNX Software Center scans these directories when it launches, so its startup performance may degrade if the directories contain many packages.)

To delete imported package data:

- Select **File → Upkeep → Clear Offline Package Repository**.
- Click **Yes** when prompted to clear the repository.

You can also free up disk space by clearing the package cache. For more information, see the discussion of the package cache location in the "Edit installation properties" section of this guide. Note, however, that clearing the package cache can reduce QNX Software Center performance.

Access inaccessible packages

The **All** tab displays all packages, including inaccessible packages that you can't currently view or install through the **Updates** tab or **Available** tab. A package may appear as inaccessible for several reasons. For instance, the package is an update to a product with an expired support plan, or the package is experimental and you have disabled installation of experimental packages.

If you do need to install one of the packages that are currently marked as inaccessible, please consult the following table for possible causes and solutions.

| Possible cause	Recommended solution
Package is a patch and installation of patches is disabled.	Open the **Edit Installation Properties** dialog and activate the checkbox for **Automatically install patches**.
Package contains only debug symbols and installation of debug symbols is disabled.	Open the **Edit Installation Properties** dialog and activate the checkbox for **Install debug symbols**. See also "Install debug symbols."
Package is experimental and installation of experimental packages is disabled.	Open the **Edit Installation Properties** dialog and activate the checkbox for **Install experimental packages**.
Package is for a specific architecture and installation of packages for that architecture is disabled.	Open the **Edit Installation Properties** dialog and activate the checkbox for your target architecture.
Package can't be installed because it's incompatible with the current host OS (for instance, you're attempting to install a Windows-only package on Linux).	Check the package release notes for host OS compatibility.
Package is incompatible with the current baseline product (for instance, it's for a different version of QNX SDP).	Check the package release notes for product compatibility.
Package is for a product that requires an active support plan.	Check with your license administrator. If you are a license administrator, use the myQNX License Manager to determine the status of the support plan.
Package is for a product that requires product registration (for instance, the QNX OS for Safety) and account setup isn't complete.	Check with your license administrator.

For more help on accessing packages, go to the QNX Software Center *Technotes* (http://www.qnx.com/qsc_technotes) and read the section on "Cannot access packages."
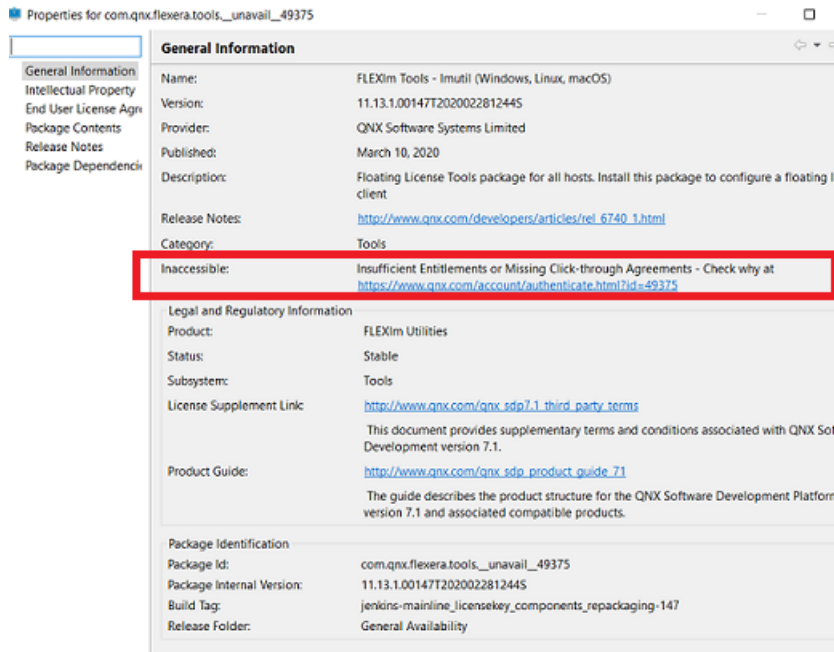
Check Why

To check more about the inaccessible packages, the **Check Why** option has a link provided next to it.

1. Select one of the inaccessible packages listed under the **All** tab.

2. Right click and select **Properties**.
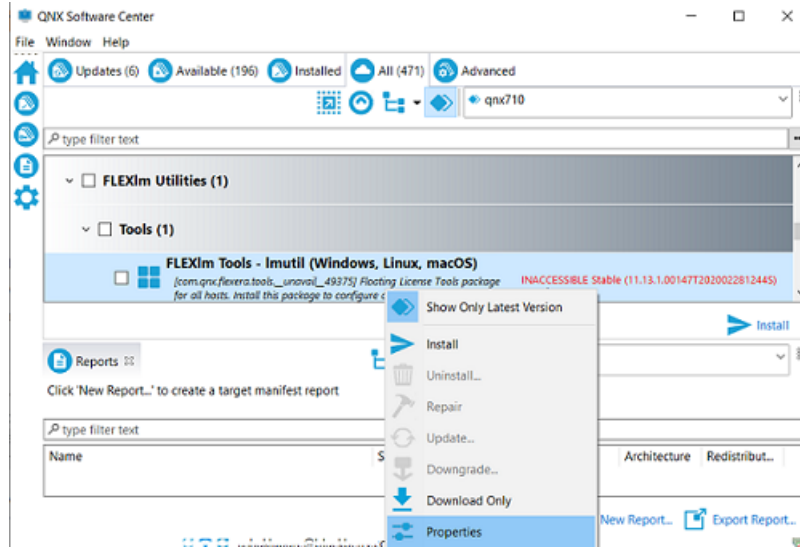
3. Click the **General Information** tab and navigate to **Inaccessible**.

Perform unattended installation of the QNX Software Center and QNX SDP

Using the QNX Software Center command line, you can perform unattended installation of the QNX Software Center and the QNX Software Development Platform (SDP). This technique is especially useful if you intend to deploy QNX SDP on a headless build server.

For details, see the guide titled "Creating and Deploying Installations of QNX SDP in Large Organizations." You can find the guide on the QNX Product Documentation page: **http://www.qnx.com/developers/docs/index.html**

Deploy installations of QNX SDP for use by large development teams

Using the QNX Software Center's Deploy Installation feature, you can create installations of the QNX Software Development Platform (SDP) that can be maintained through configuration management, copied to a shared drive, or otherwise be made accessible to members of a large development team.

For details, see the guide titled "Creating and Deploying Installations of QNX SDP in Large Organizations." You can find the guide on the QNX Product Documentation page: **http://www.qnx.com/developers/docs/index.html**

Make the most of technical support

If you ever experience an issue that may require technical support, here are a few things you can do to help resolve the issue as fast as possible:

- Read the technotes
- Verify your installation
- Check the log files
- Use the clipboard feature to report product activation issues

Before you call, read the technotes

Sometimes, you can avoid a support call together altogether by consulting the QNX Software Center *Technotes*. To access the technotes, visit http://www.qnx.com/qsc_technotes or the QNX Product Documentation page (http://www.qnx.com/developers/docs/index.html).

The QNX Software Center *Technotes* help you address issues that might occur when you are installing packages, activating products, or connecting to the myQNX server.

Verify your installation

The **Verify Installation** feature can detect and repair various issues in an installation, such as missing, unexpected, or modified files. It also identifies the severity of any issues.

To verify and repair an installation, see "Verify and Repair" in this guide.

Check the log files

Every time it runs, the QNX Software Center generates a text-based log file in the **.qnx/swupdate** subdirectory of your home directory. The filename is **qnxsoftwarecenter-*number*.log**, where *number* can range from **0** (newest log) to **9** (oldest log).

Use the clipboard feature to report product activation issues

When you attempt to activate or deactivate a product in the License Key perspective, the QNX Software Center displays a popup dialog that contains a **Copy to Clipboard** button. The information generated by this button can be key to resolving product activation issues.

Configure the default location of the license directory

By default, QSC manages user licenses in the directory ~/.qnx/license, which is inside the user directory.

To change this location, use the environment variable QNX_CONFIGURATION_EXCLUSIVE. The variable must point to the parent of license directory, i.e., QNX_CONFIGURATION_EXCLUSIVE=/opt/qnx. In this case, the licenses are in the directory /opt/qnx/license, which must be a location where QSC can write to.

QSC ignores the value of QNX_CONFIGURATION if it is set.

Chapter 7
Using the QNX Software Center Command Line

The `qnxsoftwarecenter_clt` script lets you run the QNX Software Center from the command line. The script uses the same infrastructure as the QNX Software Center's GUI.

Syntax

```
qnxsoftwarecenter_clt options
```

Runs on

Microsoft Windows, Linux, macOS

Options

-activateAll

Activate all license keys that require remote activation.

-addLicenseKey *key*

Add the specified license key and activate the product for which the key is being added.

If you are using the QNX Software Center offline, use this option to specify the desired license key for your installation. If you are using the QNX Software Center online and have multiple license keys, use this option if you wish to add and activate a specific key.

-cleanInstall

Remove the active baseline installation, if one exists.

You can run this option before creating a new installation with the `-installBaseline` option.

This option performs the same removal operations as the `-uninstallBaseline` option and the **Remove Installation** command in the GUI.

⚠️ **CAUTION:** This option deletes all of the files in the active installation.

-clearDropins

Delete the imported package data stored in the **dropins** directory.

This option can help free up disk space and improve startup performance of the QNX Software Center. See "Delete imported package data that you no longer need" for more information.

-deactivateAll

Deactivate all activated license keys.

Use this command when you need to release all licenses held by the current machine (for instance, when migrating to another machine).

-destination *path*

> Perform operations on the installation located in the specified path (i.e. make this installation the active installation).
>
> You should invoke this option before invoking any options that install or uninstall packages, or that query packages on disk. You can omit it if you work with only one installation. See the Description for details.
>
> If you omit this option, the QNX Software Center uses the last specified destination. However, if you're writing scripts, we recommend that you always specify the destination.

-exportPatchSet *output_file* **[***properties***]**

-exportPatchSetText *output_file* **[***properties***]**

> Export a patch set based on the active installation.
>
> This option comes in two forms: -exportPatchSet creates a non-editable patch set in the form of a QNX Software Center package, while -exportPatchSetText creates an editable, text-based patch set. We recommend -exportPatchSetText for most cases. See "Export and import patch sets" in this guide for more information.
>
> When naming the output file, add a **.qpkg** extension if you've specified -exportPatchSet and add a **.plist** extension if you've specified -exportPatchSetText.
>
> Here are the supported properties:

> **-authorId** *author_id*
>
>> The reverse DNS address of your organization. For example: com.mycompany

> **-packageId** *patchset_id*
>
>> An alphanumeric package ID, prefixed by the *author_id*. For example: com.mycompany.projectA.patchset

> **-packageName** *name*
>
>> A meaningful description of the patch set. Remember to use quotes if the description includes spaces. For example: "QNX SDP 7.0 Patch Set for Project A"
>>
>> This argument is optional.

> **-packageVersion** *version*
>
>> The version number of the patch set.
>>
>> The version number must contain four segments, separated by periods. The first three segments are numeric; the last segment is a string and typically specifies a date.

> ⚠ **CAUTION:** The QNX Software Center uses the first three segments for numeric comparison. For instance, it sees 1.0.2 as greater than 1.0.100. However, it uses the last segment for *alphanumeric* comparison. Thus, it sees 1.0.0.2 as later than 1.0.0.10 because 2 as a string is greater than 10.

`-includeHostPackages=false`

> Don't include host packages in the patchset (only works with
> `-exportPatchSet`).

`-status [stable|experimental]`

> Assign the specified status to the patch set. Default is stable. See Package status.

> 💡 If you use the `-destination` option, it should appear before
> `-exportPatchSet` on the command line.

`-fileInfo path`

> List the server-defined properties for the file specified by *path*.
>
> The `-fileInfo` option searches for the matching file on the myQNX server, which defines the latest properties for files supplied by BlackBerry QNX. If a match is found, `-fileInfo` generates a report of the file's server-defined properties.
>
> The specified path, which must point to an existing file in the host filesystem, can be either absolute or relative to the current directory. For example:
>
> `-fileInfo ~/qnx700/target/qnx7/aarch64le/lib/libcam.so`
>
> If `-fileInfo` can't find a match on the server, it will try to find a match in the active installation. If it succeeds, it will generate a report of the properties defined for the file by the local package data. (You can use `-destination` to choose the active installation.)
>
> You can use `-reportFormat` to format the report data, `-reportPresentation` to apply a template to the report, and `-reportSaveAs` to save the report to a file.

`-fileSearch query`

> List the server-defined properties for any file that matches the specified properties.
>
> The `-fileSearch` option searches for matching files on the myQNX server, which defines the latest properties for files supplied by BlackBerry QNX. For every match it finds, `-fileSearch` generates a report of the file's server-defined properties. If `-fileSearch` can't connect to the server or find a matching file, it prints an error.
>
> The *query* argument can specify one or more properties, expressed as field-value pairs:
>
> *property_name=value* & *property_name=value* & ...
>
> Here are the supported properties:

Property	Description
`buildId`	A unique ID, encoded in hex, that identifies the binary payload of the file.
`digestSHA512`	A Secure Hash Algorithm 512-bit checksum, encoded in base64.
`path`	The partial or full path of the file.

Property	Description
`packageId`	The unique alphanumeric ID of the package to which the file belongs.
`version`	The version of the package to which the file belongs.
`status`	The software maturity of the file (either `stable` or `experimental`).

To determine the value of any of these attributes, you can run the "`use -i`" command on the file that you're interested in (see the `use` entry in the QNX Neutrino RTOS *Utilities Reference* for details).

The *query* argument also accepts parameters to control the sorting and pagination of output:

Attribute	Description
`sortOrder`	Use the specified list of comma-separated properties for sorting. The list can contain any of the properties in the table above. You can use the + and - characters to specify ascending and descending order, respectively. For instance: `sortOrder=buildId+,path-,version+`
`pageNo`	Return the specified page of results. If you omit `pageSize`, `pageNo` is ignored.
`pageSize`	Print the specified number of results on each page. For instance, if you specify 5, each page will contain the results for 5 files. If you omit `pageSize`, the page size is unlimited. If you omit `pageNo`, `pageSize` is ignored.
`details`	Print the details for each file on a separate row. For instance, if you specify `details` and the same file is found in two packages, the report will include two rows, one for each instance of the file. But if you omit `details`, the report will use one row for both files and collapse the column data. In that case, if any column value (for instance, `packageId`) differs between the two files, the report will present both values in the same cell.

> 💡 If you run the command from a Unix shell, you need to escape any & and * characters in the *query* argument. For example:
>
> ```
> pageSize=100\&path=deva\*
> ```

You can use `-reportFormat` to format the report data, `-reportPresentation` to apply a template to the report, and `-reportSaveAs` to save the report to a file.

-help

> Display a list of QNX Software Center options.

-[http.|https.]proxy.host *host*

> Connect to the specified HTTP or HTTPS proxy host.

-[http.|https.]proxy.port *port*

> Use the specified HTTP or HTTPS proxy port.

-[http.|https.]proxy.user *user*

> Use the specified HTTP or HTTPS proxy user name.

-[http.|https.]proxy.password *password*

> Use the specified HTTP or HTTPS proxy password.

-importAndInstall *package_file*

> Import the specified package archive or patch set, and install it in the path specified by -destination.
>
> This option lets you modify an existing installation. To add a new installation, use -installBaseline.

-importPatchSetAndUnlock *input_patchset_package_file*

> Import the specified patch-set package, install it in the installation in the path specified by -destination, and unlock it.
>
> This option allows packages specified in the patch set to be updated. If you prefer that the packages stay locked, use -importAndInstall instead.

-installBaseline *package_id* **[/** *package_version* **]**

> Add a new installation and initialize it with the specified baseline package or patch set. The installation will be added in the path specified by -destination.
>
> The *package_id* argument can simply contain the package ID, or it can include the package version, in which case you separate the ID and version by a slash: *package_id* / *package_version*.
>
> This option performs the same function as the Add Installation option in the GUI.

-installPackage *package_id* **[/** *package_version* **] [,** *package_id* **[/** *package_version* **] ...]**

> Add the specified package to the active installation.

> ---
>
> ⚠️ **CAUTION:**
>
> Don't include any spaces in the argument.
>
> If you wish to install multiple packages, specify all of them with a single -installPackage option. Don't use a separate -installPackage option for each package.
>
> ---

-list

> List the packages available in any remote or local repositories.
>
> The output of this option includes packages that you can't currently install. To see packages that are available to you, use −listAccessible.

-listAccessible

> List all accessible packages.
>
> This option lists all of the packages that are available to you, including any that have already been installed.

-listInstalled

> List all packages in the active installation.

-listInstalledRoots

> List all root packages (i.e. packages explicitly installed by the user) in the active installation.
>
> The list doesn't include packages that the QNX Software Center has installed to satisfy dependency requirements.

-listLicenseKeys

> List all license keys deployed to the user and indicate the activation status of each.

-listProfiles

> List all installations.

-listQuery *query*

> List all available packages that match the specified query.
>
> You can use this option for listing packages that match specific parameters (for instance, all packages that contain a specific library). The *query* argument takes arguments specified in the p2ql query language. Visit **https://wiki.eclipse.org/Equinox/p2/Query_Language_for_p2** to learn more about p2ql.
>
> Here are some examples:
>
> List only the latest version of each available package:
>
> ```
> qnxsoftwarecenter_clt -listQuery 'Q:latest()'
> ```
>
> List only the latest version of the **com.qnx.sdp.target.screen.base** package:
>
> ```
> qnxsoftwarecenter_clt -listQuery "Q:latest(x | x.id == 'com.qnx.sdp.target.screen.base')"
> ```
>
> List all versions of the **com.qnx.sdp.target.screen.base** package:
>
> ```
> qnxsoftwarecenter_clt -listQuery "Q:select(x | x.id == 'com.qnx.sdp.target.screen.base')"
> ```
>
> List the latest package that provides "ifconfig":
>
> ```
> qnxsoftwarecenter_clt  -listQuery "Q:latest(iu | iu.providedCapabilities.exists(p |
>     p.name == 'ifconfig'))"
> ```
>
> 💡 If you run this option from a Unix shell, you need to escape any & and * characters in the *query* argument.

-listUpdates

> List available updates.

-mirror

> Mirror package metadata from the remote myQNX server into the local cache.
>
> Use this option at least once before performing any remote operations such as `-list` or `-installPackage`. See the Description for more information.

-myqnx.user *user*

> Your myQNX username.
>
> See the Description for when to use `-myqnx.user` and `-myqnx.password`.

-myqnx.password *password*

> Your myQNX password.
>
> ---
> **ALERT:** When you use this option, the QNX Software Center saves your password as cleartext in its log file. To avoid this issue, pass the option through an options file (see @*options_file*) rather than on the command line.
> ---

-offline

> Don't synchronize with the remote myQNX server.
>
> If you can't directly connect to the server, you must use `-offline` with the `-mirror` option. That will load packages from the **dropins** directory. You must import the packages there beforehand.

-setDebugSymbolsEnabled=true|false

> If true, list or install packages that contain only debug symbols, in addition to any other packages that the user is entitled to. The default is true.

-setExperimentalEnabled=true|false

> If true, list or install experimental packages, in addition to any other packages that the user is entitled to. The default is true.

-setPolicy=*update_policy*

> Use the specified installation update policy.
>
> You can specify one of the following policies:
>
> **liberal**
>
> > Install all updates during the initial installation. When subsequently installing a package, install the latest version of each dependent package.
>
> **conservative**
>
> > Install all updates during the initial installation. When subsequently installing a package, apply the following rules to dependent packages:
> >
> > • If a dependent package isn't currently installed, install the latest version.

- If a dependent package is already installed, use the existing version unless the package being added requires a newer version.

ultraconservative

Don't install updates during the initial installation. When subsequently installing a package, install the earliest version of each dependent package that is not already installed.

If you omit the `-setPolicy` option, the QNX Software Center command line uses the conservative update policy by default. For more information on update policies, including examples, see "Edit installation properties."

-reportArch *target_architecture*

Set the target architecture for the report generated by the `-reportImportQScan` option.

The *target_architecture* argument specifies one of the architectures supported by the QNX Software Development Platform (SDP):

- `aarch64le`
- `armle-v7`
- `x86`
- `x86_64`

-reportFormat *format*

Format the properties in the manifest report.

You can set the *format* argument to one of the following values:

- `csv` — Format properties as comma-separated values (i.e. in target manifest format).
- `useinfo` — Format properties as field-value pairs, with one property per line.

This option lets you format reports generated by `-fileInfo`, `-fileSearch`, or `-reportImportQScan`.

If you omit this option, the QNX Software Center generates the report in target manifest format.

-reportImportQScan *qscan_manifest*

Import the specified target manifest file and generate a report.

This option lets you create a report for a target manifest generated through a tool like Qscan. See "Generate a manifest" for more information.

The report compares the scanned properties of each file in the target manifest to the properties defined for that file on the myQNX server. If the QNX Software Center can't connect to the server, the report compares the scanned properties to the properties data for the active installation. (You can use `-destination` to choose the active installation.) In either case, the Errors field in the report flags any detected mismatches. See "Manifest report fields" for more information.

You can use `-reportArch` to specify the target architecture for the report, `-reportFormat` to format the report data, `-reportPresentation` to apply a template to the report, and `-reportSaveAs` to save the report to a file.

-reportPresentation *template*

Apply the specified report template to the manifest report.

This template:	Lets you identify:
subsystem	Target subsystems that may require a production licensing agreement
packages	Properties (name, ID, version, etc.) of each file's parent package
targetscan	Path, size, architecture, build ID, parent package, and other properties of each file
notice	Various compliance properties (end-user license, OSS compliance, eligibility for redistribution, etc.) of each file
all	All available properties of each file

This option lets you apply a template to reports generated by -fileInfo, -fileSearch, or -reportImportQScan.

If you omit -reportPresentation, the QNX Software Center uses the all template by default.

To see which fields are included in each template, launch the QNX Software Center GUI, click **Create Reports**, click **New Report**, and then click the **Report Template** dropdown menu.

-reportSaveAs *path*

Save the manifest report to the specified file.

This option lets you save reports generated by -fileInfo, -fileSearch, or -reportImportQScan.

If you omit this option, the QNX Software Center writes the report output to **stdout**.

-selfUpdate

Update to the latest available version of the QNX Software Center.

The QNX Software Center will attempt to update itself before executing any other specified options. See also the -version option.

-setRootAll

Mark all installed packages as root packages

⚠️ **CAUTION:** The -setRootAll option is deprecated. We recommend that you avoid using it.

-syncLicenseKeys

Retrieve registered license keys from the user's myQNX account.

This option saves you the effort of having to add license keys manually. After you invoke this option, you can invoke -activateAll to activate your products.

This option requires the myQNX authentication options, `-myqnx.user` and `-myqnx.password`.

-uninstallBaseline *install_path*

Remove the installation located in the specified path.

This option performs the same removal operations as the `-cleanInstall` option and the **Remove Installation** command in the GUI.

> ⚠️ **CAUTION:** This option deletes all of the files in the specified installation.

-uninstallPackage *package_id* **[/** *package_version* **] [,** *package_id* **[/** *package_version* **] ...]**

Uninstall the specified package from the active installation.

> ⚠️ **CAUTION:**
>
> Don't include any spaces in the argument.
>
> If you wish to uninstall multiple packages, specify all of them with a single `-uninstallPackage` option. Don't use a separate `-uninstallPackage` option for each package.

-unlockPatchSet *patchset_id*

Unlock the specified patch set.

This option allows packages specified in the patch set to be updated. It affects the installation in the path specified by `-destination`.

-updateAll

Update all packages to the latest version available.

This option affects the installation in the path specified by `-destination`.

-updatePackage *package_id* **[/** *package_version* **]**

Update the specified package to the specified version.

If you omit *package_version*, the package will be updated to the latest version available.

This option affects the active installation.

-useInstallationVariant *key=value*

Set the installation variant according to the specified license type.

For example, if you have a build server license key and are installing QNX SDP 7.1 or later, specifying `-useInstallationVariant lic.buildserver=true` will automatically add and activate that key for your installation; you don't need to use `-addLicenseKey`.

-verifyInstallation

> Verify the active installation.
>
> If the verification fails (for instance, the installation contains missing or modified files), the QNX Software Center sets an exit code of 4. Otherwise, it sets an exit code of 0.

-version

> Display the version of the QNX Software Center installed on this machine. See also the -selfUpdate option.

@*options_file*

> Read command-line arguments from the specified file. The file must contain one argument per line.

Description

The qnxsoftwarecenter_clt script lets you run the QNX Software Center from the command line. The script uses the same infrastructure as the QNX Software Center's GUI: everything you do at the command line is reflected in the GUI, and vice versa. The script is located under the root of your hard drive in the **\QNX\QNX Software Center** directory (for instance, **C:\QNX\QNX Software Center** in Microsoft Windows).

myQNX authentication and cache synchronization

Several command-line options need to communicate with the remote myQNX server, including those that install or update packages, add or synchronize license keys, or activate products. When using any of these options, include the myQNX authentication options, -myqnx.user and -myqnx.password.

The -mirror option, which synchronizes the contents of your local cache with package metadata on the remote myQNX server, also requires the myQNX authentication options. You should use -mirror at least once before using options that need to communicate with the remote myQNX server. If you are writing installation scripts, avoid invoking -mirror multiple times as it can reduce performance significantly.

Installation directory

The -destination option lets you specify the installation directory for any commands that install, uninstall, or query packages on disk. You must include it when using the -installBaseline option. In most cases, you need to use it only once, before calling -installBaseline. All subsequent commands will assume the same destination. You can omit -destination altogether if you work with only one installation. That said, we recommend that you always use it if you are writing scripts.

License keys

Before installing a licensed product, you must add the license key for that product. To add a license key manually, you can use -addLicenseKey, which also attempts to activate the product. Alternatively, you can use -syncLicenseKeys, which automatically synchronizes the license keys on your workstation with those deployed to you on the remote server. After using -syncLicenseKeys, use -activateAll to activate your products.

> 💡 If you are using the QNX Software Center offline, you must use the GUI to activate your products. See "Perform manual product activation" for instructions.

Patch sets

The `qnxsoftwarecenter_clt` script supports several options for working with patch sets, which allow you to install a specific set of products, patches, and updates. See "Export and import patch sets" for more information.

Proxies

In most cases, you can ignore the proxy options. You need them only if your workstation sits behind a proxy server. If your proxy server supports HTTP and HTTPS, you can omit the `http` and `https` prefixes and issue a single set of proxy commands for both HTTP and HTTPS. Otherwise, issue a set of `-http` commands for HTTP and a separate set of `-https` commands for HTTPS.

If you encounter proxy connection issues, see the QNX Software Center *Technotes*: http://www.qnx.com/qsc_technotes.

Reports

The `qnxsoftwarecenter_clt` script supports several options (`-fileInfo`, `-fileSearch`, `-reportImportQScan`, etc.) for creating manifest reports. These reports can help you perform compliance audits by indicating, for example, which subsystem a file belongs to, whether you can redistribute the file at runtime, or whether the file contains open source software. For information on using these options, see the individual option descriptions and the "Using the command line" section of the "Manifest Reports & File Lookup" chapter.

Directory structure

The QNX Software Center writes data to the following directories in your home directory:

This directory:	Contains:
.qnx/swupdate/	User settings and cache.
.qnx/swupdate/swupdate_profiles	Package installation metadata. The QNX Software Center needs this directory to see your installations and installed packages.
.qnx/license/	Licenses for the QNX Software Development Platform (SDP) and other products. The QNX Software Center uses this directory to store information about your product activations; if anything happens to the directory, you'll no longer be able to use the products. If this occurs, contact your license administrator, who can issue a special request to have your products deactivated (which will then allow you to reactivate them).
.qnx/qconfig/	Metadata that the QNX Momentics IDE and other tools use to find the list of installed baseline packages.

Chapter 8
Manifest Reports & File Lookup

Using the QNX Software Center, you can create *manifest reports* for any QNX Software Development Platform (SDP) installation, or for any QNX target image that resides on your target hardware or development host.

Overview

A *manifest* is a plain text file that lists metadata (i.e. properties) for the files found in an installation or a target image. This information can, for example, help you determine:

- which subsystem a file belongs to
- whether the file comes from BlackBerry QNX or another source
- whether the file is modified or corrupted
- whether the file is eligible for distribution in a runtime system
- whether the file contains open source software (OSS)
- the software maturity level of the file (See Package status)

The information provided by a manifest can help you perform software compliance audits. For example, a manifest can help you identify the subsystems on your target that require a production licensing agreement, or the licenses that apply to various components.

The QNX Software Center can generate or import manifests and then export them into useful reports. To save you time, the QNX Software Center provides several predefined report templates. You can use any template as is, or you can quickly customize it by adding, removing, sorting, or rearranging metadata fields. As an added benefit, the QNX Software Center exports reports in CSV (comma-separated values) format, so it's easy to process them with your existing data analysis and office productivity tools, such as Microsoft Excel.

You can create a manifest report for:

- an installation or directory on your development host
- a QNX target image that resides on a physical target
- a QNX target image that resides in a directory on your development host—this option is useful, for example, when target hardware isn't yet available or if you're evaluating the QNX Software Development Platform (SDP).

The myQNX server maintains the latest properties (release status, OSS compliance, etc.) for each file supplied by BlackBerry QNX. To create some reports, the QNX Software Center compares these server-defined properties to the scanned properties of the actual file on your host or target. If a server-defined property and its corresponding scanned property don't match, the report flags the mismatch.

This matching can be critical to determining whether, for example, a file on your host or target has been corrupted. Thus, we recommend that you always use the QNX Software Center online when creating reports.

If you have to use the QNX Software Center offline, you can still compare server-defined properties against scanned properties by importing your manifest into the QNX Software Center Portal on the QNX website: https://www.qnx.com/swcenter/

Generate a manifest

When creating a report for an installation or directory on your host, or for a target image mounted on the host, you don't have to explicitly generate and import a manifest. The QNX Software Center can generate the manifest for you, automatically. See "Using the GUI" for details.

If, however, you want to create a report for an image on a physical target, you must first collect file metadata from the target and save it in a manifest. You can then import the manifest into the QNX Software Center, where you can quickly filter, sort, and arrange the metadata, and then export a report.

To generate a manifest for a target, you can use the Qscan tool, which is available for download from the QNX Software Center. See "Using Qscan to generate a target manifest" for details.

You can also generate a manifest by creating your own scripts or by assembling the manifest manually. See "Target manifest format" if you wish to take this approach.

> When generating a manifest or exporting a report, make sure that, at a minimum, it contains the following fields: NAME, BUILDID, DIGESTSHA512. Otherwise, you won't be able to import the file and generate a report from it.

Using Qscan to generate a target manifest

The Qscan tool, `qscan.sh`, lets you generate a manifest for the QNX OS image on your target system.

The manifests generated by Qscan are in the CSV-based target manifest format, so it's easy to process them with the Reports feature of the QNX Software Center or with your existing data analysis tools.

Installing Qscan

To install the Qscan tool onto your development host:

1. Launch the QNX Software Center and click **Install New Packages**.

2. In the **Available** tab, expand **QNX Software Development Platform**, then expand **Tools**.

3. Select the Qscan Tool package and click **Install**.

   The QNX Software Center installs the tool as `qscan.sh` under *installation_directory*/**target/qnx7**/*architecture*/**usr/bin/**, where *installation_directory* is the location of your QNX SDP installation and *architecture* is the processor architecture of your target (for instance, `x86_64`).

Generating a manifest

To run the Qscan tool on your target and generate a manifest:

1. Make sure that the `qscan.sh` script and the `find`, `openssl`, and `use` utilities all reside on the target. (The `qscan.sh` uses these other utilities to generate its output.)

2. Run `qscan.sh`, passing it the directory to search under. For example, this command tells `qscan.sh` to search the entire filesystem:

   ```
   ksh qscan.sh / > /tmp/targetscan.txt
   ```

By default, `qscan.sh` writes to standard output, but in this example, we are redirecting the output to a file called **targetscan.txt**.

If you also wish to redirect the standard error stream, specify a different file. Otherwise, the output file could be mixed with error messages that prevent you from importing it into the QNX Software Center.

Because `qscan.sh` calculates checksums, the output may take several minutes to generate.

3. Once you've generated the manifest, you can import it into the Reports feature of the QNX Software Center. See "Create a manifest report" for details.

Target manifest format

If you wish to generate your own manifests for a target system (i.e. you aren't using the Qscan tool), remember that the manifest must be in *target manifest format*. To see examples of this format, simply export a few test reports from the QNX Software Center GUI, as exported reports also use the target manifest format. Exporting a report from the GUI is easy and takes only a few minutes.

When generating a manifest, follow these simple rules:

- Arrange the metadata fields for each file in a single row and separate the fields with commas.
- Use the field names listed in the "Field Name in Manifest" column of the "Manifest report fields" table.
- Make sure that, at a minimum, your manifest includes these fields: NAME, BUILDID, DIGESTSHA512. Otherwise, you won't be able to import the manifest and create a report.
- Save the manifest as a plain text file.

You can also create your target manifest as a spreadsheet in an application like Microsoft Excel and export the spreadsheet as a CSV file. You can then import the CSV file into the Reports feature of the QNX Software Center.

Create a manifest report

The QNX Software Center lets you create reports from the GUI or from the command line (`qnxsoftwarecenter_clt`). Both approaches can produce reports from target manifests and both support a variety of predefined report templates. However, they also differ in some important ways. Notably, the GUI lets you generate a report directly from an installation or a directory on your host; it also gives you greater flexibility to customize the contents of your report. The command line, on the other hand, makes it easier to create a report that lists the latest server-defined properties for a specific file.

Using the GUI

To create a report from the GUI:

1. Launch the QNX Software Center.

2. From the Welcome screen, select **Create Report**.

3. Choose the type of report you wish to create:

Choose:	To:
Scan installation folder	Create a manifest report for the current installation.
Scan host folder	Create a manifest report for the contents of a directory on your development host or of a target filesystem mounted on the host.
Import file with scan data	Create a report based on a manifest that you have generated on your target system.
	Exported reports use the target manifest format, so you can also use this option to import an exported report for further processing.

4. From the **Architecture** dropdown menu, choose the processor architecture.

   If you chose **Scan host folder** or **Import file with scan data**, you must choose the processor architecture of your target image. If you chose **Scan installation folder**, you can choose whatever architecture you intend to target.

5. In the **Report Name** field, enter the name you wish to assign to the report.

6. From the **Report Template** dropdown menu, choose a predefined template.

   Each template presents a different set of metadata, grouped by a category such as subsystem or package name. Don't worry too much about choosing the best template at this point. You can subsequently customize the template or choose another template before exporting your report.

7. Click **Finish** when you're done. The QNX Software Center displays your report in the **Reports** tab.

   If you chose **Scan installation folder** or **Scan host folder**, the QNX Software Center may take a few minutes to generate checksums for the report.

8. Customize the report as desired. For instance, you can:

- right-click over any item and click **Apply Template** to select a different template
- right-click the column heading of any field to hide or sort its data, or to add another field
- drag the column heading of any field left or right to rearrange the order of fields
- right-click over any item and select **Group By** to group files according to a specific field, such as Subsystem or Errors (if the field isn't currently included in the report, the QNX Software Center will add it)

Note that sorting the contents of a field or rearranging the order of fields currently has no effect on an exported report.

> 💡 See "Manifest report fields" for a description of each field.

9. If you wish to share the report, click **Export Report**, specify the filename of your output file, and then click **Finish**.

Using the command line

The QNX Software Center's `qnxsoftwarecenter_clt` script supports several command-line options for creating reports. These options let you:

- look up the latest server-defined properties for a file on your host or target
- import a target manifest and create a report
- apply a template to a report
- format the data of a report
- save a report to a file

Look up the latest server-defined properties for a file

The myQNX server maintains the latest properties (release status, OSS compliance, etc.) for files supplied by BlackBerry QNX. See more details about status field in status. To look up these server-defined properties for a specific file or set of files on your host or target, use one of the following options:

- `-fileInfo` *path* — searches for any file on the myQNX server that matches the specified path (the path must point to an existing file in the host filesystem)
- `-fileSearch` *query* — searches for any file on the myQNX server that matches the specified file properties

If either option finds a matching file on the myQNX server, it creates a manifest report that lists the file's server-defined properties.

The `-fileSearch` option is especially useful when you know some, but not all, of the attributes of a target file. For instance, it can search by build ID, Secure Hash Algorithm 512-bit checksum, or package ID.

For more information, see the entries for `-fileInfo` and `-fileSearch` in the "Using the QNX Software Center Command Line" chapter.

💡 If you have to use the QNX Software Center offline, you can still look up the server-defined properties of a file by using the QNX Software Center Portal: https://www.qnx.com/swcenter/

Import a target manifest and create a report

Like the GUI, the command line lets you create a report for a target manifest generated by a tool like Qscan (see "Generate a manifest" for more information). To import the target manifest and create a report, use the `-reportImportQScan` option.

This option checks whether properties listed for each file in the target scan match those defined on the myQNX server. The Errors field of the report flags any detected mismatches. See "Manifest report fields" for more information.

Apply a template to a report

To apply a predefined template to a report generated by `-fileInfo`, `-fileSearch`, or `-reportImportQScan`, use the `-reportPresentation` option. It supports the same templates as the QNX Software Center GUI.

Format the report data

By default, `-fileInfo`, `-fileSearch`, and `-reportImportQScan` generate a report in the comma-separated target manifest format. If you prefer a more human-readable format, use the `-reportFormat` option and specify the `useinfo` argument, which formats each property in the report as a field-value pair.

Save a report to a file

By default, `-fileInfo`, `-fileSearch`, and `-reportImportQScan` write the report to standard output (**stdout**). To redirect the report to a file, use the `-reportSaveAs` option.

Manifest report fields

If you create your own manifests, use the field names listed in the "Field Name in Manifest" column to name your fields.

Field Name in Manifest	Field Name in UI	Description
General properties:		
BASENAME	Basename	The filename portion of the file's pathname.
		If this field is highlighted, the QNX Software Center found a matching file, but with a different basename. See the description of the Errors (VIOLATION) field for more information.
DIRECTORY	Directory	The directory in which the file resides.
NAME	Name	For the manifest generated by a scan, the basename of the file. For a report, the path of the file on the target or host, depending on whether the report is based on a target scan or host scan.
PUBLISHED _DATE	Published	Represents the publication date or the build date of the package. It is derived from various package timestamps.
TARGETPATH	Target Path	For a target scan, the path of the file on the target. For an installation scan, the path of the file from the root of the installation.
File attributes and checksums:		
ARCHITECTURE	Architecture	The processor architecture of the file.
BUILDID	Build ID	An ID, encoded in hex, that uniquely identifies the binary payload of the file. This ID doesn't change even when the binary is stripped.
		If this field is highlighted, the QNX Software Center found a matching file, but with a different build ID. See the description of the Errors (VIOLATION) field for more information.
DIGESTSHA512	Digestsha512	A Secure Hash Algorithm 512-bit checksum, encoded in base64, for validating the integrity of the file.
		If this field is highlighted, the QNX Software Center found a matching file, but with a different checksum. See the description of the Errors (VIOLATION) field for more information.

Field Name in Manifest	Field Name in UI	Description
DIGESTSHA512HEX	Digestsha512hex	A Secure Hash Algorithm 512-bit checksum, encoded in hex. This field contains values if you generate a target manifest with `qscan.sh`. If this field is highlighted, the QNX Software Center found a matching file, but with a different checksum. See the description of the Errors (VIOLATION) field for more information.
LINKS_TO	Links to	If a file is a symbolic link, this field lists the actual file that the link points to.
SIZE	Size	The size of the file, in bytes. If this field is highlighted, the QNX Software Center found a matching file, but of a different size. See the description of the Errors (VIOLATION) field for more information.
TYPE	Type	The file format (for instance, QNX/Elf).
Parent package properties:		
CATEGORY	Category	The category of the package to which the file belongs. The QNX Software Center uses the value of this field to organize packages into logical sets.
PACKAGE	Package	This field combines the information in the Package Id and Package Version Internal fields.
PACKAGEID	Package Id	The unique alphanumeric ID of the package to which the file belongs.
PACKAGEINVERSION	Package Internal Version	The internal version of the package to which the file belongs.
PACKAGENAME	Package Name	The name of the package to which the file belongs.
PACKAGENAMEV	Package Full	This field combines the information in the Package Name and Package Version fields.
PACKAGEVERSION	Package Version	The version of the package to which the file belongs.
PRODUCT	Product	The product to which the file's parent package belongs. Examples of products include QNX Software Development Platform, QNX Hypervisor, and QNX OS for Safety.
RELEASE	Release	Refers to the "tag.release" property of the package (package artifact).
FOLDER	Release Folder	Refers to the "server.folderName" property of the package (package artifact).

Field Name in Manifest	Field Name in UI	Description
Compliance properties:	💡	To access any licensing documents mentioned below, visit the Licensing Information page on the QNX website: http://www.qnx.com/download/group.html?programid=29179
CODE_PROVIDED	Code provided	Relevant only to QNX SDP 7.0 and related products. See the QNX Software Development Platform 7.0 License Supplement for more information.
END_USER_LICENSE	End user license	Indicates the end-user license terms or where to find them.
ENGAGEMENT_MODEL	Engagement model	Relevant only to QNX SDP 7.0 and related products. See the QNX Software Development Platform 7.0 License Supplement for more information.
OSS	OSS Compliance	If applicable, this field provides a link to the text of the OSS licenses.
REDIST	Redistributable	If set to true, this field indicates that the file is eligible for distribution as part of a target system.
STATUS	Status	The software maturity of the file (See Package status under "Deploy licenses to users").
SUBSYSTEM	Subsystem	Files that are eligible to be licensed for distribution are assigned to a runtime subsystem. This field identifies the runtime subsystem under which the given file can be licensed for production.
SUPPORT_PROVIDED	Support provided	Relevant only to QNX SDP 7.0 and related products. See the QNX Software Development Platform 7.0 License Supplement for more information.
TYPE_OF_SOFTWARE	Type of software	Relevant only to QNX SDP 7.0 and related products. See the QNX Software Development Platform 7.0 License Supplement for more information.
Errors:		
VIOLATION	Errors	Some report options compare the scanned properties of each file in a target manifest to the properties defined for that file by the myQNX server or by the current installation. If a mismatch is detected, the Errors field flags it. (See the table below for the comparisons performed by each report option.) When performing these comparisons, the QNX Software Center attempts to match files across the following properties: Basename, Architecture, Build ID, Size, and Digestsha512 checksum. If the match is inexact, the Errors field contains a weighted percentage figure to indicate how much the files match.

Field Name in Manifest	Field Name in UI	Description
		If all of the files in a directory match, the Errors field for that directory is empty. Otherwise, the field contains a message that indicates the number and type of errors found in the top level of the directory.

> 💡 Hover your pointer over the Errors field to see which properties do or don't match. Exported reports include this matched/unmatched data.

How file properties are compared

The following table summarizes how each report option compares file properties. (See the description of the Errors field for how property mismatches are reported.)

Option	Comparison
Scan installation folder	Compare the scanned properties of each file to the properties defined for that file in the package data of the current installation.
Scan host folder	Same as **Scan installation folder**.
Import file with scan data	Compare the scanned properties of each file to the properties defined for that file on the myQNX server. If the server can't be reached, compare the scanned properties with the properties defined in the package data of the current installation.
`-reportImportQScan`	Same as **Import file with scan data**.
`-fileInfo`	No comparison performed.
`-fileSearch`	No comparison performed.

File lookup

The sections that follow contain various methods that you can use to obtain status information (maturity level) on the files and packages, as well as other legal information. See "Deploy licenses to users" to learn more about the package status.

Data collection method

Single file lookup - by attribute

You can look up file information using known file attributes (i.e. DigestSHA512, BuildID, Package)

- Desktop GUI: Not supported.
- Client Portal: Supported.
- Command line: `-fileSearch` *query* — for more information, see "Using the command line".

Single file lookup - by file

You can look up file information for a specific file by providing the file itself.

- Desktop GUI: You may do this indirectly by scanning the parent directory. For more information, see "Using the GUI".
- Client Portal: Not supported
- Command line: `-fileInfo` *filepath* — for more information, see "Using the command line".

Single file lookup - by use

You can look up file information for a specific file by providing the output of the use `-i` command.

- Desktop GUI: Not supported.
- Client Portal: Supported.
- Command line: Not supported.

Multiple file lookup - host directory scan

You can lookup information on a set of files by scanning the host directory, or the mounted target directory.

- Desktop GUI: Supported. For more information, see "Using the GUI".
- Client Portal: Not supported.
- Command line: Not supported.

Multiple file lookup - target manifest

You can lookup information on a set of files by scanning the target file system using the Qscan tool, and then importing this report.

- Desktop GUI: Supported. For more information, see "Using the GUI".
- Client Portal: Supported.
- Command line: `-reportImportQScan` *Qscan report* — for more information, see "Using the command line".

Data Import Method

Reconcile with local installation data

- Desktop GUI: The desktop GUI will attempt to reconcile with local data, see "Using the GUI".
- Client Portal: Not supported.
- Command line: Not supported.

Reconcile with server data

- Desktop GUI: The desktop GUI will attempt to reconcile with server data, see "Using the GUI".
- Client Portal: Supported.
- Command line: Not supported.

Reporting Method

The QNX Software Center currently only supports custom and templated reports. Reports come in the form of a tree table where you can group, sort, hide, and show columns.

The columns, grouping and sorting of the templated reports are pre-configured. For more information, see "Manifest report fields".

- Desktop GUI: Supported. For more information, see "Using the GUI".
- Client Portal: Supported.
- Command line: `-reportPresentation` *template* — for more information, see "Using the command line".

Export Method

The QNX Software Center currently only exports reports in CSV (comma-separated values) format, so it's easy to process them with your existing data analysis and office productivity tools.

- Desktop GUI: Supported. Use the **Export** command in the **Report** view.
- Client Portal: Supported.
- Command line: `-reportSaveAs` *path* — for more information, see "Using the command line".

Appendix A
Glossary

To make best use of the QNX Software Center, we recommend that you become familiar with the following terms, which are used throughout the center's user interface and documentation.

Package

The minimal unit of software you can install. The QNX Software Center lets you work with a variety of package types, including baselines, board support packages (BSPs), patch sets, reference images, and source bundles. A package contains metadata (name, version, dependencies on other packages, etc.) and, in most cases, a payload of files.

Addon

An optional package that you can add to an installation.

Baseline

A package that serves as the basis of an installation; it must be installed before you can add any other packages (i.e. addons, updates) to the installation. A baseline has a name and a version, but has no dependencies on other packages.

Installation

Contains a baseline and any additional packages or updates that you've added to the baseline, all installed in a single directory tree. Your workstation can have multiple installations of the same or different baselines.

Package archive

Multiple packages packed into a single archive container such as **.zip**, **.tar**, **.tar.gz**, or **.tar.xz**. Typically used when you need to run the QNX Software Center offline.

Patch set

A package that acts like a recipe for installing packages. A patch set tells the QNX Software Center not only which packages to install, but also which version of each package to install. Patch sets enable all members of a team to standardize on the same software base; they also make it easier to reproduce issues across systems.

Repository

A database of software packages that the QNX Software Center uses to display the packages you can install or update.

Root package

A package that you've explicitly added to an installation (as opposed to a package that the QNX Software Center has installed to satisfy dependency requirements).

Status

The Status field identifies the maturity of the software at a file level, although it is usually defined at package level so that all included files have the same maturity. See Package status under "Deploy licenses to users".

Index

A

activation of licenses 93
activation of products
    deactivation 28, 78
    history 26
    issues, resolving 90
    manual 71
    monitoring 26
addon
    defined 119
administrators
    allocating 16–17, 21
    deallocating 17
allow foreign host packages
    choosing 53
asterisk icon
    meaning of 47

B

baseline
    defined 119
build server 88
    license key, activating 102

C

cache for packages
    clearing 52
clear saved credentials feature 80
command line
    for QNX Software Center 93
        help, displaying 97
content filter 46

D

deactivation of products 28, 78
    See also activation
deallocate
    administrators 17
debug symbols
    installing 52, 59, 99
dependencies
    viewing 48

disk space
    freeing 52, 85, 93
documentation
    release notes 55
        QNX Software Center 35
downgrades
    installing 62
dropins directory
    clearing 85, 93

E

Edit Installation Properties dialog 51
email
    notification 56
empty
    installing 42
end user license agreement 48
    viewing 48
        See also license
error log 90
experimental packages
    installing 53, 99
export function
    myQNX license manager 21
export patch set, *See* patch set

F

field
    status 20, 119
files
    properties of 49
        See also manifest reports
filter packages, *See* package, filtering
floating license, *See* license
forget password feature 80

G

glossary of common terms 119
group by
    custom 54
    release 54
    release_folder 54
    status 54

T

tabs
    restoring access 47
target architecture
    choosing 53
target manifest, *See* manifest reports
technical support 8, 90
terminology used by QNX Software Center 119
troubleshooting
    activation issues 90
    inaccessible packages 86
    installation issues 90
    log files 90
    login issues 80
    technotes 90
    user interface issues 47
typographical conventions 6

U

uninstall
    installation 40, 102
    package 63, 102
    QNX Software Center 41

updates
    installing 60, 102
        automatically 52
    listing 99
    of QNX Software Center 101
    update policy
        setting 51, 99
upgrades (product collection feature) 23
user
    revoking 27
user interface
    QNX Software Center 45
        resetting 47, 50

V

variant
    installation 102
verify installation 84, 103

W

Welcome screen
    accessing 37
window layout
    resetting 47, 50