FORUM NOKIA

Getting Started with the Nokia Qt SDK

Version 1.0; April 27, 2010

Qt

NOKIA

Contents

Change history

April 27, 2010	Version 1.0	Initial document release

1 Download and set up the right tools

The Nokia Qt SDK is an easy-to-install and use development environment for Qt application developers targeting mobile platforms such as Symbian and Maemo™. There is only one installation file to download. The Nokia Qt SDK Setup wizard will guide you through the installation process.

Note: Support for the Nokia N900 device is expected to be available once the PR1.2 firmware upgrade for the Nokia N900 firmware is released.

1.1 System requirements

Download the correct version of the Nokia Qt SDK into your operating system:

- Windows XP Service Pack 2 / Windows Vista / Windows 7
- (K)Ubuntu Linux 7.04 32 bit and 64 bit with the following:
    - g++
    - make
    - Libglib2.0-dev
    - libSM-dev
    - libxrender-dev
    - libfontconfig1-dev
    - libxext-dev
    - libfreetype6-dev
    - libx11-dev
    - libxcursor-dev
    - libxfixes-dev
    - libxft-dev
    - libxi-dev
    - libxrandr-dev
    - libgl-dev and libglu-dev (if using QtOpenGL)

Note that you need 4 GB of free disk space for the installation.

Download the SDK from http://forum.nokia.com.

Figure 1: The setup wizard opens.

Note: This may take a while on slower PCs.

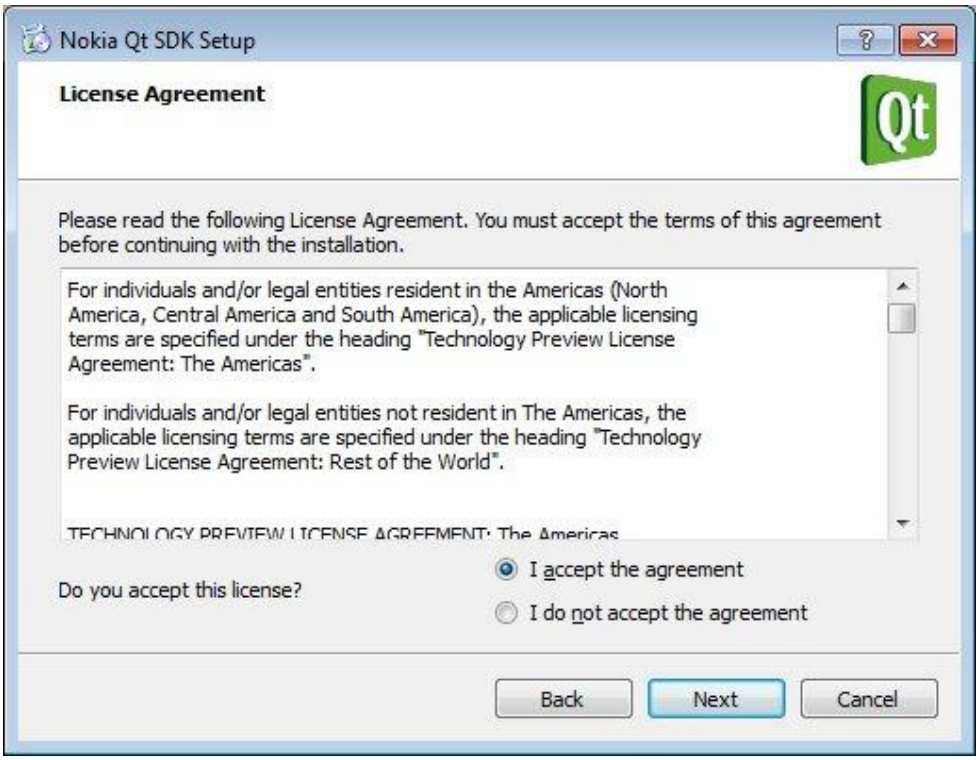

Figure 2: Accept the licence agreement.

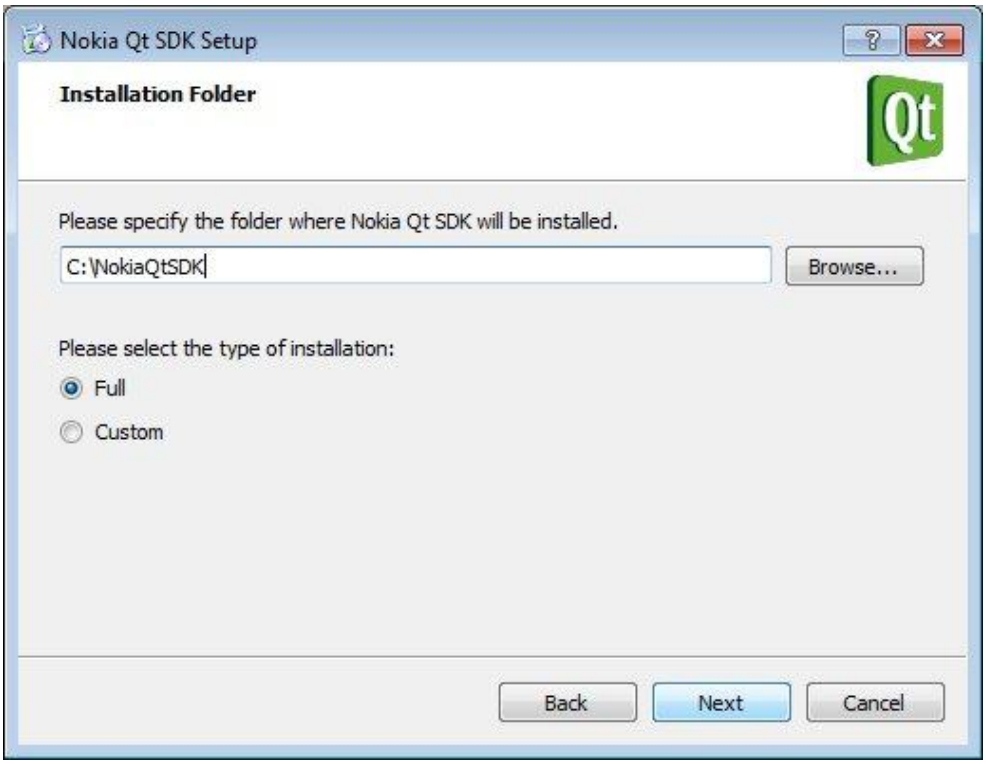

Figure 3: Select the installation directory.

After the installation is complete, launch Qt Creator as prompted. You are also prompted to open Symbian and Maemo development tutorials, but skip them for now. You can find them later in the `<NokiaQtSDK installation folder>\Symbian\readme` and `<NokiaQtSDK installation folder>\Maemo\readme` directories.

Basically no further configuration should be needed because you do not configure the Nokia Qt SDK with any existing Symbian S60 or Maemo SDKs. You will only need to make settings for the device connection later. Now, continue with the project creation.

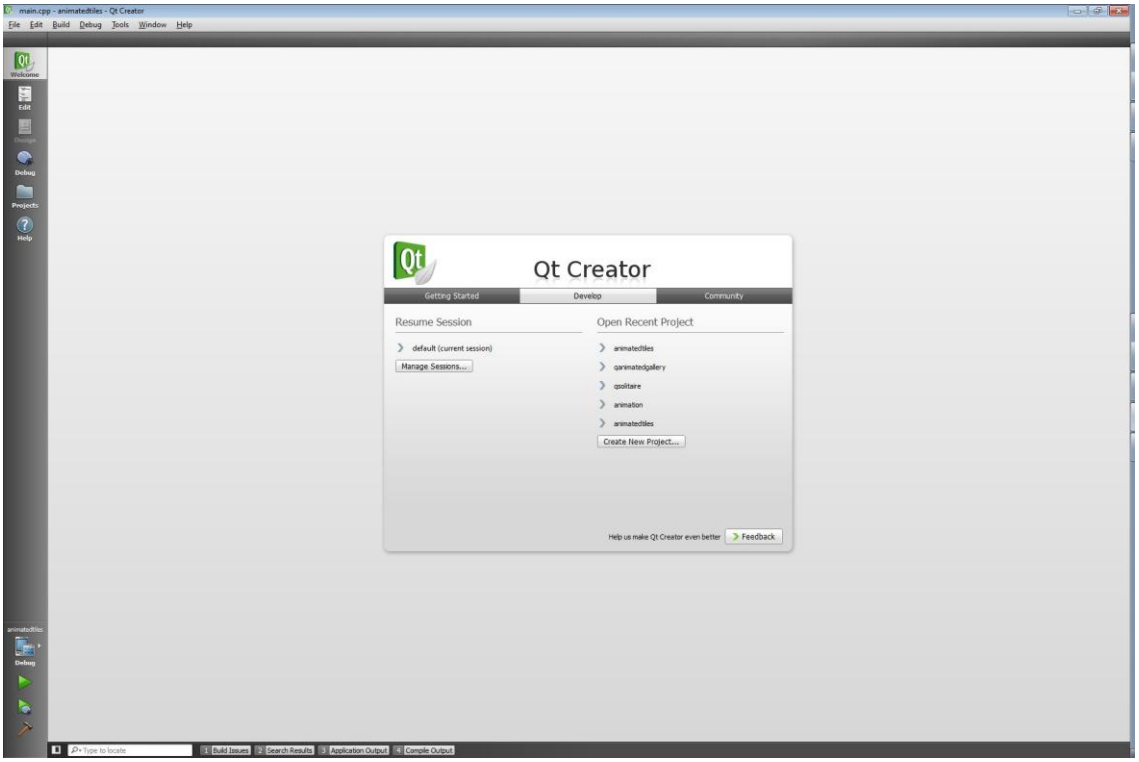

Figure 4: Qt Creator has been successfully launched.

2 Create your first application

2.1 Import a project

Qt Creator opens in a Welcome view. Test that your installation is successful by opening an existing example application project.

Figure 5: Expand the **Choose an example...** drop-down list and select **Animated Tiles**.

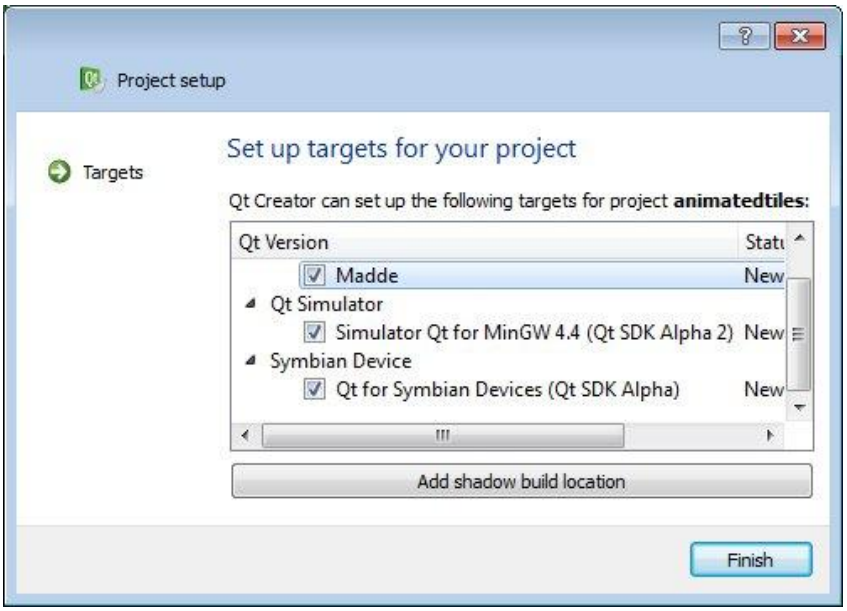

Figure 6: Select the desired targets.

Select at least **Qt Simulator** and one of the mobile targets, Maemo Device or Symbian Device, depending on the device you are using.

Note: If you want to add targets later, this is possible from the **Projects** view, which you can open from the left-hand toolbar.

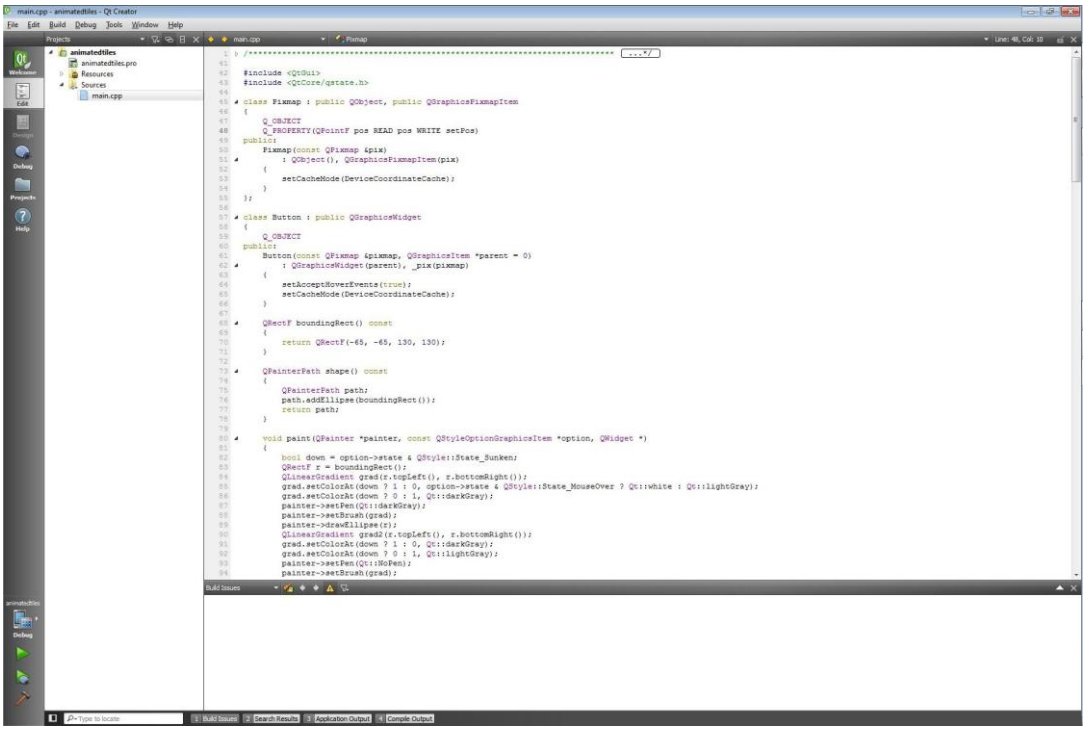

Figure 7: The project opens. Click on any of the source or header files to view and edit the code.

Click the **Build** button in the bottom left-hand toolbar (shown in Figure 8). The Build progress bar in the toolbar becomes green once the project has been successfully built. If you want to see the compilation progress, open the Compile Output log (shown in Figure 9).

Figure 8: **Target**, **Run**, **Debug**, and **Build** buttons appear in the left-hand toolbar.

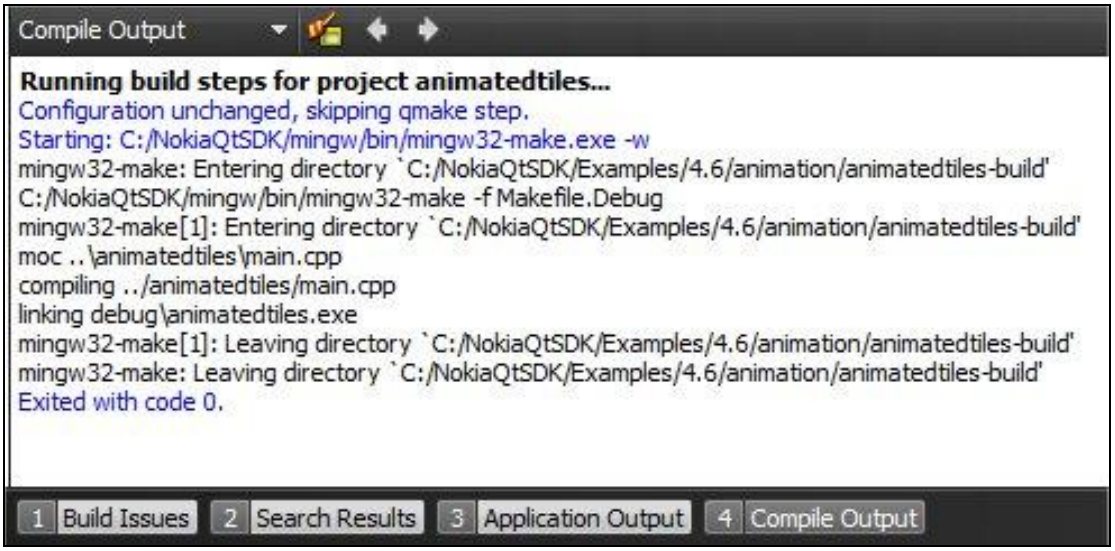

Figure 9: You can monitor the compilation progress from the **Compile Output** log.

3 Test your application

Next, run the application in the simulator. Click the device icon in the left-hand toolbar to select the target. A pop-up menu opens (see Figure 10). Click **Qt Simulator** as a target (the active target has a blue background colour). If you have multiple projects open in Qt Creator, make sure the correct project is active. You can change it from the Project bar of the pop-up menu.

Figure 10: The target selection menu appears.

Click the green **Run** button (see Figure 8).

You can see the application in the simulator (see Figure 11).

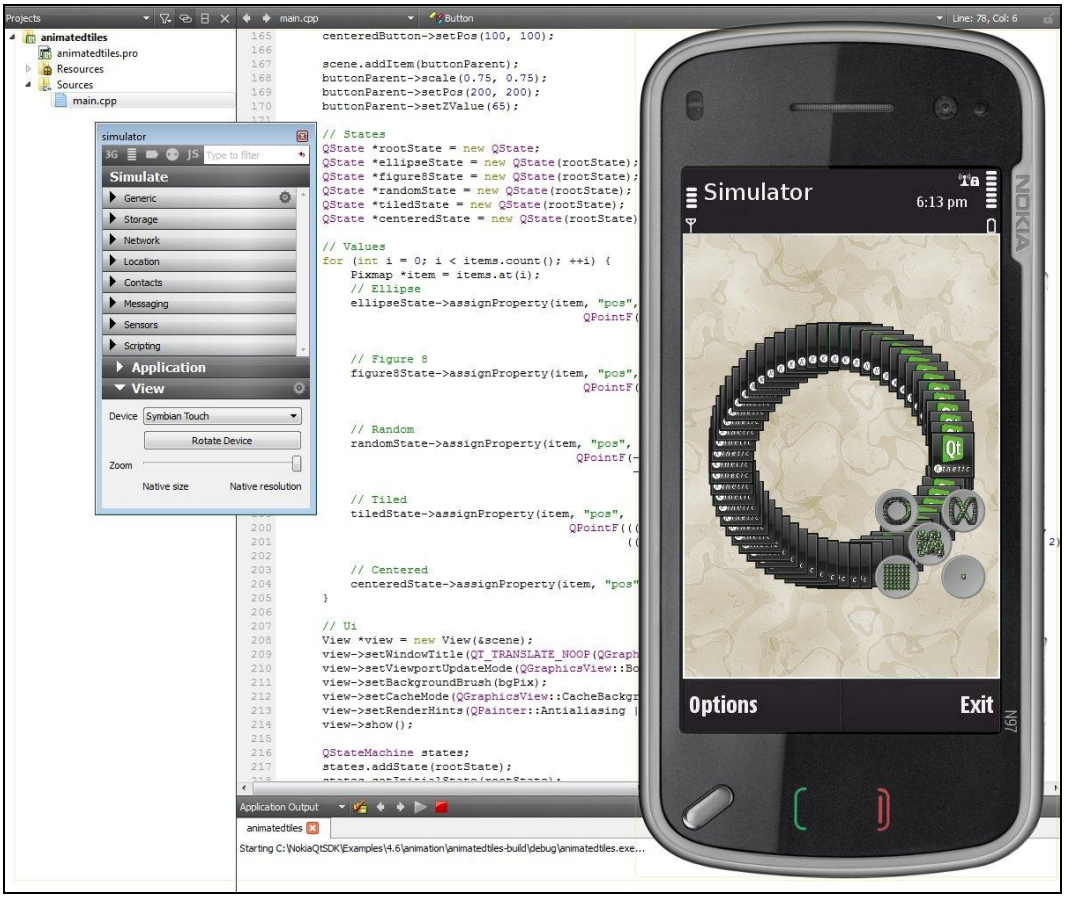

Figure 11: The application is running in Qt Simulator with the Symbian Touch configuration.

Note: The Symbian and Maemo platform macros are not supported by the simulator. This means that, for example, screensize settings made for Symbian and Maemo targets do not apply in the simulator. Also the Symbian or Maemo menu or **Exit** controls do not currently work in the simulator.

Next to the simulator there is a specific view to change settings. Under the **View** tab you can, for example, toggle the orientation (Rotate Device) or choose from the various Symbian and Maemo configurations (Device). You can also simulate various mobile functions and even create your own scripts. For more information on the simulator features, see **Qt Creator Help**.

Now you have successfully run your application on Qt Simulator. Let's test it on a real device.

4 Deploy the application on the device

This section describes building the application for the Symbian GCCE target and deploying the application on the device. Building for GCCE itself can be done without additional installations but to conveniently deploy and debug the application on the device, Qt 4.6.2 libraries and a debugging application need to be installed on the device. The latter also requires some configurations in Qt Creator.

Follow this documentation carefully, step by step (especially to first set up the device and then the host PC).

Note: If you wish to debug the application on a Nokia N900 device, follow the instructions provided by the SDK (for example, in Windows OS, check <`NokiaQtSDK installation folder>\Maemo\readme`). Note however that full support for the Nokia N900 device is expected to be available only after the PR1.2 firmware upgrade for the Nokia N900 firmware is released.

4.1 Prerequisites

Debugging is only supported in Windows OS (32 bit and 64 bit supported). Additionally, you need:

- A Symbian device with a USB connect cable (to fully run the example application, an S60 5th Edition Touch UI device is needed)
- Nokia Qt SDK
- Ovi Suite or Nokia PC Suite

4.2 Set up the Symbian device

Connect the Symbian device to the PC using the USB cable delivered with your device. Select PC Suite mode. If you haven't previously used the device with Ovi Suite or PC Suite, all the needed drivers are first installed automatically. This takes approximately one minute.

4.2.1 Install Qt 4.6.2 on the device

For this you need a programme that can install `.sis` files (Ovi Suite or Nokia PC Suite). To install Qt, use the shortcut that Nokia Qt SDK has created in the Start menu (**StartMenu -> Nokia Qt SDK -> Symbian -> Install Qt to Symbian device**) and follow the steps on your screen.

Alternatively, you can find the Qt installation package using Windows Explorer. Navigate to <`NokiaQtSDK installation directory>\Symbian\sis` and select `qt_installer.sis`.

Note: In some combinations Ovi Suite has problems installing .sis files. However, you can copy the file (<`NokiaQtSDK_install_path>\Symbian\sis`) to the device in USB storage mode or send the file via Bluetooth and install it with the help of the device file browser.

4.2.2 Install and configure the TRK debugging application on the device

Use the shortcut in the Start Menu to install the TRK application (**StartMenu->Nokia Qt SDK -Beta release->Symbian->Install TRK to Symbian device**).

Alternatively, you can navigate to <`NokiaQtSDK installation directory>\Symbian\sis` and select `s60_5_0_app_trk_3_1_2.sisx`.

Once the installation is complete, navigate to the folder where all the installed applications go on your Symbian device (Applications, Install, etc.) and start the TRK application.

Figure 12: TRK application on the Symbian device

Figure 13: The TRK application is now open but not yet connected.

Say 'No' to the Bluetooth connection prompt. Then open **Options** | **Settings** and change the connection type to USB (see Figure 14).

Figure 14: Open **Options** | **Connect** and change the connection type to **USB**.

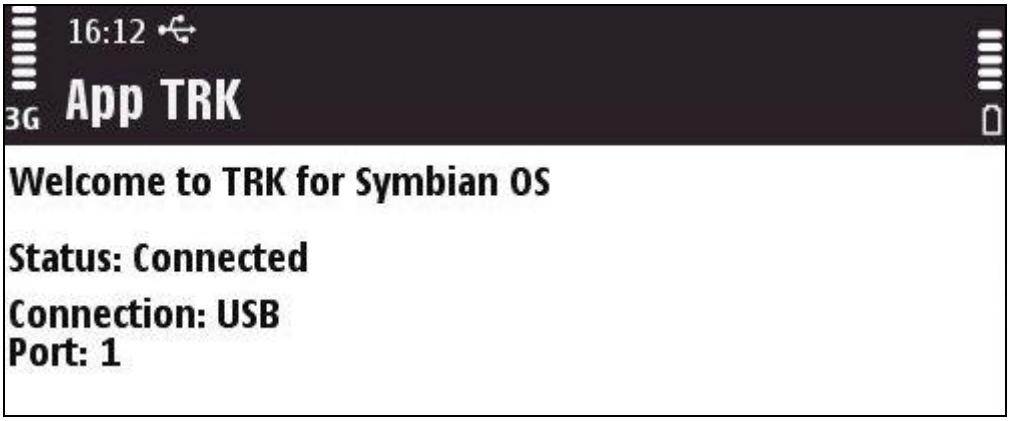

Figure 15: The TRK application is now connected to the PC.

Now the Symbian device has been configured (see Figure 15).

4.3 Set up Qt Creator

In Qt Creator, go to the **Projects** view from the left-hand toolbar. Then go to the **Symbian Device | Build** tab (see Figure 16).

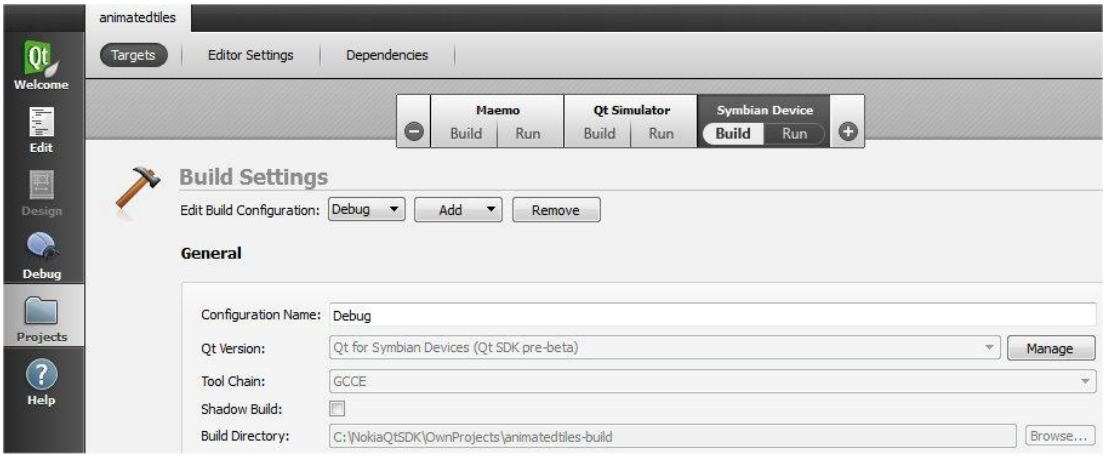

Figure 16: The Symbian Device **Build Settings** view is opened.

Note: If you did not add a Symbian target in the project import/creation phase, this tab is not visible. However, you can add it now by pressing '+' and selecting **Symbian** from the drop-down list.

Make sure that you have **Debug** selected in the **Edit Build Configuration** drop-down menu.

The Qt version should be **Qt for Symbian Devices (Qt SDK Beta)**. If it is not, you can change it by clicking **Manage**.

Now go to the **Symbian Device | Run** tab (see Figure 17). Make sure that your active project is in the **Run** configuration, in this case 'animatedtiles on Symbian device'.

Also check that the **Device on Serial Port** field shows information on your device model and the COM port.

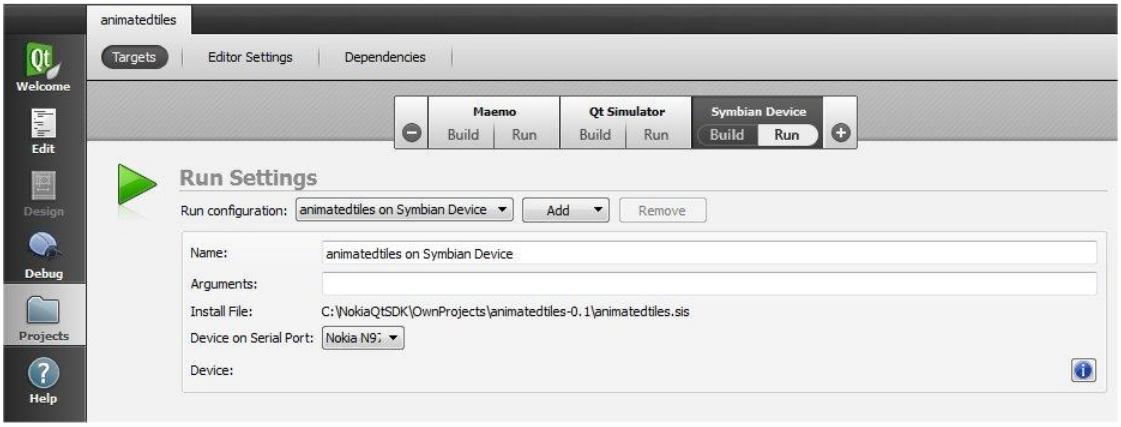

Figure 17: The Symbian Device **Run Settings** view is opened.

All the needed configurations are now done.

4.4 Build the application for GCCE Debug build

Select Symbian Device as a target in the left-hand toolbar (shown in Figure 8). Make sure that you have selected Debug build and the correct project is active (in this case, 'animatedtiles on Symbian Device').

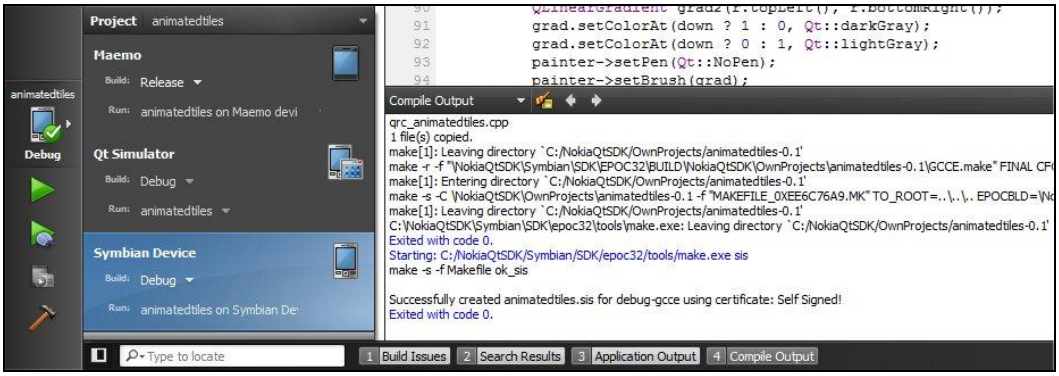

Figure 18: The Symbian Device **Debug** build is selected. **Compile Output** shows a successful build.

You can follow the build progress from the **Compile Output** tab and also see that a self-signed `.sis` installation file has automatically been created.

Note:	You can create the **Release** build of the application in a similar fashion.

4.5 Deploy and run the application on the device

You are ready to deploy and run the application on the device. Click the green **Run** button.

Optional: To add breakpoints and examine the code line by line, click the **Debug** button instead. See Qt Creator Help for more information on how to add breakpoints in Qt Creator.

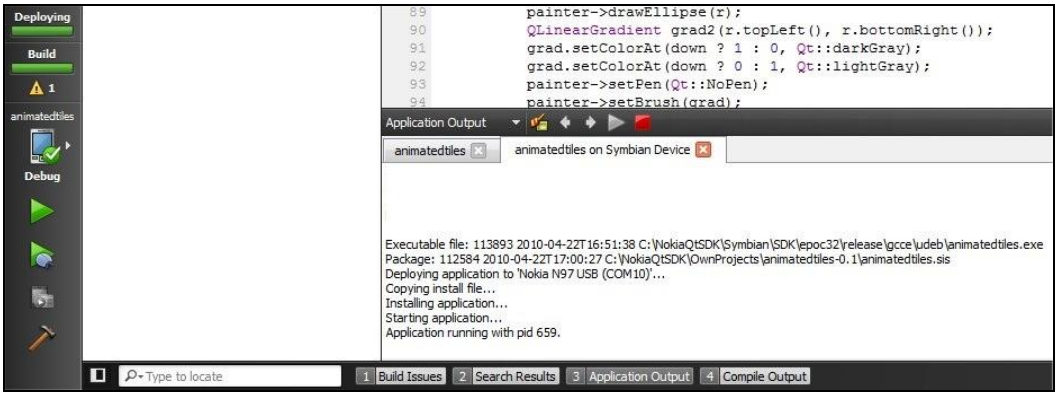

Figure 19: The debug build of the application is running.

From the **Application Output** you can see that the installation file has been deployed and installed on the device, and the application has started.

Figure 20: The application in its original screensize is running on the device.

4.6 Optimise the code for a mobile device

You can see that the application is not scaling to the whole screen (Figure 20).

To fix this, go to the **Edit** view in Qt Creator and open the `main.cpp` file. On line 214 you will see that the application is defined to take the window size (which works nicely in the desktop builds):

```
view->show();
```

In Symbian and Maemo, a better choice is to use `showMaximize()`. This scales the application to fit the entire application area while still keeping the status and control panes.

Change the line as follows:

```
view->showMaximized();
```

Now repeat the steps described in Section 4.4 and Section 4.5 to build and run (actually you can directly click the **Run** button if you are sure about the changes you have made).

Congratulations — you now have a fully scalable application running on your Symbian device.

Figure 21: A fully scalable application is now running on the device.

Note: Should you need the `animatedtiles.sis` installation package later, it can be found in the `<NokiaQtSDK installation directory>\Examples\animation\animatedtiles` directory.