

Qtopia® Open Source

FIC Neo SDK

Developer Quickstart Guide

Disclaimer

WARNING: This device's primary purpose is a mobile telephone application development platform and is not intended as a primary end-user mobile device. This device is not assured to be bug-free and should not be used in safety critical environments or where confidentiality needs to be preserved.

Table of Contents

1	SDK INSTALLATION	3
1.1	SDK Linux Installation.....	3
1.2	SDK Windows Installation.....	3
2	SETUP ON X86 FOR THE FIC NEO	3
2.1	Phonebounce Mode	3
2.2	VMWare Player	4
3	FLASHING THE NEO FROM THE DEVELOPMENT ENVIRONMENT	4
3.1	Update via the USB Cable	5
3.2	Low-level Flash using usbflash Utility.....	5
3.3	Update via the mini-SD Flash Process	Error! Bookmark not defined.
4	SETUP, DEVELOP AND CREATE QTOPIA PACKAGES	5
5	REMOTE DEBUGGING FOR NEO	6

6	NEO CONSOLE ACCESS	7
7	INSTALLING QTOPIA SOURCE CODE INTO THE SDK.....	7
8	BUILDING A NEO FLASH IMAGE.....	7
9	TRANSFERRING DOCUMENTS TO AND FROM THE NEO	8
10	SDK COMMAND UTILITY	9

1 SDK Installation

1.1 SDK Linux Installation

1. Download the latest version of VMWare® Player from <http://www.vmware.com>.
2. Install VMWare Player in accordance with the instructions in [VMWare Documentation](#).
3. Insert the Qtopia® SDK CDROM.
4. Go to the mount point of the CDROM drive.
For example: `cd /mnt/cdrom` or `cd /media/cdrom`.
5. Run the script: `./install.sh`.
6. Follow the instructions in the Qtopia SDK installer.
7. Run VMWare Player and select `greenphone.vmx` from the SDK installation directory.

1.2 SDK Windows Installation

1. Download the latest version of VMWare Player from <http://www.vmware.com>.
2. Install VMWare Player in accordance with the instructions in [VMWare Player Documentation](#).
3. Insert the Qtopia SDK CDROM.
4. The autorun installer should automatically display.

If the installer does not display follow these steps:

- a) Click `My Computer`.
- b) Click the CDROM icon.
- c) Click `autorun.exe`.

2 Setup on X86 for the FIC Neo

The following sections use X86 as an example of how to compile and install the Qtopia example application.

2.1 Phonebounce Mode

Phonebounce allows the use of the phones modem from within Qtopia running in QVFb.

To configure desktop Qtopia to use the Neo as a modem:

1. Ensure the Neo is connected via USB.
2. In VMWare Player click on the icon labeled 'runqpe (phonebounce mode)'.
3. On the Neo, go into Settings and in Startup Flags select Local – Phone Bounce mode then reboot the Greenphone for the change to take effect.

To test that phonebounce is working correctly:

1. `ssh root@neo:12345`
once connected you can send AT commands directly to the modem.
2. By typing AT you should see OK displayed.
3. Type `exit` to close and exit the `ssh` session.

2.2 VMWare Player

1. Click the `runqpe` icon to start [phonesim](#) and Qtopia in the [QVFb](#) with the Neo skin.
2. Click the `Konsole` icon to open a command shell and enter the following commands:
 - `source /opt/Qtopia/SDK/scripts/devel-x86.sh`
 - `cd ~/projects/application`
 - `qtopiamake`
 - `make clean` ! if previous build exists
 - `make` ! `make loud` displays compiler output
 - `sdk -p` ! create the package file
 - `sdk -i` ! install the package to the device
 - On the device, select *Install* from the dialog that appears.
3. Launch the `Example` application in the Qtopia environment by selecting `Applications->Example`.

3 Flashing the Neo from the Development Environment

The Qtopia SDK requires the latest image to be loaded onto the device and there are two methods to do this:

1. via `updatedevice` script.
2. via low-level flash using the `flash-neo` or `dfu-util` utility.

3.1 Update via the UPDATEDEVICE script

Note: Before commencing this process it is recommended to backup all documents on the Greenphone as per the instructions in Section 9: *Transferring Documents To and From the* .

1. From the VMWare Player's **Device** menu, select **Trolltech Greenphone** then run one of the following commands in a command shell:
 - `updatedevice` to update the device with the default image from the SDK **or**
 - `updatedevice [image]` to update the device with the specified image.
2. The first time you connect to the Neo, you will be asked whether to continue connecting. Answer "yes".
3. The screen displays *Figure 1* to indicate that Qtopia is being flashed - the flash process takes a few minutes.
4. The screen will change to either *Figure 2* on success, or similar to *Figure 3* on failure. If flash is successful the phone will automatically power-off after several seconds of the pass display. **Note:** The phone will automatically restart after it has powered off.

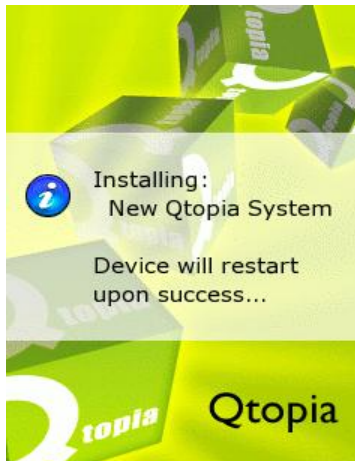


Figure 1: Flashing

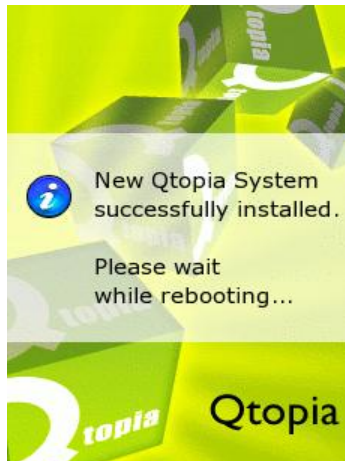


Figure 2: Pass



Figure 3: Fail

3.2 Low-level Flash using usbflash Utility

Use the following command and then follow the instructions in the console:

- `sudo usbflash -i /opt/Qtopia/extras/images/qtopia-neo-flash.jffs2`

Note: The Neo will automatically reboot once the flash process is complete.

4 Setup, Develop and Create Qtopia Packages

The process to setup, develop and create packages for the Neo is as follows:

In VMWare Player, open the console and enter:

1. `source /opt/Qtopia/SDK/scripts/devel-ficgta01.sh`
2. `cd ~/projects/application`
3. `qtopiamake`
4. `make clean` ! if previous build exists
5. `make` ! make loud to display compiler output
6. `sdk -p` ! create the package file
7. `sdk -i` ! install the package to the device
8. On the device, select *Install* from the dialog that appears.
9. On the device, launch the Example application by selecting Applications->Example.

For further `sdk` command line options refer to Section 10: *sdk Command Utility*.

To uninstall the package, use the Software Packages settings application on the Neo.

5 Remote Debugging for Neo

To remotely debug the example Qtopia application follow these steps:

1. `source /opt/Qtopia/SDK/scripts/devel-ficgta01.sh`
2. `cd ~/project/application`
3. `qtopiamake -debug`
4. `make clean` ! if previous build exists
5. `make` ! make loud to display compiler output
6. `sdk -p` ! create the package file
7. `sdk -i` ! install the package to the device
8. On the device, select *Install* from the dialog that appears.
9. `sdk -r -debug` ! run the application in gdbserver on the device
10. In the SDK, click the `arm-linux` debugger icon.
11. Open the example application by selecting File->Executable from the kdbg menu, and navigate to the application executable, in this case `/home/user/projects/applications/example`.
12. Open the source code of the example application by selecting File->Open Source from the kdbg menu, and select the file `example.cpp`.

13. In the source listing, set a breakpoint by clicking on a line of code and then right-clicking and selecting `Set/Clear Breakpoint` from the popup menu. The first line of the `Example` class constructor is a good place to set a breakpoint.
14. Execute the program by selecting `Execution->Run` from the `kdbg` menu. The application will run to the breakpoint and then stop.
15. Further debugging activities can then be performed. For example, to resume running the example application, choose `Execution->Run` from the `kdbg` menu.

To view debug information click the `Neo Log` icon.

Note: To rerun the example application it is necessary to repeat steps 9-14 above to re-initialize the remote connection to the Neo.

6 Neo Console Access

For development and debugging it is sometimes useful to have direct access to a command prompt on the Neo device.

To get a command prompt on the Neo:

1. Login to the device:

```
sdk -console
```

Once you have a command prompt, you will be able to run any of the standard Linux commands that are installed on the device.

7 Installing Qtopia Source Code into the SDK

Follow the instructions for Building a Neo Flash Image below.

8 Building a Neo Flash Image

To build a Neo flash image follow these steps:

1. If you have a commercial source package you will have received a Qt license file. This file must be copied to `/home/user/.qt-license` in the SDK. This step is not required if you are using an Open Source package.
2. Download source packages to `/devel/sdk-source/package`.
 - a. `qtopia-phone-<version>.tar.gz`
 - b. `qtopia-helix-commercial-src-<version>.tar.gz` (optional)
3. Extract Qtopia source package:

```
cd /devel/sdk-source/package
```

```
tar -xzf qtopia-phone-<version>.tar.gz
cd qtopia-phone-<version>
export QTOPIA_DEPOT_PATH=$PWD
```

4. Extract Helix source package (optional):

```
cd $QTOPIA_DEPOT_PATH
tar -xzf ../qtopia-helix-commercial-src-<version>.tar.gz
```

5. Build Qtopia:

```
export QPEDIR=/devel/build/qtopia-<version>/
mkdir -p $QPEDIR
cd $QPEDIR
$QTOPIA_DEPOT_PATH/configure -device ficgta01
make
make install
```

6. Create the flash and updatedevice image:

```
$QTOPIA_DEPOT_PATH/devices/ficgta01/scripts/make-flash.sh
```

7. Install the new Qtopia on the Neo:

```
updatedevice qtopia-ficgta01-update.tar.gz
```

9 Transferring Documents To and From the Neo

To transfer documents to or from the Greenphone follow these steps:

1. Ensure the device is connected via USB cable.
2. Ensure the device is selected in VMWare Player's **Devices** menu.
3. Open a Konsole on the SDK and enter the command: `sdk -net .`
4. Open the Konqueror web browser.
5. Select the bookmark `fish://root@neo/`.
6. Navigate the browser to `Documents`.
7. Files can now be copied to and from the Desktop or another web browser.
8. Open the `Devtools` application list on the phone.
9. Run the `Rescan Documents` application.

Files are now available in the Documents application.

10 sdk Command Utility

sdk is a command line utility used to streamline the build process and simplify Neo related tasks such as network setup. The following describes the command-line format and available options:

FORMAT: `./sdk <options>`

Options:

- `-h` display command help
- `-q` quiet mode
- `-c` clean
- `-C <opt>` reconfig <x86> or <ficgta01>, default ficgta01
- `-b` make
- `-p` make package
- `-i` install package to device
- `-r` run application on device
- `-all` reconfig, make, make package, install, run
- `-debug` force configure for debug build, override default
- `-release` force configure for release build, override default

- `-options "<opts>"` pass extra options to build system
eg. `./gph -C x86 -all -options "CONFIG+=debug"`

- `-net` bring up network to neo
- `-console` console on neo
- `-ipk <file>` install ipk to neo, next arg it ipk filename
- `-rescan` Tell Qtopia running in QVfb that new application exists