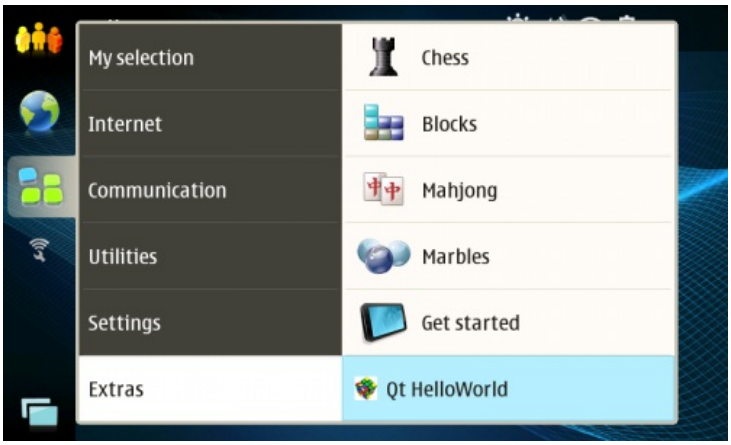
Creating Debian packages for Maemo 4 Qt applications and showing in the application menu

Archived: This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

The article is believed to be still valid for the original topic scope.

This article shows how to build a **.deb** package for Maemo 4 Qt applications, and how to deploy the Qt application so that it is visible on the device menu.

The proposed strategy does not use dh_make to create the basic debian package structure, as described in Qt for Maemo Developers Guide.



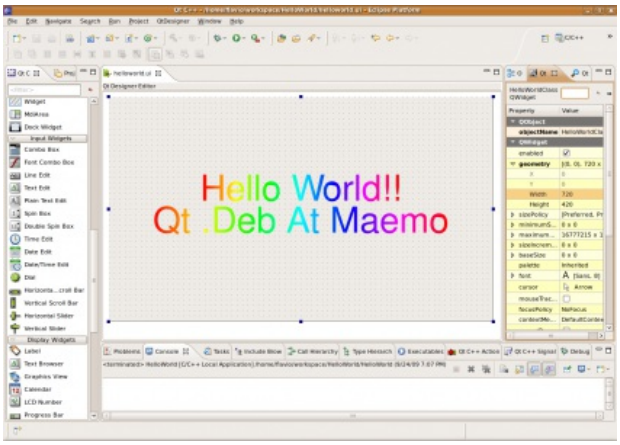
.Deb Packages In Brief

Deb is the extension of the Debian software package format and the most often used name for such binary packages. Like the **Deb** part of the term Debian. **Debian** packages are used in distributions based on **Debian**, such as **Ubuntu** and others.

Prerequisites

You need to have **libqt4-dev** ,**ScratchBox** and **Maemo SDK** installed on your Computer and **libqt4-dev** installed on your **Scratchbox ARM** target and your **Maemo Device** . For more information on **ScratchBox** Installation click here and for **Qt** on **Maemo** click here.

Creating the Application



You need to create a simple **Qt HelloWorld** Project on computer. I created a project with two labels how is showed below. Remember of to set the size of QWidget to 720 of Width and 420 of Height because the resolution of Maemo Device.

Copying the Application to Scratchbox

First, copy the sources files of application to a folder at **Scratchbox**. In general the address is **/scratchbox/users/username/home/username/**. I do this:

(C) Copyright Nokia 2013. All rights reserved.

```
flavio@flavio-desktop:~/workspace$ cp -r HelloWorld/ /scratchbox/users/flavio/home/flavio/
```

Compiling the Application at Scratchbox

At Start, you need **login** at **Scratchbox**:

```
~$ /scratchbox/login
```

If necessary select the **ARM** target using **sb-conf se DIABLO_ARMEL** . To compile do this :

```
[sbox-DIABLO_ARMEL: ~] > cd HelloWorld
[sbox-DIABLO_ARMEL: ~/HelloWorld] > qmake
[sbox-DIABLO_ARMEL: ~/HelloWorld] > make
```

Creating the Tree of Debian Package

After that, you need to create the following folders:

```
[sbox-DIABLO_ARMEL: ~/HelloWorld] cd ..
[sbox-DIABLO_ARMEL: ~] > mkdir HelloWorldApplication
[sbox-DIABLO_ARMEL: ~] > mkdir HelloWorldApplication/DEBIAN
```

You'll need to create a control file for your package using a simple text edit. In this post I use the **Vi**.

```
[sbox-DIABLO_ARMEL: ~] > vi HelloWorldApplication/DEBIAN/control
```

In this step we'll look at what each field in the control file means and what kind of information you need to provide. The syntax for the control fields is very simple: the name of the field, followed by a colon, and then the body or value of the field.

```
Package: HelloWorld
Priority: optional
Version: 1.0
Architecture: all
Maintainer: Flavio Fabricio <flaviofabricio@gmail.com>
Depends: libqt4-dev
Description: This is a HelloWorld Application
```

Returns to **HelloWorld** Application directory and create the folders below:

```
[sbox-DIABLO_ARMEL: ~] > cd HelloWorldApplication/
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication] > mkdir usr
```

Creating the /usr subdirectories:

```
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication] > cd usr/
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication/usr] > mkdir bin
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication/usr] > mkdir lib
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication/usr] > mkdir share
```

Creating the /share subdirectories:

```
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication/usr] > cd share/
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication/usr/share] > mkdir applications
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication/usr/share] > mkdir pixmaps
```

Creating the **hildon** subdirectory that is the folder where is the **.desktop** files :

```
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication/usr/share] > cd applications
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication/usr/share/applications] > mkdir hildon
```

Creating the file thats put the Application on Maemo Menu

You'll need to create too, a .desktop file that the **Maemo** uses to put application at Menu.

```
[sbox-DIABLO_ARMEL: cd /HelloWorldApplication/usr/share/applications/hildon
[sbox-DIABLO_ARMEL: ~/HelloWorldApplication/usr/share/applications/hildon]>vi helloworld.desktop
```

Here we'll to add the address of executable file, image icon file, etc...

```
[Desktop Entry]
Categories=Qt;
Encoding=UTF-8
Version=0.1
Type=Application
Name=HelloWorld
Exec=/usr/bin/HelloWorld
Icon=helloworld
```


(C) Copyright Nokia 2013. All rights reserved.

Copying the Application Compiled to Deb Tree

```
[sbox-DIABLO_ARMEL: ~] > cd HelloWorld
[sbox-DIABLO_ARMEL: ~/HelloWorld] > ls
HelloWorld helloworld.cpp main.cpp helloworld.png
HelloWorld.pro helloworld.h helloworld.ui ui_helloworld.h
```

In this step, you will to copy the executable file to /usr/bin folder. This executable is called by helloworld.desktop at line **Exec=/usr/bin/HelloWorld** on the file described above.

```
[sbox-DIABLO_ARMEL: ~/HelloWorld] > cp helloworld ../HelloWorldApplication/usr/bin/
```

Now, you will to copy the sources files to /usr/lib folder.This is recommended, but not required.

```
[sbox-DIABLOE_ARMEL: ~/HelloWorld] > cp HelloWorld.pro ../HelloWorldApplication/usr/lib/
[sbox-DIABLO_ARMEL: ~/HelloWorld] > cp helloworld.cpp ../HelloWorldApplication/usr/lib/
[sbox-DIABLO_ARMEL: ~/HelloWorld] > cp helloworld.h ../HelloWorldApplication/usr/lib/
[sbox-DIABLO_ARMEL: ~/HelloWorld] > cp helloworld.ui ../HelloWorldApplication/usr/lib/
[sbox-DIABLO_ARMEL: ~/HelloWorld] > cp main.cpp ../HelloWorldApplication/usr/lib/
```

You will need a picture to use like Icon. You can search some image at the Google Images.In this case I renamed the picture to **helloworld.png**.This image is called by **helloworld.desktop** at line **Icon=helloworld** on the file described above and save to **/usr/share/pixmaps/** folder.

```
[sbox-DIABLO_ARMEL: ~/HelloWorld] > cp helloworld.png ../HelloWorldApplication/usr/share/pixmaps/
```

Creating the Deb Package

```
[sbox-DIABLO_ARMEL: ~/HelloWorld] > cd ..
[sbox-DIABLO_ARMEL: ~] > dpkg-deb -b HelloWorldApplication
```

Sending the .Deb Package to Maemo Device

I used the **Scp** command with the **IP** address of device. The **IP** address of device is searched at **Settings->Connection Manager** , after that, click at Connection Manager at **left-Up-Corner->internet connection->IP address**.

```
[sbox-DIABLO_ARMEL: ~/HelloWorld] > scp HelloWorldApplication.deb root@10.0.0.13:/home/
```

Installing the Deb Package at Maemo Device

Open a new terminal and do this:

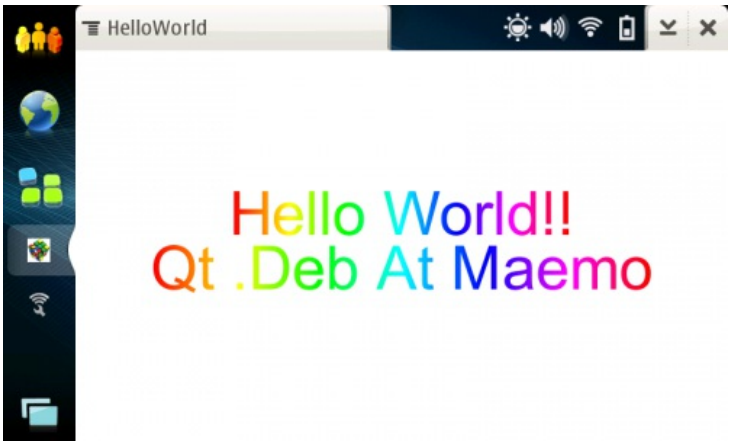
```
ssh root@10.0.0.13
```

enter with the **password** of **Maemo** Device

```
root@10.0.0.13's password:
Nokia-N810-43-7:~# cd ..
Nokia-N810-43-7:/# cd /home/
Nokia-N810-43-7:/home# dpkg -i HelloWorldApplication.deb
```

Now, the application is installed at **Maemo** Device.

Snapshot of HelloWorld Application



(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.