Exercises Lecture MeeGo 1 – An Introduction to Qt and Meego

Aim: This exercise will help you setup a working Qt development environment for MeeGo.

Duration: 1h (excluding download times)

Introduction

In this exercise you will set up your environment for developing Qt applications for MeeGo. The development environment will be based on a VirtualBox virtual machine and Qt Creator as the IDE for developing and packaging MeeGo software. It is possible to use QEmu instead of VirtualBox, but this approach is not covered by the instructions provided here.

The exercise is prepared for release 1.2 of MeeGo. If you are using another release, details may differ. However, the steps to carry out will remain the same.

The setup consists of three crucial pieces. First, you will need to setup VirtualBox. Then you will setup an embedded MeeGo system for deploying and testing MeeGo applications. Finally, you will prepare the environment for writing and compiling source code.

> **Important!** This exercise requires downloading large files from the web. If you are performing the exercise under supervision of a tutor or a teacher, check whether you are supposed to download the files yourself or whether the files already have been downloaded for you and are available locally. If you are to download the files yourself, consider downloading the files in step 3 before starting to work on the exercise, as it will take a significant time to download them.

Install VirtualBox

Before installing VirtualBox make sure your computer satisfies the minimum requirements. For this exercise, you will need a recent Intel/AMD based machine with at least one gigabyte of RAM and about ten gigabytes of mass storage available. For performance reasons it is highly advised that your machine is equipped with hardware-assisted virtualization, e.g. Intel VT-x or AMD-V. You may have to enable these features in the BIOS of your machine.

Linux

If you are running Linux, you possibly already have the OpenSource Edition of VirtualBox installed, or that you can install it using a standard package manager for your distribution. If you are running ubuntu, the relevant package is called *VirtualBox OSE*.

It is preferred that you install version 4.0 (or later) of VirtualBox. If you don't have it in your repositories, proceed to http://www.virtualbox.org/wiki/Linux_Downloads and either download the appropriate deb or rpm file, or add a repository matching your distribution release and use the package manager. Remember that you may require administration privileges to install software in a system directory.

If you don't have superuser privileges, or there is no dedicated package for your distribution, you can download the self-extracting archive labeled “All distributions”. Before executing the downloaded file, you must enable the execution flag. You do that using the following command, executed from the command prompt:

```
chmod a+x VirtualBox-version-arch.run
```

Substitute version and arch with the data for the file that you downloaded.

Windows/Mac OS X/Solaris

Proceed to http://www.virtualbox.org/wiki/Downloads and download the latest release of VirtualBox for your system. Install the downloaded application using the installer provided.

Creating a MeeGo virtual machine

Start by downloading the MeeGo release for Tablets from https://www.meego.com/downloads. Download the latest live image.

Start VirtualBox and choose to create a new virtual machine. In the wizard, provide the following information.

- Name : “MeeGo Tablet”
- Guest operating system : Linux Fedora
- Amount of RAM : at least 512MB
- Create a new virtual harddisk for your MeeGo installation
  - Size: at least 8GB
  - Accept the defaults from the harddisk wizard

After both wizards have finished, select your newly created machine, but do not start it. Instead, choose to visit the settings. Apply the following settings:

- In the system group:
  - Enable PAE/NX
  - Enable all options in the acceleration tab
- In the display group:
  - Enable 3D acceleration
- In the storage settings group:
  - Select the empty CD entry listed under the IDE controller and choose to setup a new virtual CD/DVD drive by choosing the virtual disk file.
  - In the file dialog select your downloaded MeeGo image. In order to be able to use the MeeGo image, you might have to change the file's extension to `iso`.
- In the network group:
  - Configure the active network card to be connected as “host-only”. This limits the virtual machine to only reach your machine. If you need access to an external network, you can choose to connect the network card via NAT or Bridged mode. Please consult the VirtualBox documentation to learn the different settings available and what is applicable in your network setting.

Installing MeeGo

When the virtual machine has been configured, it is time to install MeeGo on it.

Start by booting the virtual machine you just created. From the boot menu, select “Installation Only”. After a while the installation wizard is launched. Complete the wizard, filling in your data as you go along.

If you get an error message about an uninitialized disk, simply choose to reinitialize the drive.

Accept the defaults and confirm writing the changes to disk. This will start the installation. It will

take a few minutes to complete. When the system has been installed, click the “Close” button. The machine reboots and the boot menu is shown one again. Then stop the machine and dismount the MeeGo image through VirtualBox.

Now boot the virtual machine again without the MeeGo in the CD drive. You will see a welcome screen and a wizard. Complete the wizard, choosing user name, password, etc.

After the wizard, the machine will reboot again (you may have to reboot it manually if all you see is a black screen).

Now the machine will start in the screen shown below.

Fixing issues with the graphical user interface not starting

If you experience problems booting, then reboot it and immediately press the Esc key to get into boot menu. There, press the Tab key and to modify the boot command. Remove the “quiet” keyword and append an “s” at the end. The resulting command should look something like the line shown below. This will boot into a shell instead of into the graphical user interface.

```
vmlinuz-2.6.35.3-10.3-tablet ro root=/dev/sda2 vga=current s
```

From the command line, issue the following command.

```
chmod +s /usr/bin/Xorg
```

This will allow the window manager to boot properly. Now reboot the machine and you should reach the screen above.

Enabling the mouse pointer

Since tablets are touch-based, the mouse cursor is invisible by default. This makes it hard to operate the machine using a standard mouse. To show the cursor, boot into the shell as described earlier (altering the boot command in the boot menu). You will need to edit the /etc/sysconfig/uxlaunch file. It is easiest to do it using nano. Issue the following command:

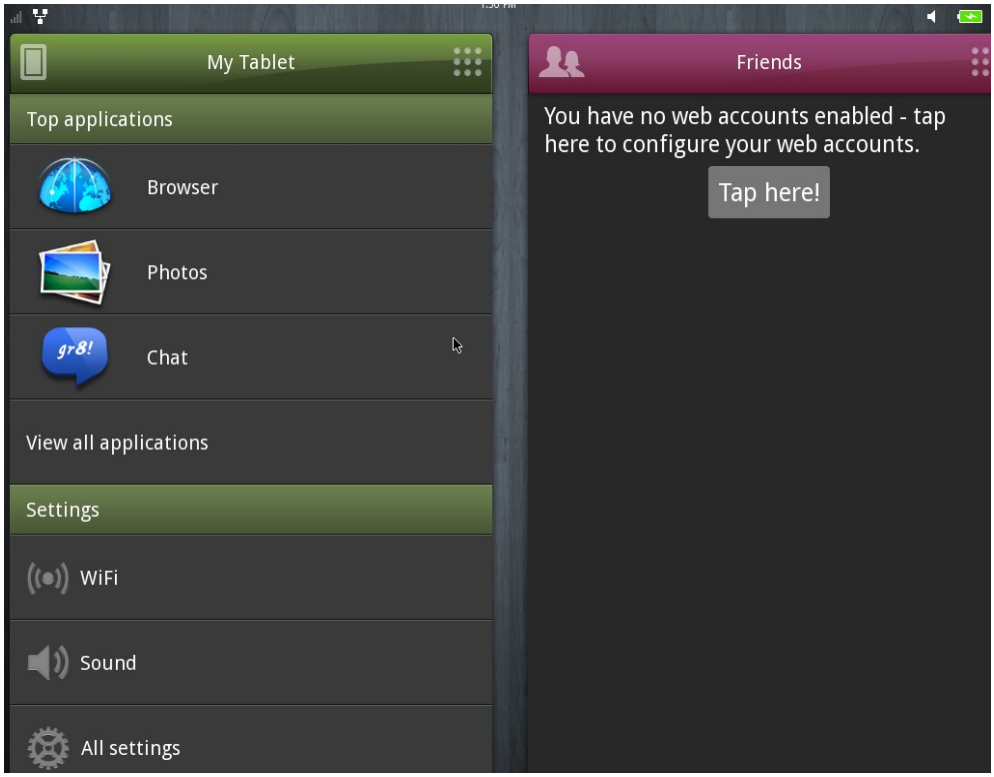
```
vim /etc/sysconfig/uxlaunch
```

nano is a console-based text editor and all its operations are performed using keyboard. Use cursor keys to find the line containing the following entry:

```
xopts=-nocursor
```

You need to comment it out by placing a hash (#) character in front of the entry. Save your modified file using Ctrl+O, then exit nano by pressing Ctrl+X.

After saving the modifications you can reboot the virtual machine and let it boot to the user interface. You will notice that now the mouse cursor is shown. After unlocking the display you should see a screen similar to the one below.



Installing MADDE

There is one thing left to do before the virtual target installation is ready to use. To enable proper deployment of applications to the virtual target, a special tool chain called MADDE has to be installed. Launch your virtual machine, let it boot to the UI, access the applications panel and start the terminal. From there, issue the following command:

```
su
```

You will be asked for the root password that you specified earlier in the installation. To install MADDE, run the following command.

```
zypper install mad-developer
```

Finally install and start SSH server to enable remote logins.

```
zypper install ssh
/etc/init.d/ssh start
```

MeeGo SDK

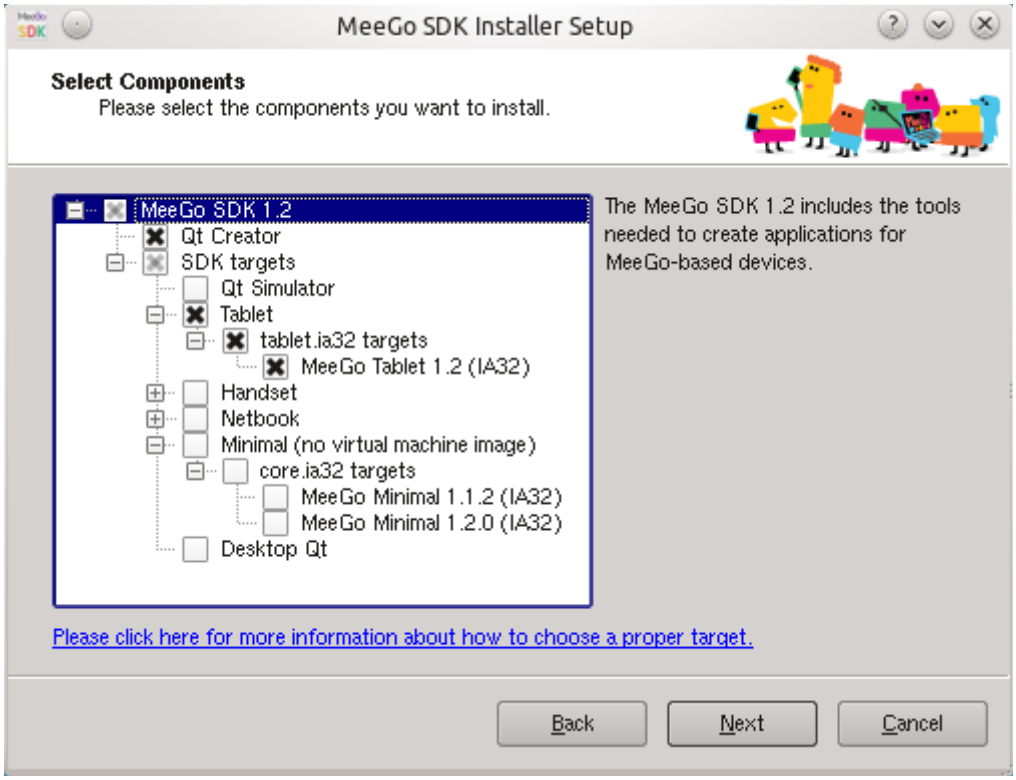
Currently, the MeeGo SDK can only be installed in a 32 bit environment. If your machine is 64 bit, you will need to create another Virtual Machine to host the SDK.

There are two ways to install the SDK – one relies on an installer and the other is a manual process. If the installer approach fails or is not available for your system, the manual approach can be used. If you can use the installer, that is the recommended procedure.

Using MeeGo SDK Installer

Go to https://meego.com/downloads and download the MeeGo SDK installer. If you're running Linux, set the execution bit on the downloaded binary (i.e. by using chmod) and then start the installer.

When you are asked for what to install, install Qt Creator and the tablet environment. It is also recommended that you check Desktop Qt and the Qt Simulator.



The installer will take a considerable amount of time as it will be downloading packages from remote repositories.

Manual Installation

Manual installation is performed using command-line. In these instructions, we demonstrate how to perform the installation from a machine installed with Debian or Ubuntu.

The installation consists of three steps. First, MADDE and the SDK are installed. The the build target is installed.

Installing MADDE and the SDK

Browse to http://download.meego.com/live/Tools:/SDK:/Host:/. There follow the link to the version of MeeGo you want to install (i.e. '1.2') and then click the link that best matches your distribution release. Copy the address from the browser's address bar. Use the address you have just obtained and add the following line to /etc/apt/sources.list or equivalent, replacing the address with the one you have copied from the browser.

```
deb http://download.meego.com/live/Tools:/SDK:/Host:/1.2/Ubuntu_10.10/ /
```

The next step is to add the MeeGo signing key to your keychain. Run the following commands to do so.

```
sudo gpg --keyserver pgpkeys.mit.edu --recv 0BC7BEC479FC1F8A
sudo gpg --export --armor 0BC7BEC479FC1F8A | sudo apt-key add -
```

Now, make apt aware of the changes:

```
sudo apt-get update
```

You can then install the MeeGo SDK by issuing:

```
sudo apt-get install meego-sdk-ia32
```

This will install the SDK for Intel processors which is enough for the exercise. If you want to be able to deploy to an ARM device you'd also need meego-sdk-armv7l.

Installing targets

To install a target, use the mad-admin tool.

```
sudo mad-admin create -f meego-tablet-ia32-<version>
```

After the installation is complete you can verify the current state of your MeeGo application development environment by issuing the following command.

```
sudo mad-admin list
```

The MeeGo target you have chosen should be marked as “installed”.

Configuring build target

Having installed MADDE and the SDK, Qt Creator needs to be made aware of the build target. Start this by launching Qt Creator. Open the settings dialog and go to Qt versions configuration panel.

Choose to add a new Qt version. Call it meego-tablet-ia32. To find the proper location of qmake for the target using the following command.

```
mad query install-dir
```

The qmake binary resides in the targets/<targetname>/bin/ subdirectory. Enter the path in the settings dialog in Qt Creator. Optionally, click the Rebuild button to build the debugging helper for the target. You should now be able to build for MeeGo by choosing this Qt version for your project.

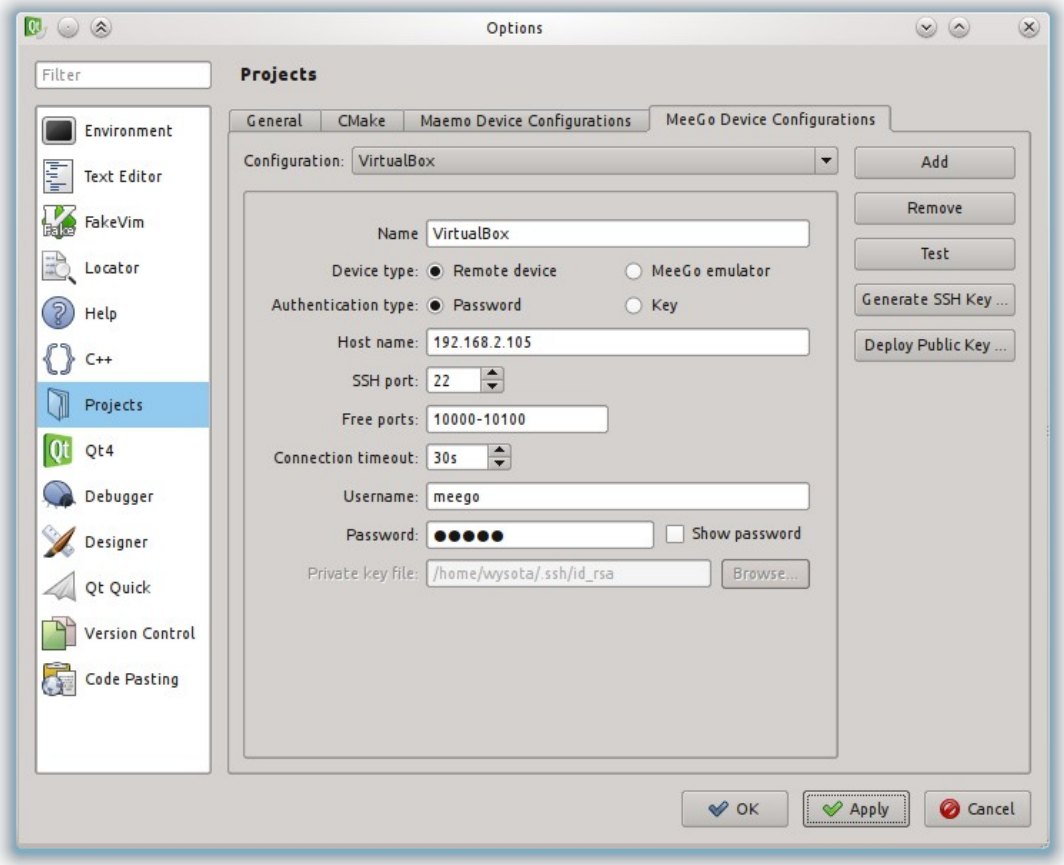
Configuring the Target

Having setup a development environment, you need to target the virtual machine installed with MeeGo. Start by launching your VirtualBox MeeGo machine and make sure it has ssh enabled. You can test it by trying to connect to the machine using ssh or PuTTY.

Then launch Qt Creator on your development machine and access the “Projects” group in the settings. Switch to the “MeeGo Device Configurations” tab or, if it is missing from your Creator installation, to the “Maemo Device Configurations” tab.

Create a new configuration according to the picture below. Replace the IP address in the 'Host Name' field with the IP address of your MeeGo virtual machine. Then click “Test” to test the connection. If the connection fails, do the following:

- check the IP address of the virtual machine,
- verify ssh is installed and running on the virtual machine,
- check that you can connect to the machine manually using credentials provided in the configuration,
- ensure that the 'mad-developer' package is installed on the virtual machine.



If you get a positive result of the test, it means your MeeGo development environment is now correctly installed and working. Congratulations!