

# How to obtain and build the QNX kernel source

## ***Purpose***

This document describes, in detail, the process of obtaining, and then building the QNX kernel sources using the Momentics Tools hosted on the Windows XP or Vista operating system.

## ***Pre-requisites***

The pre-requisites for using this document as a guide to obtaining and building the source, are:

- A working knowledge of the ‘C’ programming language
- A properly installed and configured Windows XP or Vista operating system with Sun’s Java5 JRE.
- A working internet connection from the host

Note: *If the Sun Java5 JRE is not installed, it (and instructions for installation) can be obtained from [http://java.sun.com/javase/downloads/index\\_jdk5.jsp](http://java.sun.com/javase/downloads/index_jdk5.jsp) (Java Runtime Environment (JRE) 5.0 Update 12).*

## ***How long will this take?***

Depending on level of pre-configuration, the internet connection speed, and machine speed (compilation) this process will typically take anywhere between 4 and 12 hours.

## ***What will I have when finished?***

After completing all the steps in this process, you will have a completely functional installation of the X86 (host and target) Momentics Integrated Development Environment (IDE), as well as the complete ‘C’ library and kernel sources (as well as various other libraries and system services).

After registering, installing, connecting, download and building the source, your IDE will remain connected to the “live” subversion source server which will allow you to get diffs between your environment and the tip of the “head” (trunk) development branch at QNX Software Systems. While you will not have commit privileges, you can request commit rights from:

[http://community.qnx.com/sf/sfmain/do/myProjects?selectedTab=all&&\\_pagenum=2](http://community.qnx.com/sf/sfmain/do/myProjects?selectedTab=all&&_pagenum=2)  
*(don’t click this link yet, as you’ll need to have registered before you will be granted access to this page).*

## ***Terminology***

To aid in understanding of this document, the following terms are used:

***Community Member:*** This is you, the reader of this document. After registering, you are a member of the Foundry27 community, which is the organization that maintains the publicly available QNX kernel source.

<b>Community:</b>	The group of individuals who participate in the development of the QNX kernel (this does not necessarily imply “committer”, many individuals participate by testing, documenting, and by offering suggestions.
<b>Foundry27:</b>	The “connection point” for the community. Foundry27 is the set of collaboration technologies (e.g. subversion server, web sites, forums and wikis) that tie together the community.
<b>Subversion:</b>	No, this is not a series of covert actions designed to disrupt or topple a government :-). Subversion is the source control system that the community uses to track development activities.
<b>Forum:</b>	Discussion group accessed through the Foundry27 site.
<b>Host:</b>	Your computer. AKA: the machine that you will use to build the kernel.
<b>Target:</b>	The computer on which the QNX kernel (the product of the build on the host) will execute.

## ***Ok, I'm ready now what?***

Great! The entire process spans 5 phases:

1. Registering
2. Installing
3. Connecting
4. Downloading
5. Building

So let's get started...

### ***Phase 1: Registering***

Before the source can be accessed, the prospective community member must register as a participant in the community. This section documents the process by which a prospective community member registers.

Fortunately this process is very easy. There only two steps in this process.

To register as a community member you need to access the login URL at:

<https://www.qnx.com/account/login.html>

You should see the following screen...

The screenshot shows the myQNX member login page. At the top, there is a navigation bar with the QNX logo and the text "QNX SOFTWARE SYSTEMS". Below the logo, it says "Middleware, development tools, realtime operating system software and services for superior embedded design". There are links for "Login", "Downloads", and "Search". A secondary navigation bar includes "Products", "Support + Services", "Markets", "Partners", "Community", "Company", and "Sales". The main content area has a header with "Home", "myQNX Account", and "Login". Below this is a large blue banner with the text "myQNX member login". Underneath the banner, there is a form titled "Please enter your email and password to login" with fields for "Email Address:" and "Password:", and buttons for "Login", "New Member?", and "Forgotten Password?". To the right of the form is a section titled "Why Join?" with a list of benefits: "Get exclusive downloads, code samples", "Participate in the community forums", "Submit enhancement requests", and "Access to a subversion account". Below this list is a link "Register Now." and the text "It's quick. It's easy. It's free."

Click on the "New Member?" button, and the screen should update to...

The screenshot shows the myQNX member registration page. The top navigation bar and secondary navigation bar are identical to the previous screenshot. The main content area has the same header and banner. Below the banner, the "Please enter your email and password to login" form is still present. To the right of the form is the "Why Join?" section. Below the "Why Join?" section is a new section titled "New to myQNX?" with three input fields labeled "Email:", "Password:", and "Verify:".

In the section titled "New to myQNX?" enter your email address (must be a valid email address) and choose a password, verify it, and then press enter. The screen will now update to this...

The screenshot shows the QNX Software Systems website's 'myQNX member login' page. At the top, the QNX logo is on the left, and navigation links for 'Products', 'Support + Services', 'Markets', 'Partners', 'Community', 'Company', and 'Sales' are on the right. Below the navigation is a breadcrumb trail: 'Home > myQNX Account > Login'. The main content area features a blue header with the text 'myQNX member login'. Below this is the 'myQNX Account Profile Form' with the instruction: 'Please fill out this short form to create your account.' The form contains the following fields: 'First Name', 'Last Name', 'Company/Organization', 'Address', 'Suite or P.O. Box', 'City', 'Country' (a dropdown menu with '<Select>' selected), 'ZIP/Postal Code', 'Telephone' (with separate fields for area code and extension), and 'Web'.

Fill out this form, and click “submit”, then (in a few minutes) you will receive an email at the address provided that will provide a link to validate your registration. To validate your registration, you’ll need the confirmation number from the email. Once you have the confirmation number in hand, you can click on the link in the email and re-enter your user id (email address) and password. You’ll then be prompted (this time only) to enter the confirmation key. Enter the confirmation number and your password and you’re done!

**Welcome to the Foundry27 community!**

## ***Phase 2: Installing***

In order to lay the groundwork for building the kernel (and libraries) the QNX tools must be installed on your Windows XP or Vista host. This is fundamentally a three step process.

1. Download the Momentics ISO CD image
2. Obtain a license key for the tools
3. Install the tools from the ISO CD image

### **Download the Momentics ISO CD image**

The Momentics ISO image can be obtained from:

<http://www.qnx.com/products/getmomentics>

At this URL you will see the following screen...

QNX  
QNX SOFTWARE SYSTEMS

Middleware, development tools, realtime operating system software and services for superior embedded design

Downloads Search

Products | Support + Services | Markets | Partners | Community | Company | Sales

Home | Products | QNX Momentics Download | Download QNX Momentics

## Download QNX Momentics

Download the QNX Momentics development suite 6.3.2 including the QNX Neutrino RTOS, fully integrated IDE, BSPs, driver development kits, libraries, tools and more. Only two simple steps to get you started.

**Step 1 Pick your host**  
Select your development platform from the options below. [Click here](#) to find out what's included. In parallel, once you've initiated your download you can proceed to get your license - right away.

Lite Version - x86 target only (~268Mb)

DOWNLOAD Windows Lite

DOWNLOAD Linux Lite

DOWNLOAD QNX Neutrino Lite

All targets: ARM/Xscale, MIPS, PPC, SH4, x86 (~542Mb)

DOWNLOAD Windows Host

DOWNLOAD Linux Host

DOWNLOAD QNX Neutrino Host

Related links

- What's included?
- System requirements
- Instructions on burning an ISO
- BSPs and chipsets

Quick-start videos

- English
- German

Evaluation support

- Forums
- Contact our global sales team

In order to get you up and running as fast as possible, this document assumes that the “Windows Lite” version will be downloaded. Don’t let the “lite” moniker concern you, this package is fully functional, except that it does not have the libraries for non X86 targets. Since you can easily update to the “full” version, once you are familiar with QNX, the extra time this download would take is probably not worthwhile at this time.

So go ahead, and click the “DOWNLOAD Windows Lite” button. This will bring you to the “Windows Lite” download page, where you can click “Download now” to begin the download. While the download proceeds the next step can be done in parallel; so let’s move on...

### Obtain a license key for the tools

In order to use the QNX tools you need to self-identify your intended use for the tools. There are four different use categories under which the tools can be licensed. All but one use category provides a full perpetual use license. The categories are:

- Non-commercial
- Academic
- Partner
- Evaluation

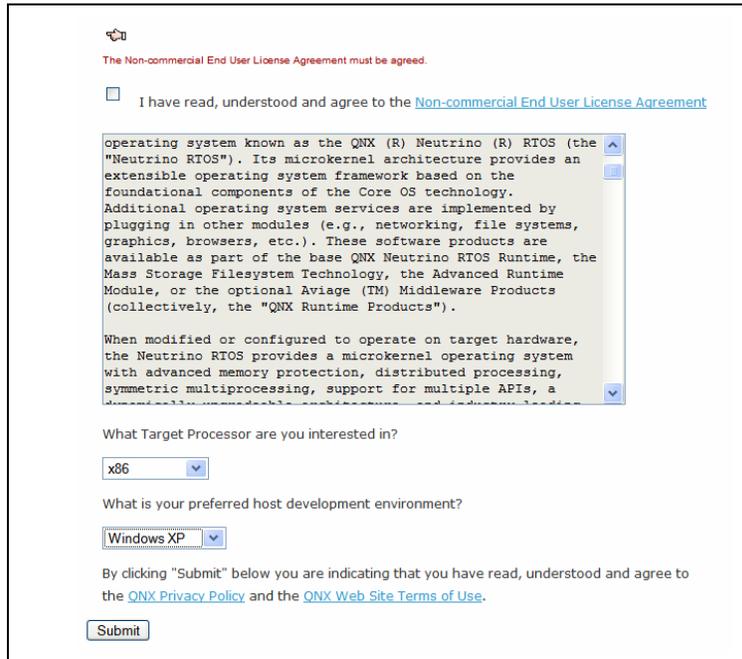
The first category (non-commercial) is intended for hobbyists who intend to use the tools for self-education on QNX, or to develop non-commercial software (i.e. home or open source project).

The second category (academic) is intended for credentialed educational institutions, who wish to use QNX as part of a defined curriculum.

The third category is intended for partners (e.g. silicon vendors, 3<sup>rd</sup> party s/w vendors, self-employed consultants).

The final category is the one which does not provide a perpetual license, and this is intended for prospective commercial customers of QNX. This license is good for 90 days, and is intended to allow commercial prospects to fully evaluate QNX technology.

Once you have determined which category applies to you, then return to the download page (<http://www.qnx.com/products/getmomentics>) and scroll to the section titled “Step 2 Get your license key”. Select the appropriate category, and you’ll be taken to a page that will re-confirm your personal details and to present you with the terms of the license. It will also ask what host/target you will be using; please select Windows XP/x86 (as shown).



The Non-commercial End User License Agreement must be agreed.

I have read, understood and agree to the [Non-commercial End User License Agreement](#)

operating system known as the QNX (R) Neutrino (R) RTOS (the "Neutrino RTOS"). Its microkernel architecture provides an extensible operating system framework based on the foundational components of the Core OS technology. Additional operating system services are implemented by plugging in other modules (e.g., networking, file systems, graphics, browsers, etc.). These software products are available as part of the base QNX Neutrino RTOS Runtime, the Mass Storage Filesystem Technology, the Advanced Runtime Module, or the optional Aviage (TM) Middleware Products (collectively, the "QNX Runtime Products").

When modified or configured to operate on target hardware, the Neutrino RTOS provides a microkernel operating system with advanced memory protection, distributed processing, symmetric multiprocessing, support for multiple APIs, a

What Target Processor are you interested in?  
x86

What is your preferred host development environment?  
Windows XP

By clicking "Submit" below you are indicating that you have read, understood and agree to the [QNX Privacy Policy](#) and the [QNX Web Site Terms of Use](#).

Once you have read, understand and agree with the license, click “Submit”. Within a few minutes you will receive a key in your email. If your email is accessible from the machine you are installing on (i.e. the host), then keep this email handy, as copying and pasting the key will save a lot of typing!

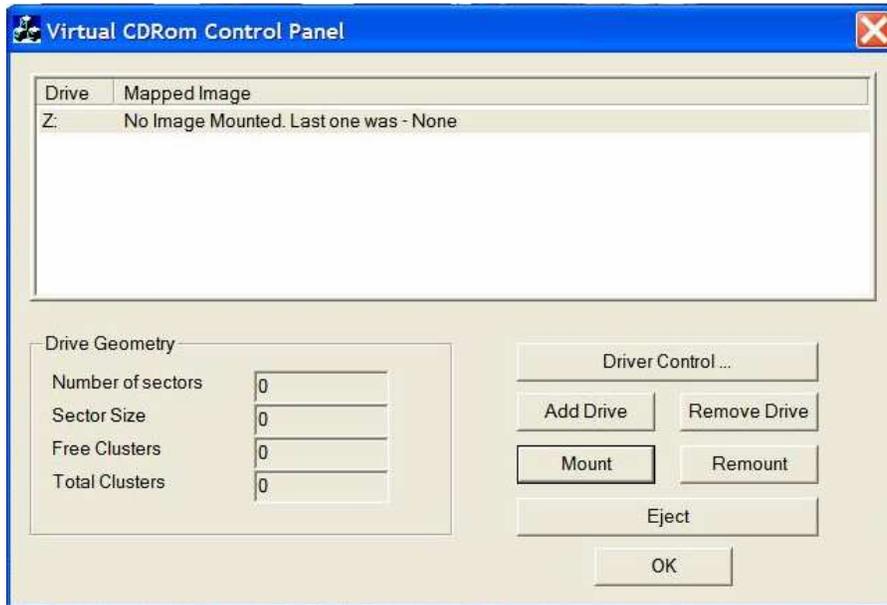
### **Install the tools from the ISO CD image**

Ok, you now have the key, however, the process of getting the key may have been shorter than the download process, so at this point you need to wait until the download completes. Once the download is complete, resume at the following paragraph.

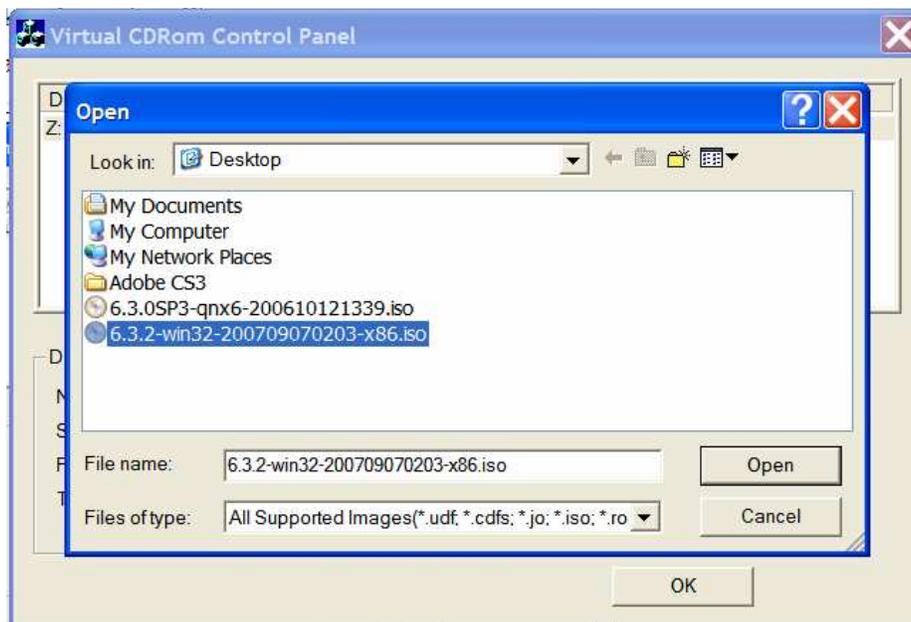
Ok, you now have the key and the download, so you are ready to install. Although the QNX website suggests burning the image to a CD, it is often faster and easier to use the Microsoft Virtual CDROM Utility (you should have received this with the email that contained this document). If you intend to burn the image to a CD then you can proceed directly to page 8, and continue with launching the installer from Windows File Explorer.

If you intend to use the Microsoft Virtual CDROM Utility, then execute the self-extracting zip file that was attached to the same email as this document. The file name should be winxpvirtualcdcontrolpanel\_21.exe.foo (the .foo extension allows the file to pass through anti-virus filters). The zip file contains three files, follow the readme.txt file to install the software (very quick and easy install).

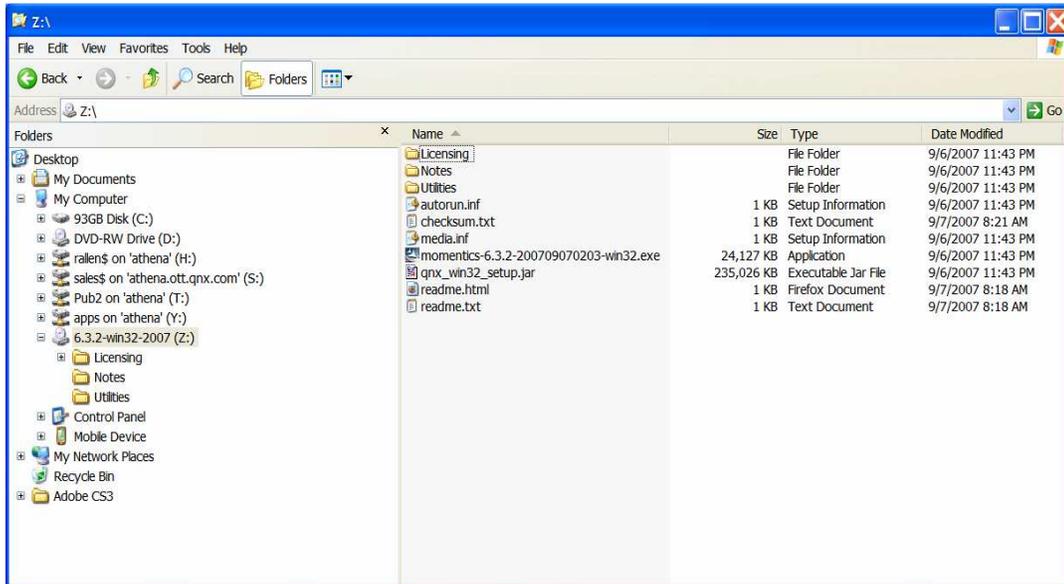
Once you have installed the Virtual CDROM utility, then execute the VcdControlTool.exe. You will see a display similar to the following...



If there is no drive already present, then click the “Add Drive” button, and one will be added. Then simply select a drive letter that currently does not have an image mounted, then click “Mount”. You will then see a pop up dialog similar to the following. Select the ISO image that you downloaded earlier (you may need to navigate to a different folder), then click on the file: 6.3.2-win32-????????????-x86.iso, and select “Open” (on the dialog that follows this, just click “Ok”).



Now click “OK” to dismiss the entire dialog, and there will be a new drive letter (accessible from Windows Explorer) that holds the QNX Momentics Installer.



Use Windows explorer to navigate to the root of the new drive letter, then locate the file: Momentics-6.3.2-????????-win32.exe, and double-click on this file. Now the QNX installer (which is not covered in this document, since it is self explanatory) will launch and prompt for a key. You can now access the email (containing the key) that you received from QNX, and copy the key (of the form XXXX-XXXX-XXXX-XXXX-XXXX) into the copy/paste buffer. You can then right-click on the first field of the key prompt and (from the pop-up menu) select paste (the prompt will be correctly filled with the key), and you can finish the install. At this point the installation phase is complete.

### **Phase 3: Connecting**

Let's recap. At this point you have registered as a community member, you have downloaded, obtained a license for, and installed the tools. Next you need to configure the IDE to connect to the community subversion server, so that the source code can be pulled into the IDE.

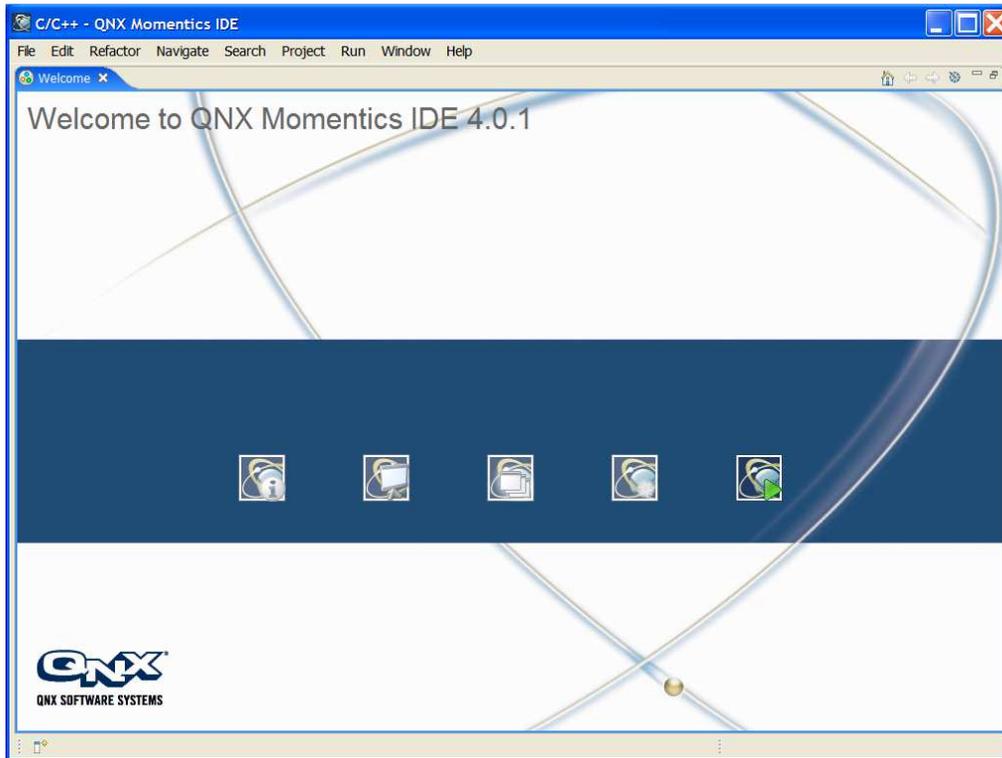
This phase consists of 2 steps:

1. Install the subversion plug-in
2. Connect to the public subversion repository

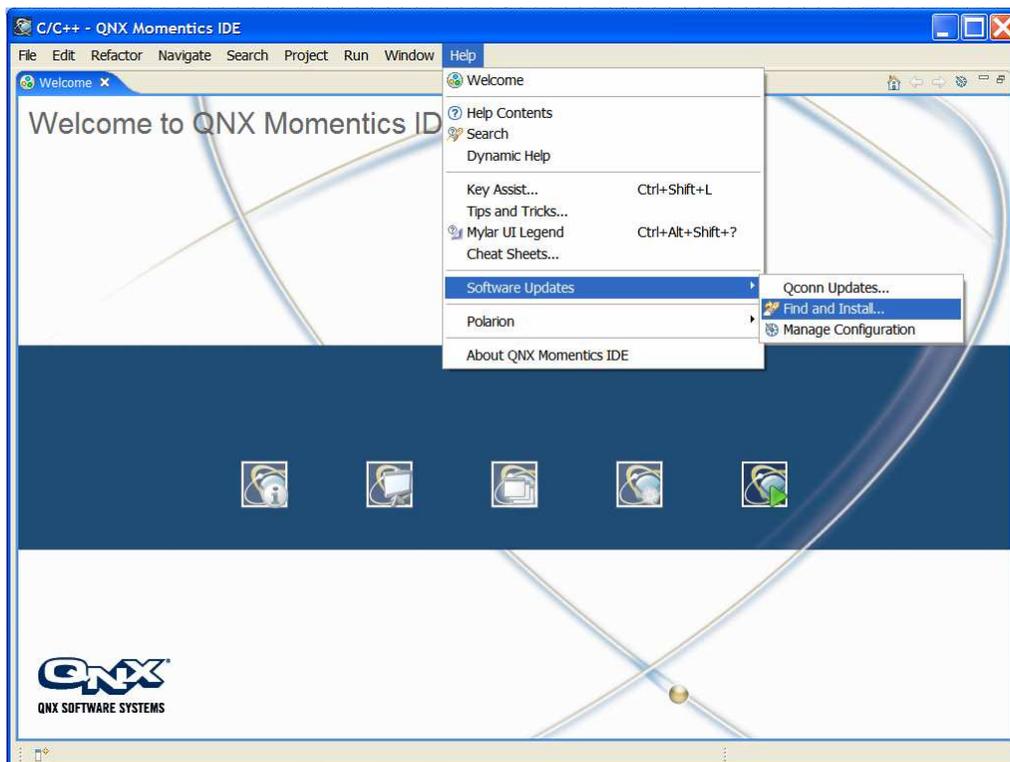
#### **Install the subversion plug-in**

To install the subversion plug-in (subversive), first you will need to launch your newly installed Momentics Integrated Development Environment (you should have an entry in the "Programs" menu called "QNX Momentics...").

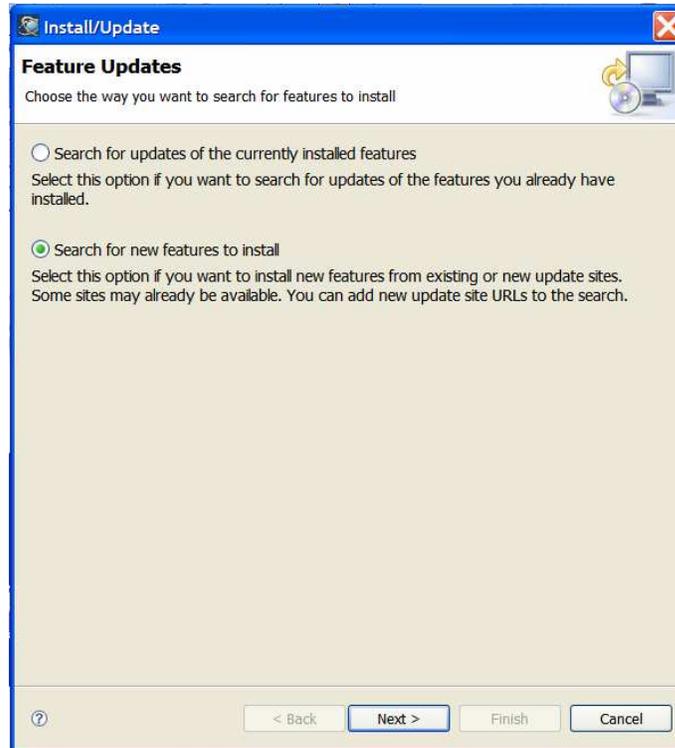
Once the IDE is running, you should see something like this...



From this point you need to open the “Find and Install” dialog, which is available from the “Help->Software Updates” menu.

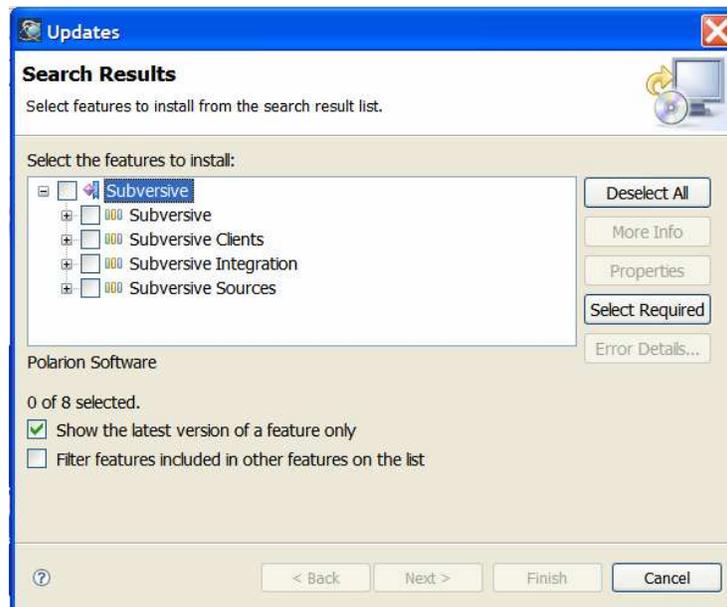


Selecting “Find and Install...” will bring up the following dialog...



Click on the “Search for new features to install” radio button, then select “Next”. On the following dialog click on “New Remote Site...”, and then (in the URL field) enter: <http://www.polarion.org/projects/subversive/download/1.1/update-site/>. For the name field anything is fine, but “Subversive” might be a good choice.

Once you have entered the name and url, then click “OK”. The entry will appear in the “Sites to include in search” pane. Check this entry only, then click “Next”. You should now see a dialog similar to the following...



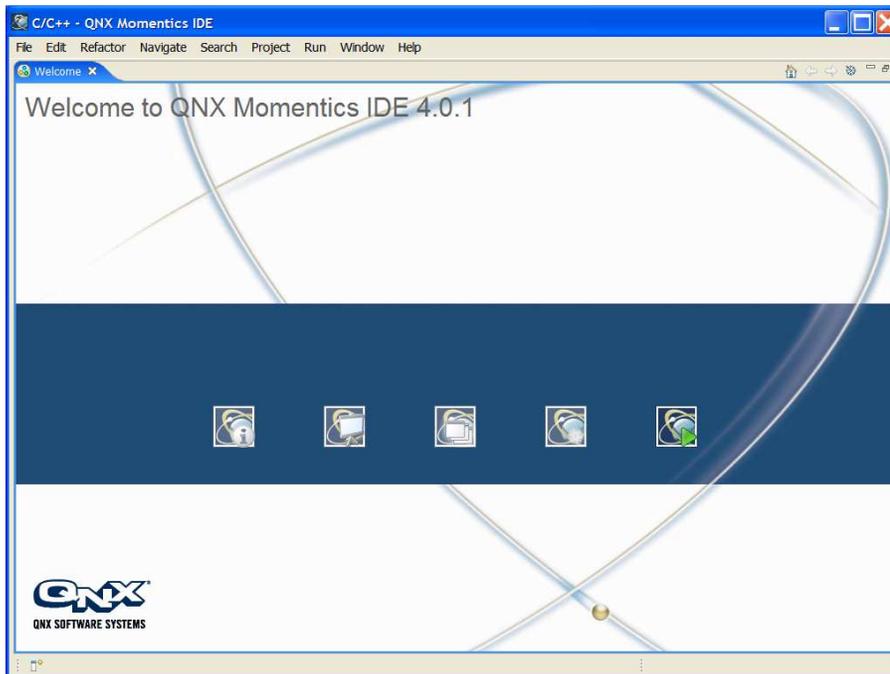
Check the highest level Subversive item (sub-items will check automatically), then uncheck “Subversive Integration”, and click “Next”. On the next dialog, accept the terms of the license, and then click “Finish”. After a few moments the subversive

software will be installed, and you will be prompted to re-start the workbench. Go ahead and say OK to re-starting the workbench.

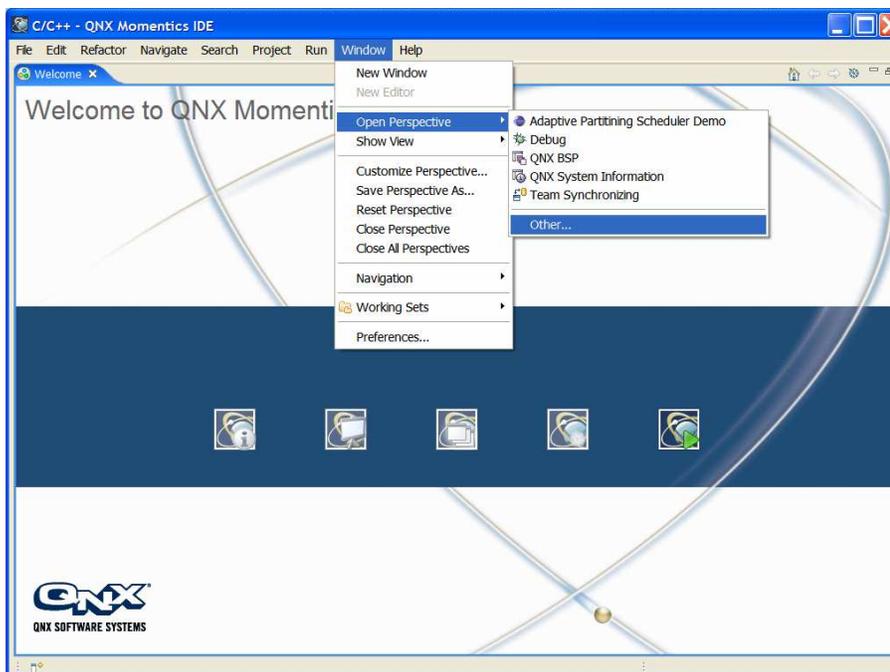
Once the workbench restarts you now have the ability to connect to the community subversion server.

## Connect to the public subversion repository

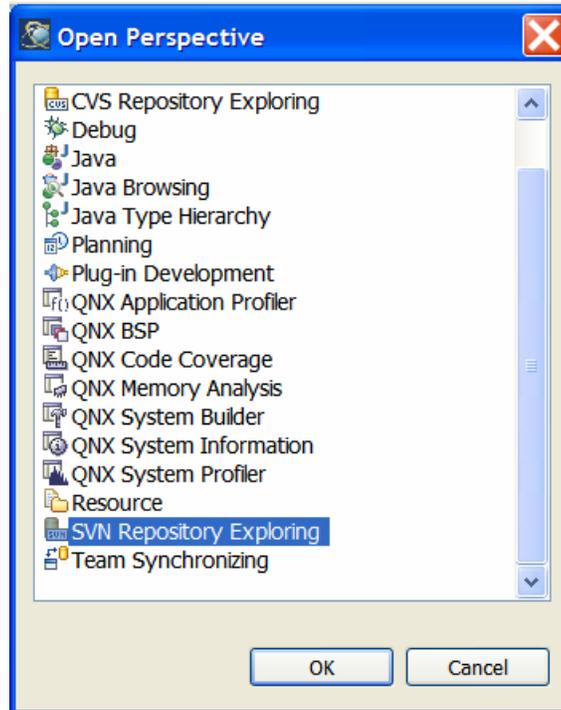
After restart you should have returned to a screen something like this...



From here select “Window->Open Perspective->Other”...

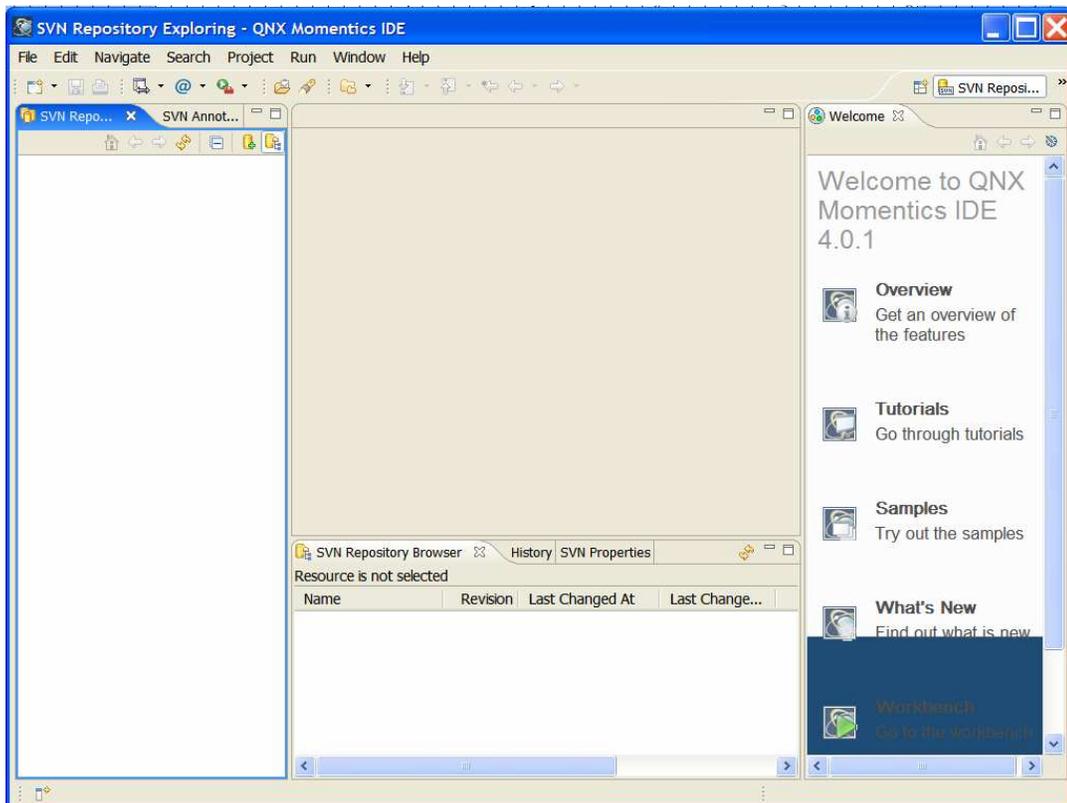


Clicking on “Other...” will result in the following dialog...

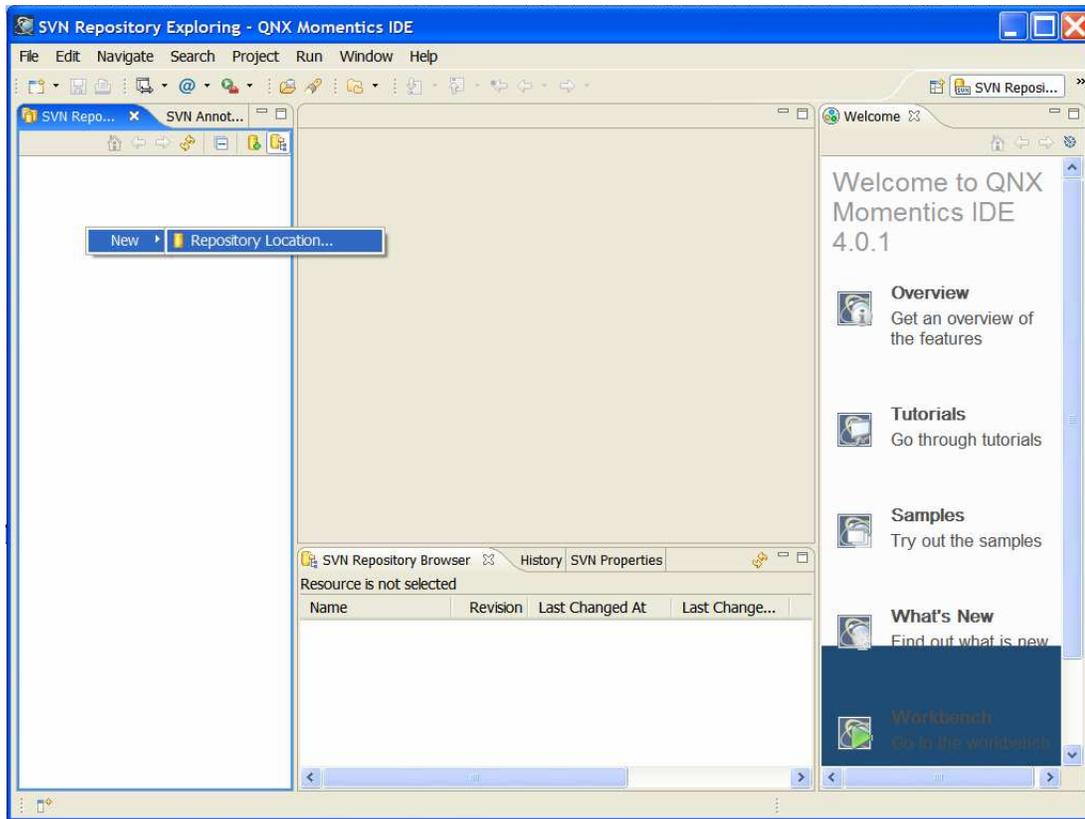


Select “SVN Repository Exploring” and click “OK”.

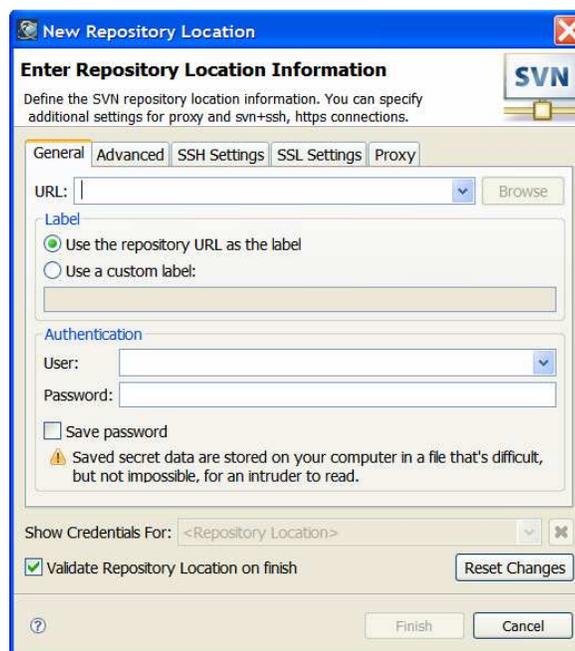
You should now see something like this...



Move the mouse over the “SVN Repository Exploring” pane (on the left), and right click. Select “New>” from the pop-up menu. The screen should now appear like this...



Clicking on the only option “Repository Location...” should yield the following dialog...



Fill in the URL, User and Password fields. Both the user and password fields are the same as the email address you supplied in the “Registering” phase earlier. The URL is:

[http://community.qnx.com/svn/repos/coreos\\_pub](http://community.qnx.com/svn/repos/coreos_pub)

Check the “Save password” field and select “Finish”.



The “SVN Repository Exploring” pane should now contain the source folders. This concludes the “Connecting” phase. If you are continuing with the process at this point, then leave the IDE in this state. If you will be continuing the process at a later time, then if you exit the IDE from this Perspective, when you re-start, it will automatically return to this point. In any case, this is where the IDE should be when you begin Phase 4.

### ***Phase 4: Downloading***

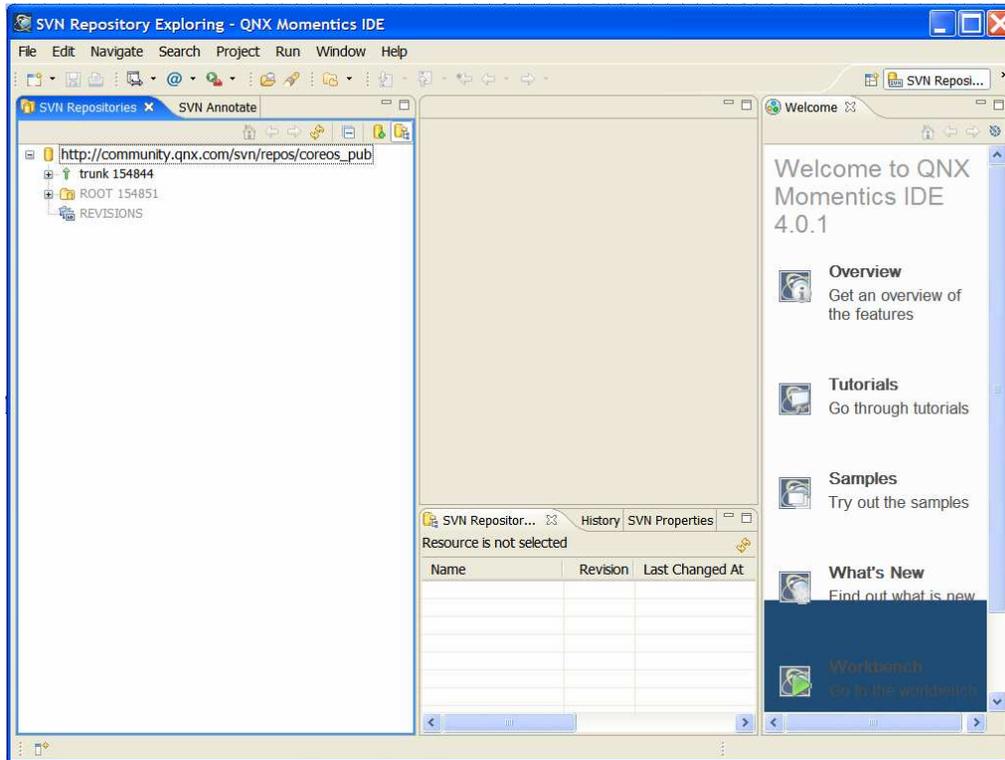
Let’s recap. At this point you have registered as a community member, you have downloaded, obtained a license for, and installed the tools. You have also connected the IDE to the community subversion server. Next you need to check-out the source code from the source repository.

This phase involves 2 steps:

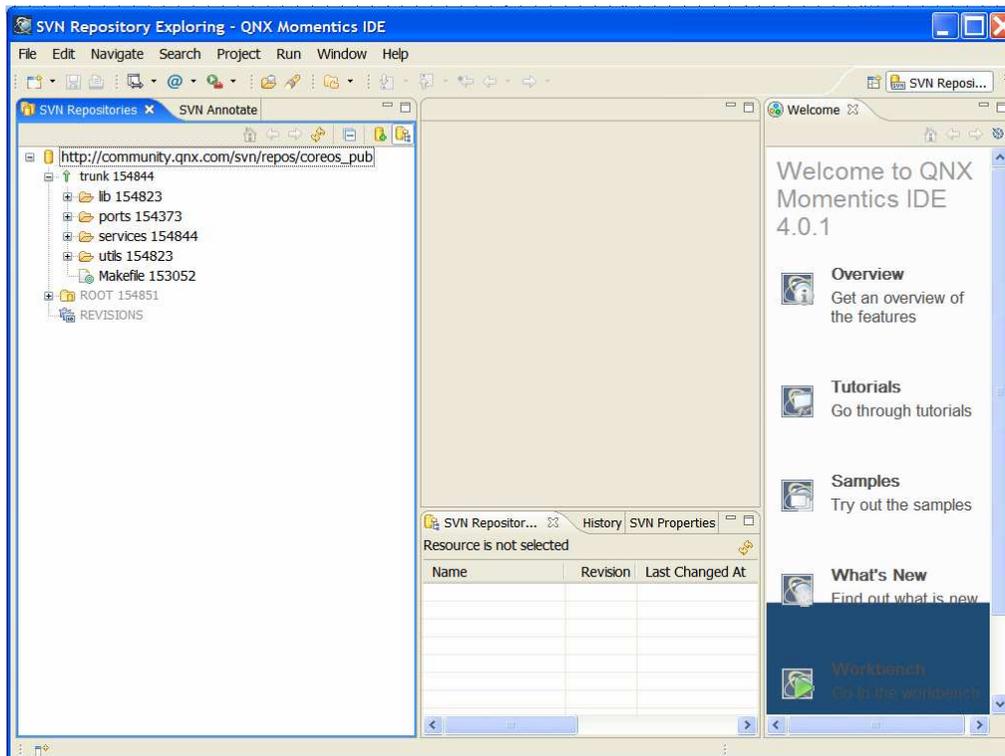
1. Check out the library source tree
2. Check out the services source tree

#### **Check out the library source tree**

The first set of source that must be checked-out is the libraries. As noted in the last paragraph of Phase 3, you should be in the “SVN Repository Exploring” perspective in order to begin the process of checking out. So you should be looking at something like this now...



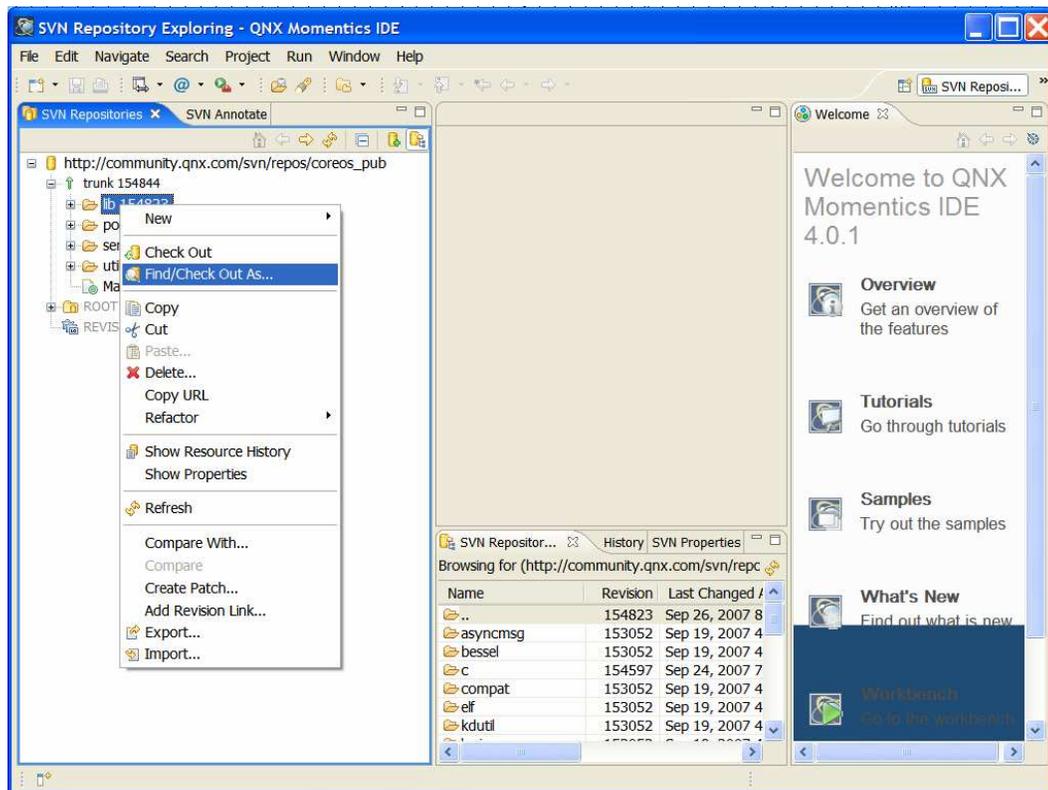
The view on the left shows the community repository, and is expanded one level deep to show the branches available. You are only interested in the “trunk” (ok, admittedly, trunk is not the best name for a branch). So expand trunk, and you should see the following...



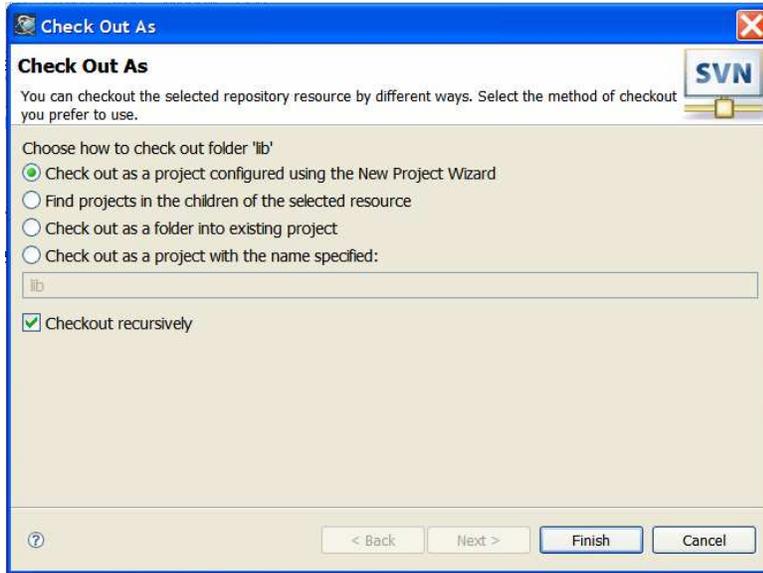
You are interested in two source trees:

- lib
- services

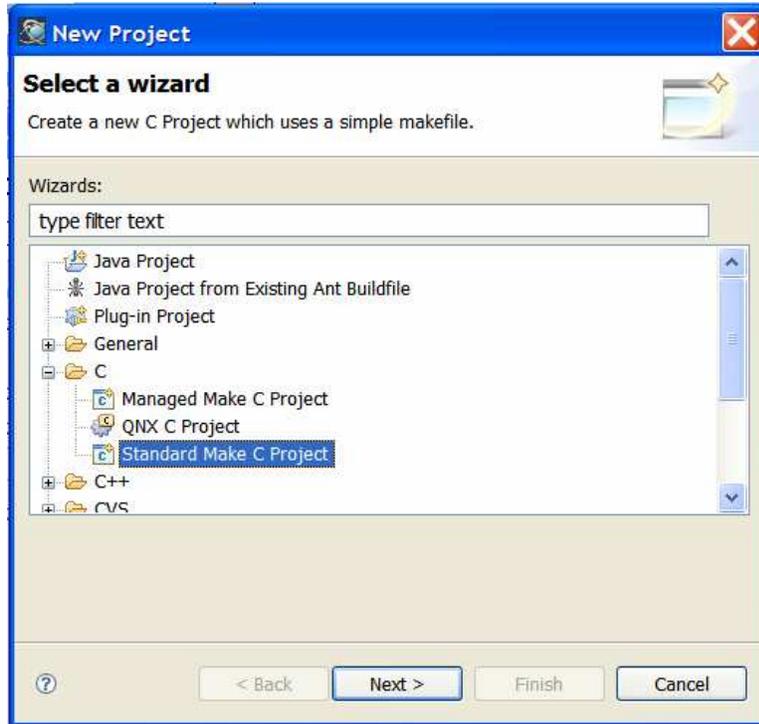
Let's check-out "lib" first. Right-click on the "lib" folder. You should be looking at the following...



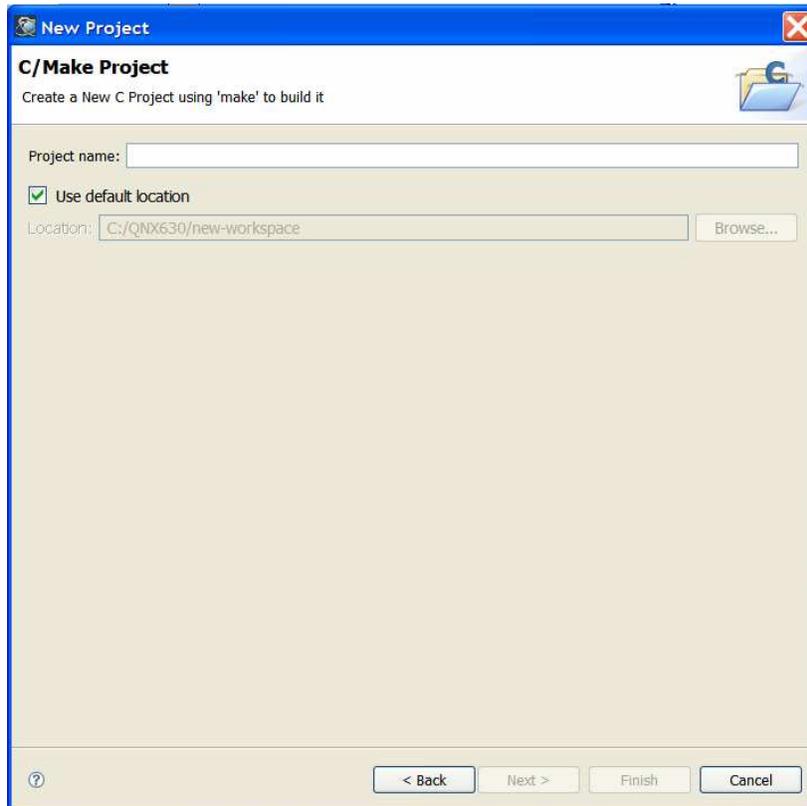
Select "Find/Check Out As...", which should result in the following dialog...



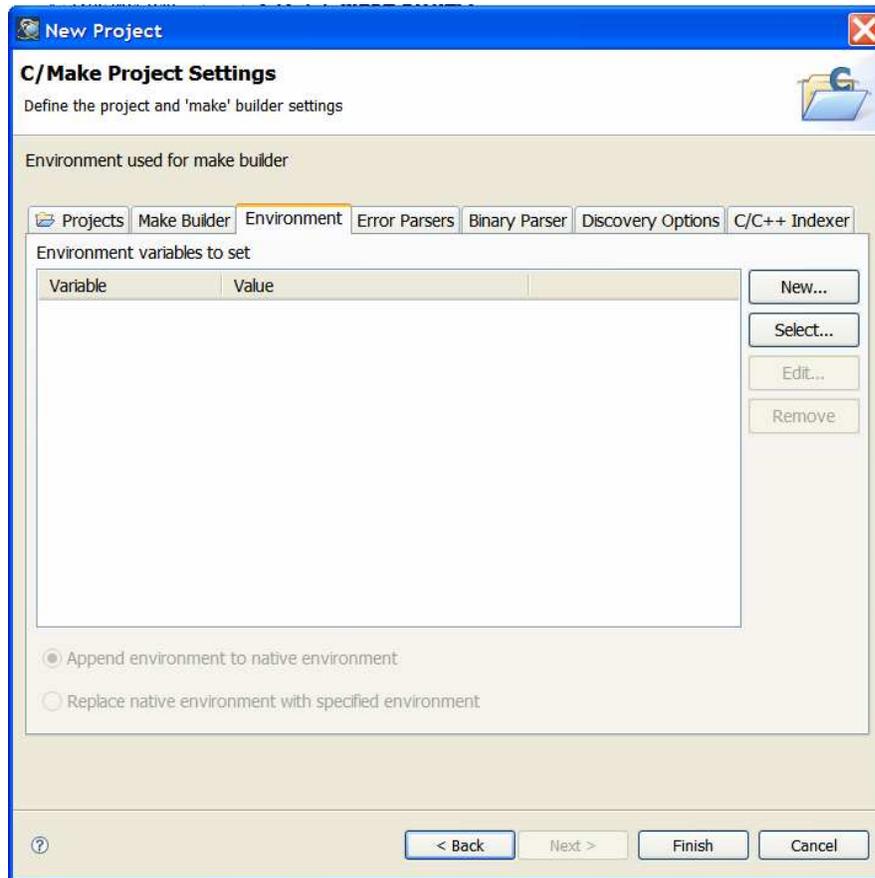
Select the “Check out as a project configured using the New Project Wizard” radio button, then click “Finish”. You will now be presented with this dialog...



Select a 'C' Standard Make C Project, and press next. You'll now see...



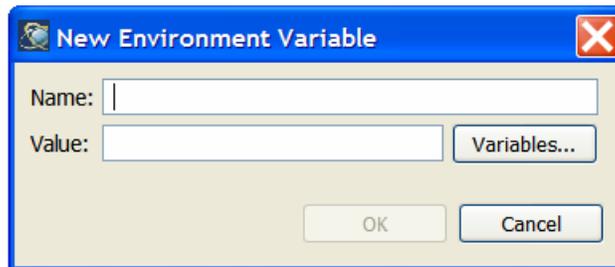
Here, you need to enter the project name. Enter “lib” as the project name, and click “Next”. You’ll then see...



On this dialog you need to create three environment variables:

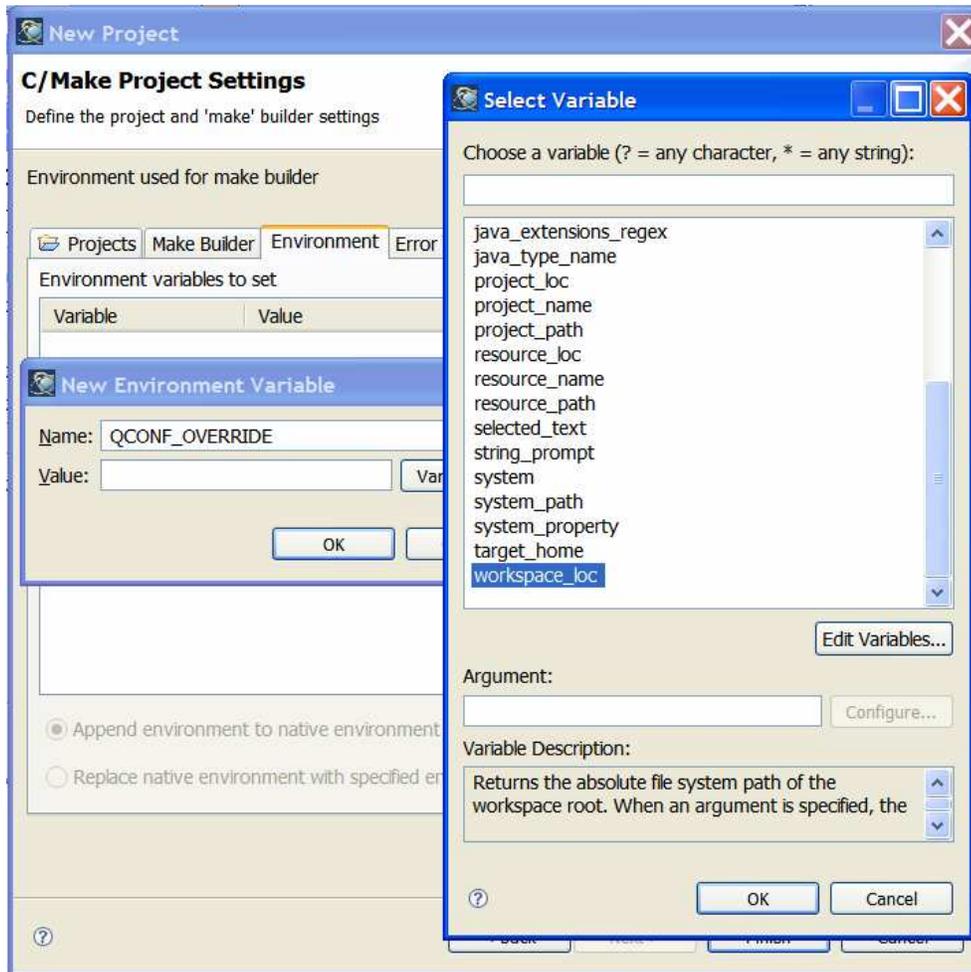
- CPULIST
- OSLIST
- QCONF\_OVERRIDE

To create an environment variable click “New”. You should get the following dialog...



To create the CPULIST environment variable, type “CPULIST” in the name field, then “x86” in the value field. For OSLIST type “OSLIST” in the name field, then “nto” in the value field.

QCONF\_OVERRIDE is a little more complicated. For QCONF\_OVERRIDE you can enter “QCONF\_OVERRIDE” for the name, but for the value field, you need to click on the “Variables...” button. This will produce a dialog like this...

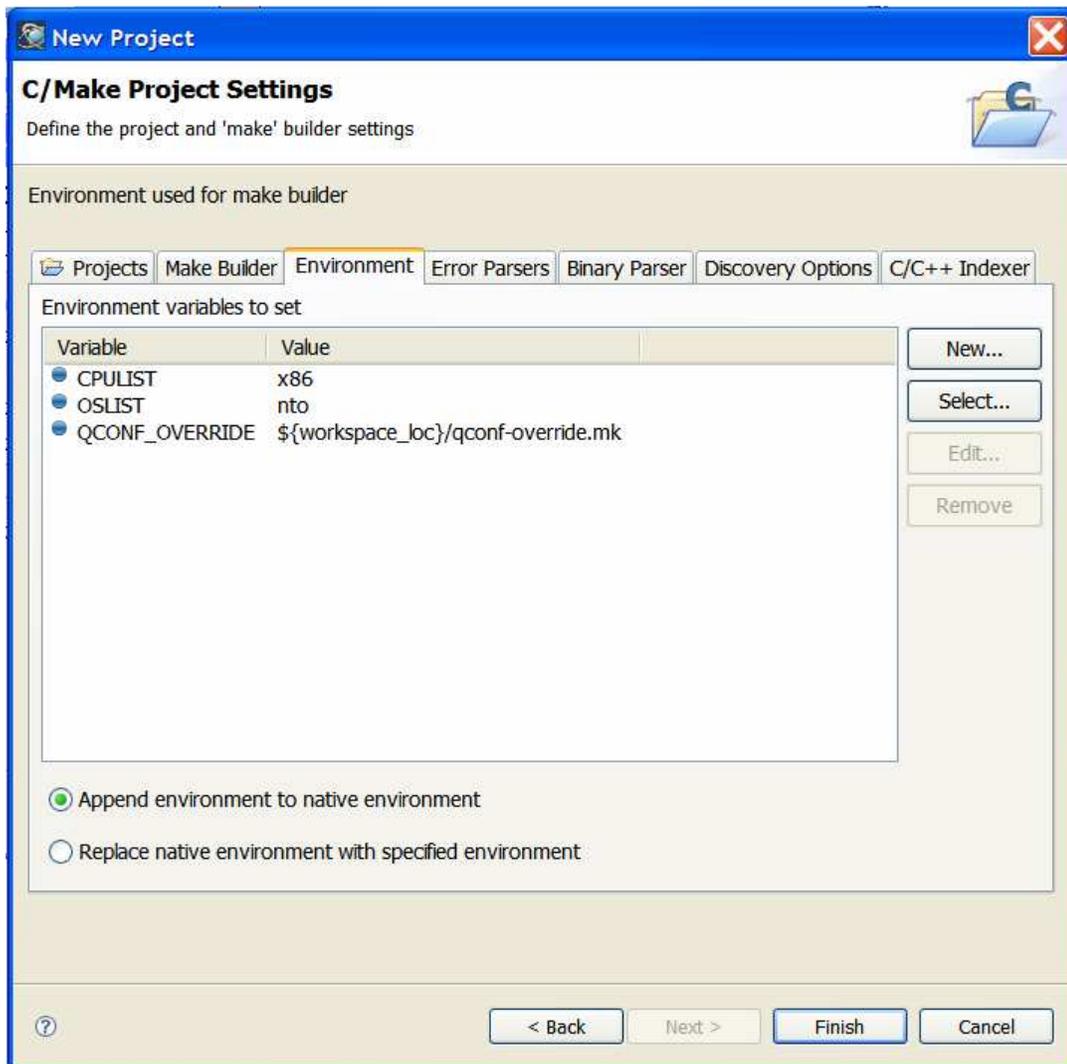


On the “Select Variable” dialog choose “workspace\_loc”, then click “OK”. This will enter “\${workspace\_loc}” into the value field. You then need to add “/qconf-override.mk” to the end of this string, so that the final result looks like this...

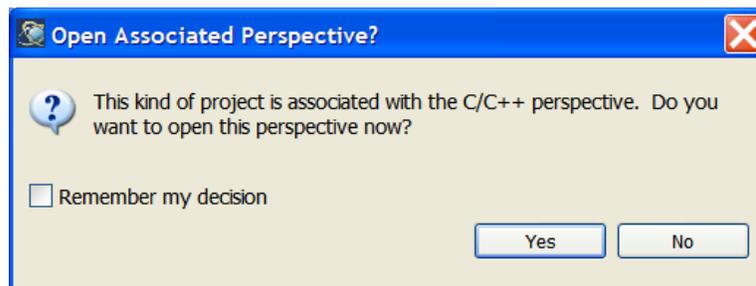


Click “OK”.

The “Environment variables to set” pane should now appear as follows...



Click “Finish”. You’ll be asked to confirm a switch to the “C/C++” Perspective. Go ahead and click “Yes”.



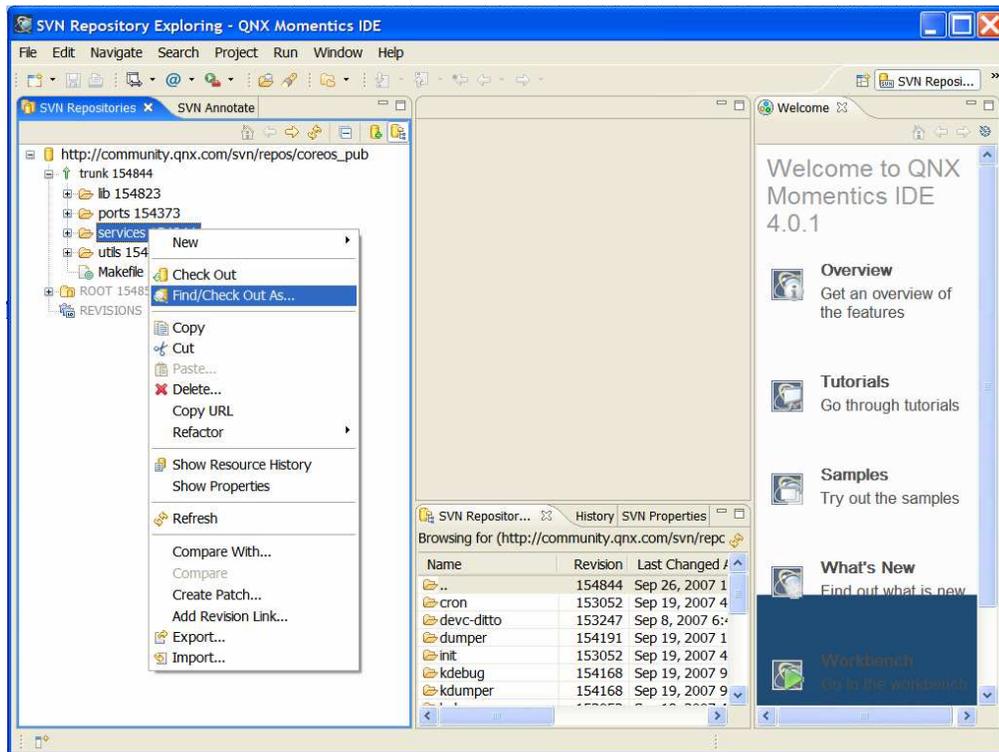
At this point the “lib” source will begin to check-out from the source repository. Depending on your connection speed this could take a while. When the “lib” source has finished checking out, you can proceed to “Check out the services source tree”.

## Check out the services source tree

Checking out the “services” source tree is identical to checking-out the “lib” source tree, with the obvious exception that “services” should be substituted for “lib”. All environment variables will be identical to those set for the “lib” source tree.

Following the check-out of the “lib” sources, the IDE automatically switched to the “C/C++” perspective, so to start the check-out of the “services” sources, you need to get back into “SVN Repository Exploring” perspective. To do this, select “Window->Open Perspective->Other...” and from the dialog select “SVN Repository Exploring”.

So you should now be here...



... and repeating the instructions for “Check out the library source tree” (exchanging “services” for “lib” where appropriate) will complete phase 4. The next phase is phase 5 “Building”.

## Phase 5: Building

Let’s recap. At this point you have registered as a community member, you have downloaded, obtained a license for, and installed the tools. You have also connected the IDE to the community subversion server, and checked-out the sources. Next you need to build the source code.

Building the source involves 4 steps:

1. Configure a staging area
2. Populate the staging area
3. Build the libraries
4. Build the services

## Configure a staging area

When building the system libraries, and kernel, a separation between the working tools environment, and the “sand-box” environment of the (potentially) new interfaces to the libraries and/or kernel needs to be created. If this were not done, then the possibility exists to “break” the known working tools environment.

You will now configure Momentics to build the libraries and kernel against “themselves” rather than against the installed tools environment. This allows new interfaces to be created and tested, without impacting the “stock” tools environment.

When you checked-out the source from the community repository, you instructed the IDE to check-out the source as a “standard make” project. The repository source that was checked-out includes the QNX recursive make environment. The QNX recursive make environment recognizes a special variable, that allows for building within a “sand-box” or “staging area”. You may recall that you set this variable (along with a couple of others) at the time the sources were checked out (QCONF\_OVERRIDE).

You may also recall what you set this QCONF\_OVERRIDE variable to. You should have set it to `${workspace_loc}/qconf-override.mk`. The variable `${workspace_loc}` will expand to the disk location of the IDE’s workspace. So what you need to do, is make sure that this `qconf-override.mk` file is present in the workspace folder (this will be “C:\QNX632\ide40-workspace” if the default values were used for the Momentics installation).

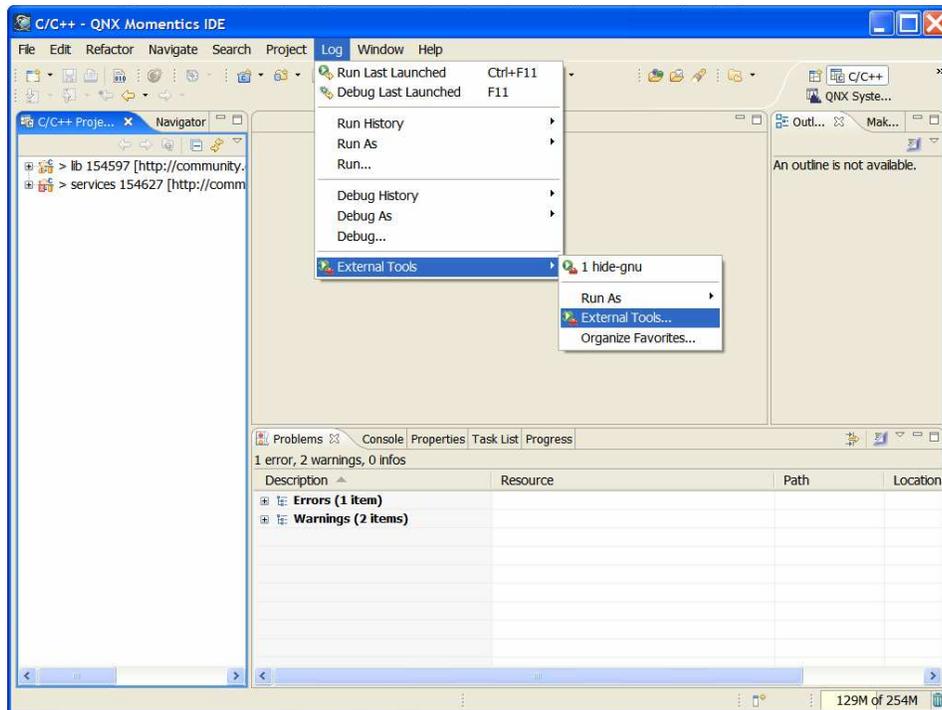
Using the Windows File Explorer, copy the two files you received in the email with this document (the files named `hide-gnu.sh`, and `qconf-override.mk`) into the workspace folder. You could create these files yourself, but they are required to be in unix text format and this is not straightforward to produce in Windows, so they are supplied for convenience.

Once these files are in the workspace folder, you may need to edit `qconf-override` if you did not install Momentics at the default location. If you did not install at the default location, then you need to edit `qconf-override.mk` with WordPad (it is very important to use wordpad and not notepad, as wordpad will preserve the unix text file format and notepad will not). Change the line:

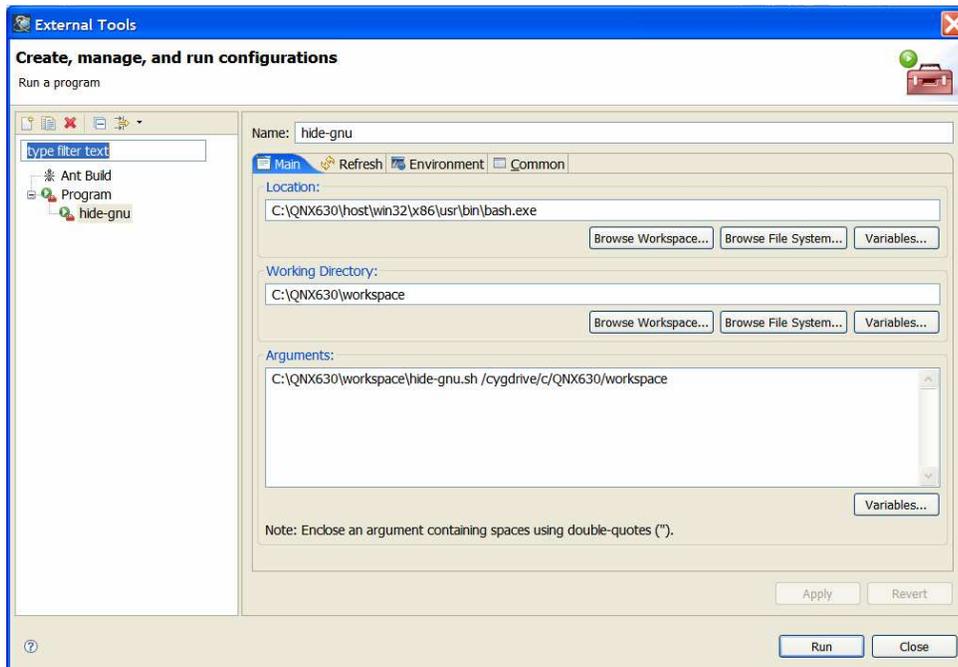
```
INSTALL_ROOT_nto=/QNX632/workspace/stage
```

to reflect the proper location of your workspace (be sure to start with a slash, avoid any reference to C: and use forward slashes, not back slashes) and a (yet to be created folder) named “stage” must be the final component of the path.

Once qconf-override.mk is correctly configured, you now need to run the “hide-gnu.sh” script. To do this, bring up the Momentics IDE, and select “Log->External Tools-> External Tools...”...



This should result in the following dialog...



Click on “Program” then click the new document icon (📄) and fill in the form as above, but using the correct path for your workspace. Nothing on any of the other tabs needs to be populated, so when you have provided the location of bash.exe, the working

directory, and the correct arguments (note the unix path conventions for the second argument), click “Apply” then “Run”.

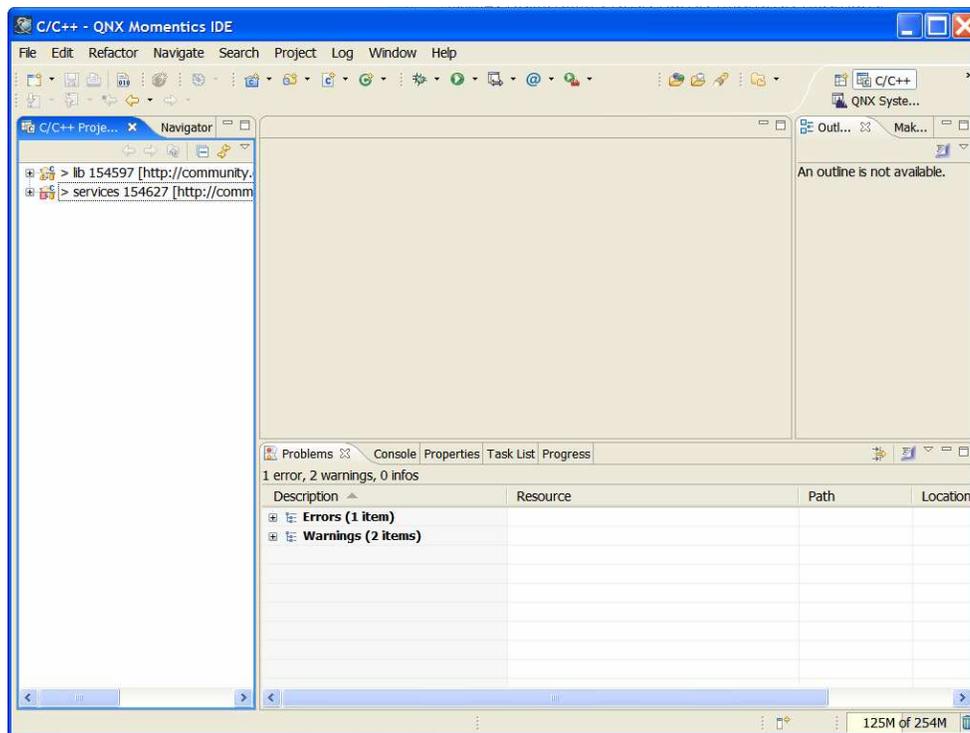
This script will setup the environment for a non-native build, and may take a while to run. After this script has finished running, the last thing that needs to be done is to create a folder named stage (mentioned earlier). You can do this with Windows file explorer.

With the complete execution of the “hide-gnu.sh” script, and the creation of the “stage” folder, the configuration of the staging area is now complete.

The next step is to populate the staging area.

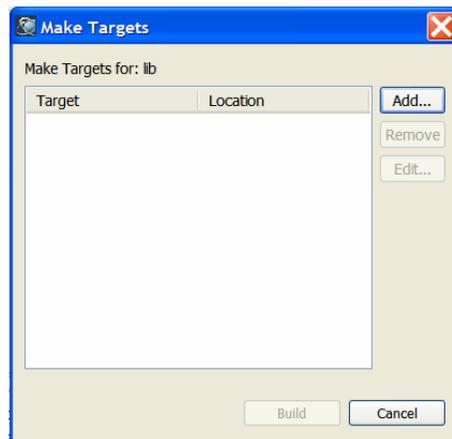
### Populate the staging area

Populating the staging area is easy, but it will take some time. Unless you have been exploring the Momentics environment, you should still be in the “C/C++” perspective that you were placed in after you checked-out the “services” sources. If you are not in the “C/C++” perspective, then get into that perspective by selecting “Window->Open Perspective->Other...” and selecting “C/C++”. You should now see this...

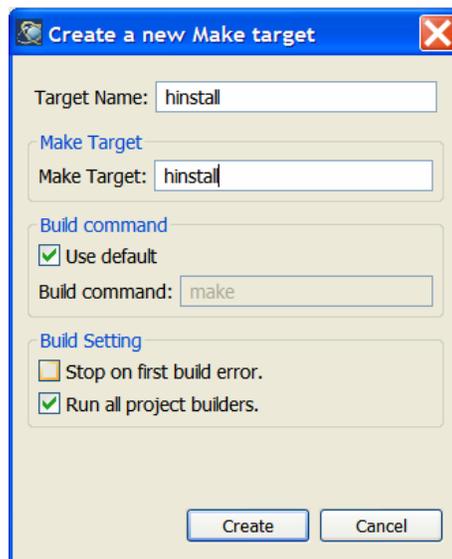


To begin populating the staging area, right-click on lib and from the pop-up menu select “Build Make Target...”.

This will result in the following dialog...



Click “Add...”, you’ll see...



Actually, you’ll see “all” for “Target Name” and “all” for “Make Target”, delete these and change both to “hinstall” as illustrated. Click “Create” then select the new target, and click “Build”. This will begin the process of populating the staging area.

Once this process has finished for the “lib” source, you’ll need to repeat this process for the “services” sources as well.

Once this process is complete for both the “lib” and “services” sources, then you have completed populating the staging area. Next step is to build the library sources.

### **Build the library sources**

Building the library sources is very easy. From the “C/C++” perspective, simply right-click on the “lib” project and select “Build Project” from the pop-up menu. This will take some time. When completed, the next step is to build the services sources.

## **Build the services sources**

Like the library sources, building the services sources is very easy. From the “C/C++” perspective, simply right-click on the “services” project and select “Build Project” from the pop-up menu. This will take some time. When complete there may be an error regarding something called “mig4nto”. This error can be safely ignored.

***Congratulations! You’re done.*** You now have a completely configured and populated development environment for the QNX kernel, essential services and libraries.

Once again, welcome to the Foundry27 community! We expect great things from you! Oh, and as stated at the beginning of this document, you can request commit rights by clicking on this link:

[http://community.qnx.com/sf/sfmain/do/myProjects?selectedTab=all&&\\_pagenum=2](http://community.qnx.com/sf/sfmain/do/myProjects?selectedTab=all&&_pagenum=2)