Installing Xith3D from SVN

Authors : Deive, Amos Wenger

Introduction

Using Xith3D from SVN provides you the latest and greatest improvements to the Xith3D API. Alas, sometimes, the sword can cut the other way : latest bugs induced by features additions or various optimizations are there also. Using the SVN version is fine if you want to stay tuned and have the maximum potential, but you'll probably have to change your code sometimes to conform to the API changes. Whichever versions you use, don't forget to report bugs, suggestions, or any comment to the forum : http://xith.org/forum

Enjoy!

1. Install Eclipse

Download your favourite Eclipse, mine is "EasyEclipse" (from http://www.easyeclipse.org) –   The best distribution is probably the "Expert Java" (http://www.easyeclipse.org/site/distributions/expert-java.html). I use this because I'm lazy – it has a nice installer with everything needed, most notably **Subclipse** is included!
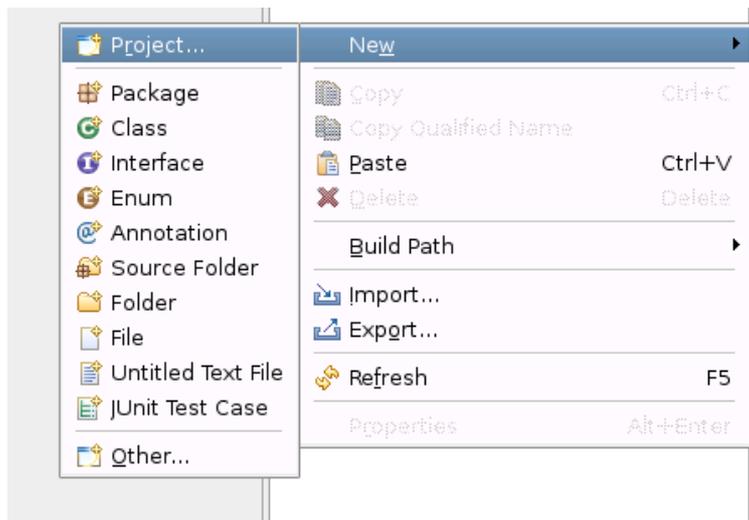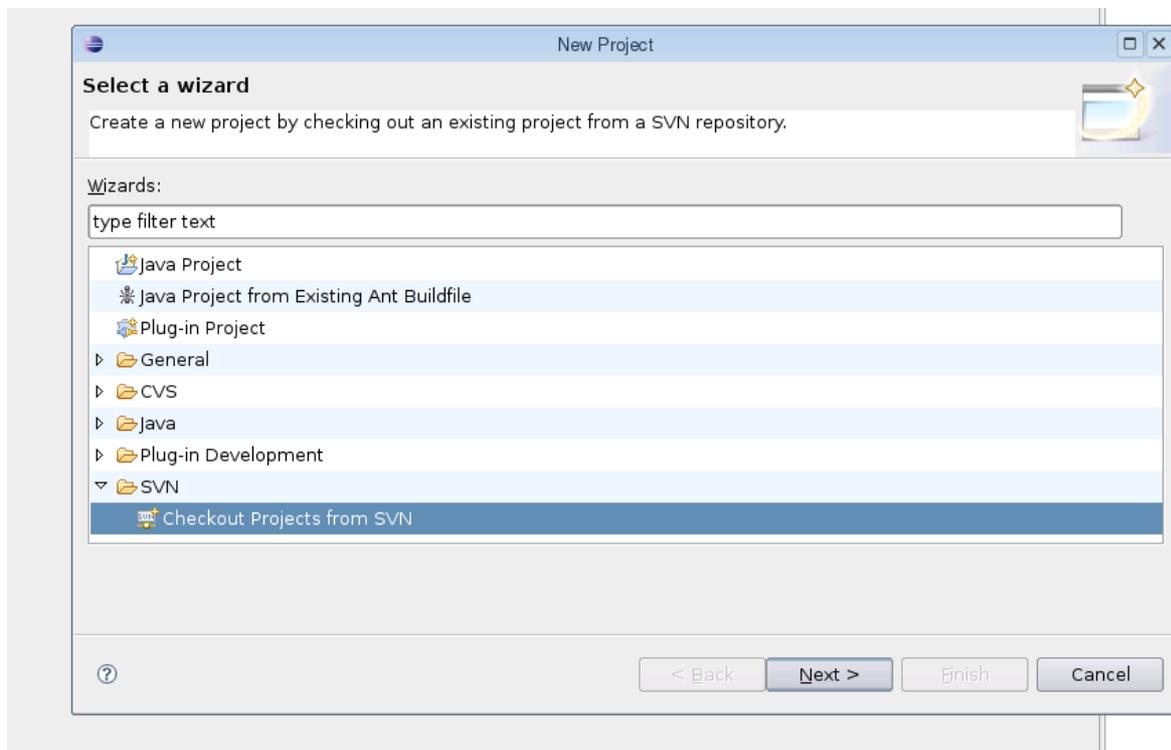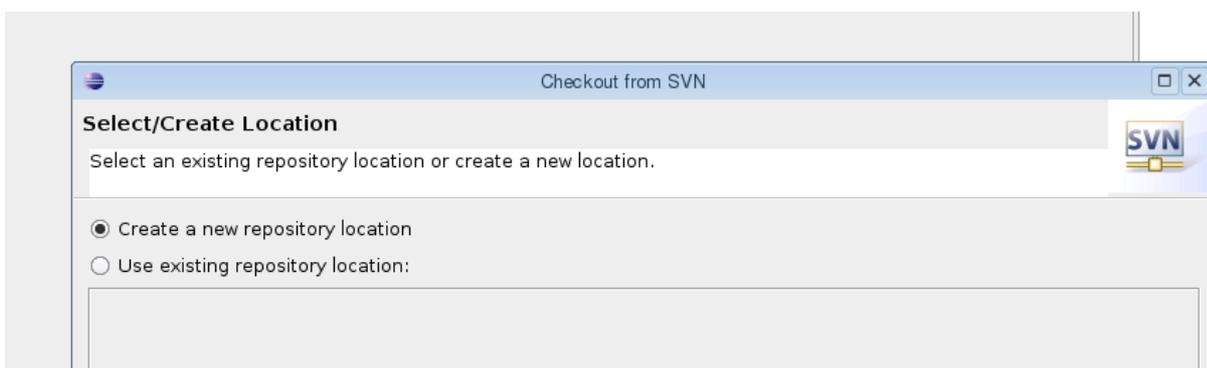
2. Checkout Xith3D core from SVN

- Start Eclipse

- *New project* (Right click in the Package area, "*New*", "*Project...*")

- *SVN, "*Checkout Projects from SVN*", Next.*

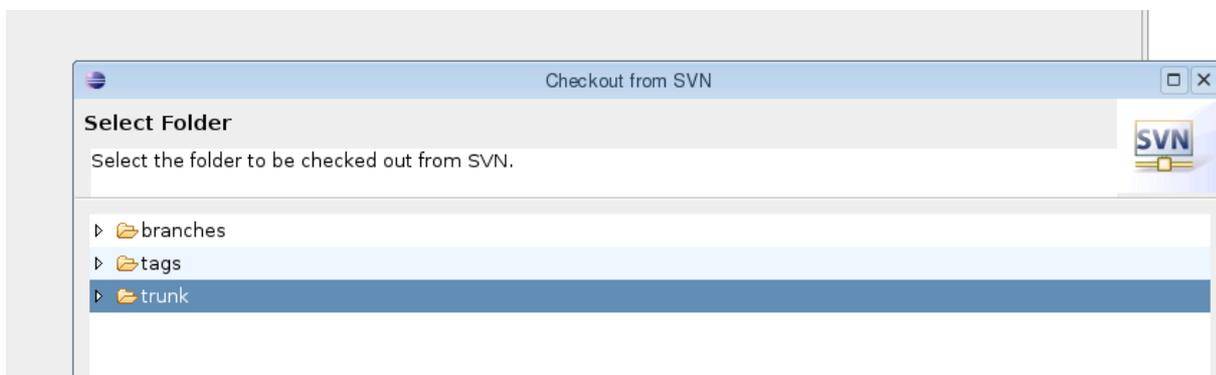- *"*Create a new repository location*", Next.*
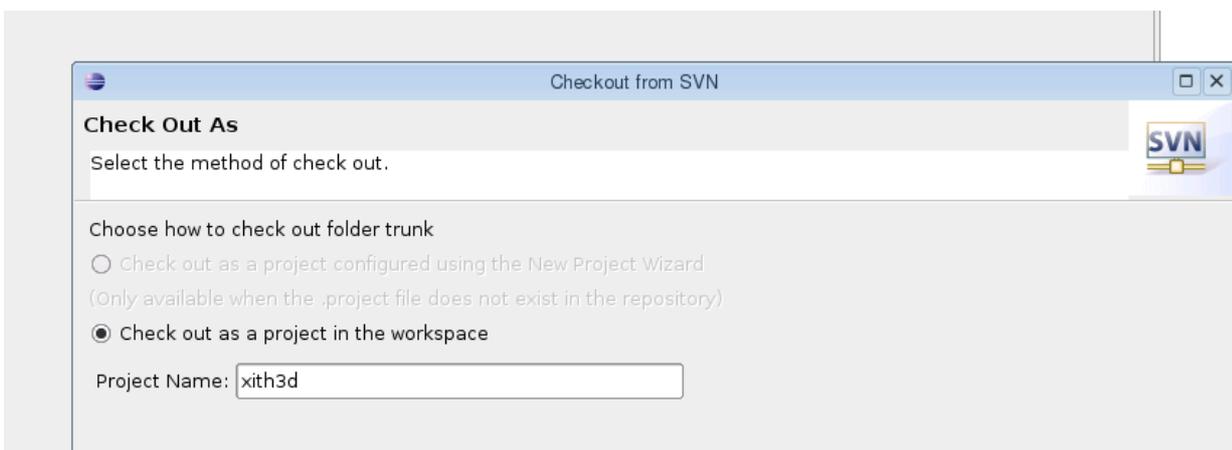
- Type "https://xith3d.svn.sourceforge.net/svnroot/xith3d" in URL, *Next*. (I get a load of "Error validating server certificate" messages, but just accept the certificate anyway!)
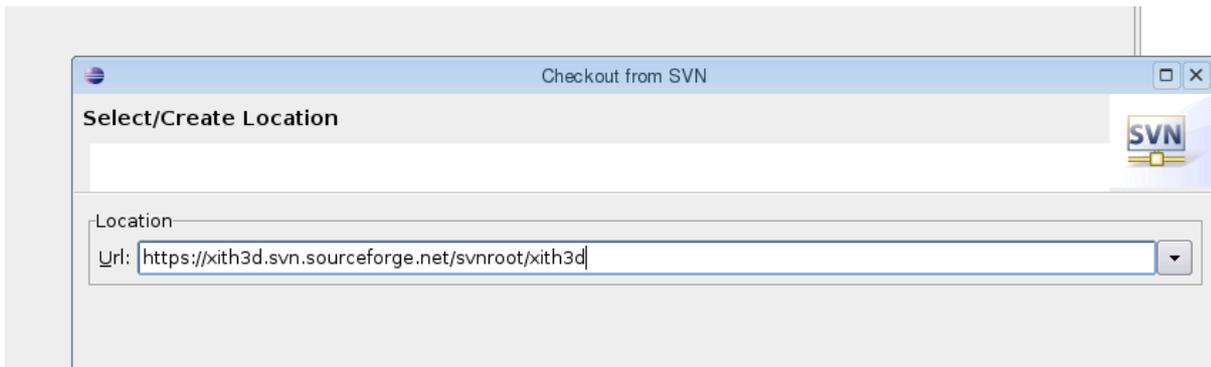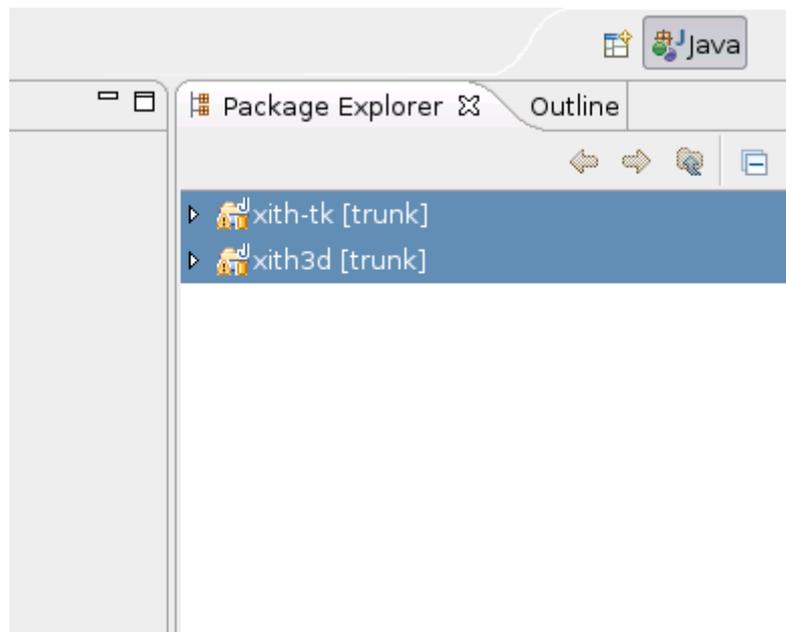

- Select "*trunk*", Next.

- Select "*Check out as a project in the workspace*", with *xith3d* as the project name (make sure you do not already have a project with this name!), Finish

3. Checkout Xith3D toolkit from SVN

- Repeat 2 replacing "*xith3d*" with "*xith-tk*" (at https://xith-tk.svn.sourceforge.net/svnroot/xith-tk)
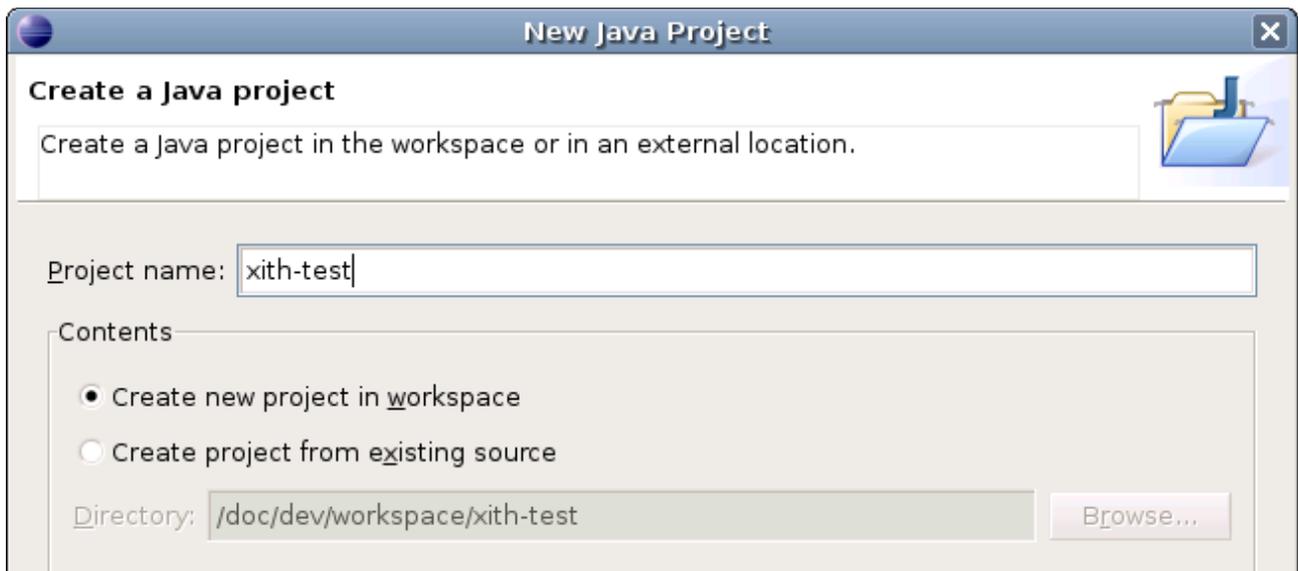- You should now have two new projects :
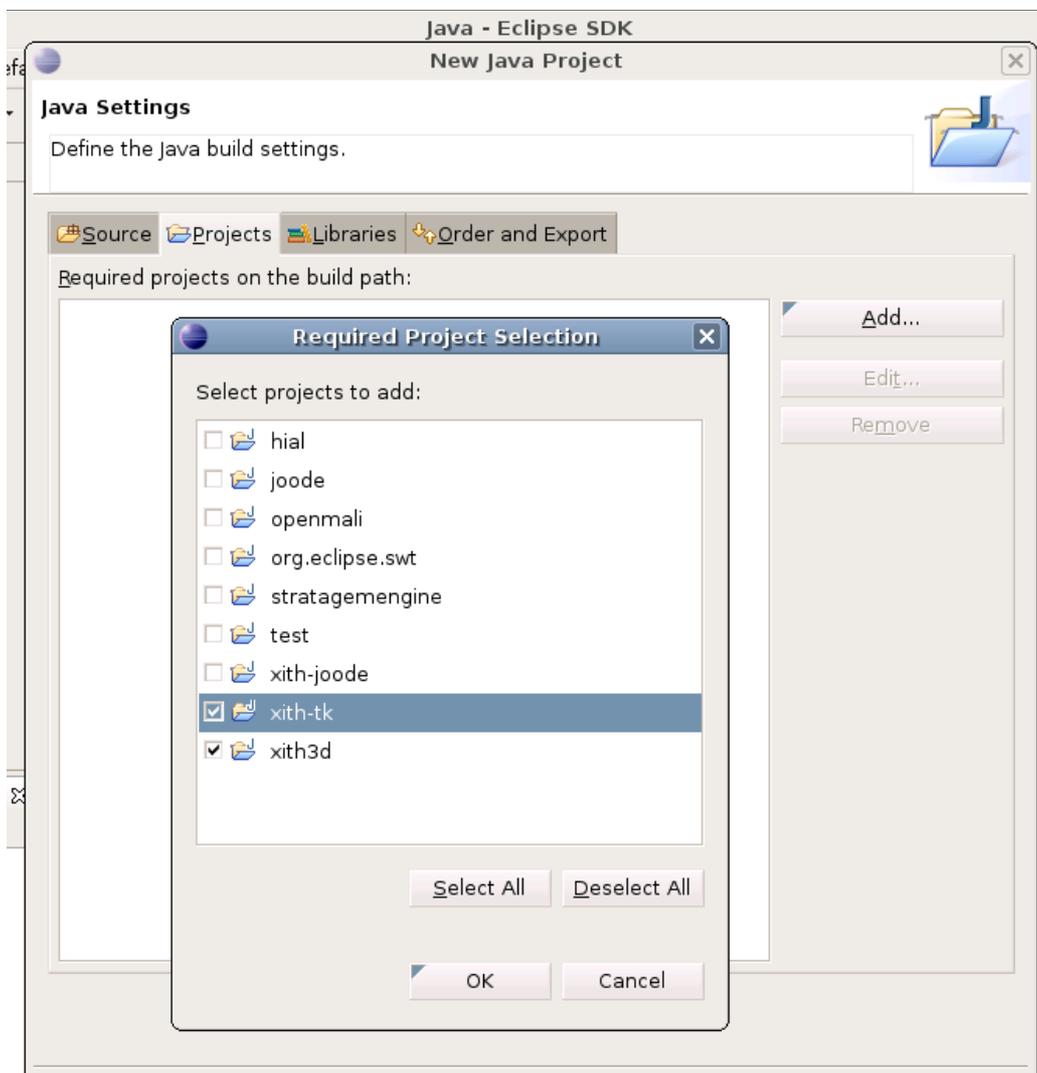
4. Create a test project
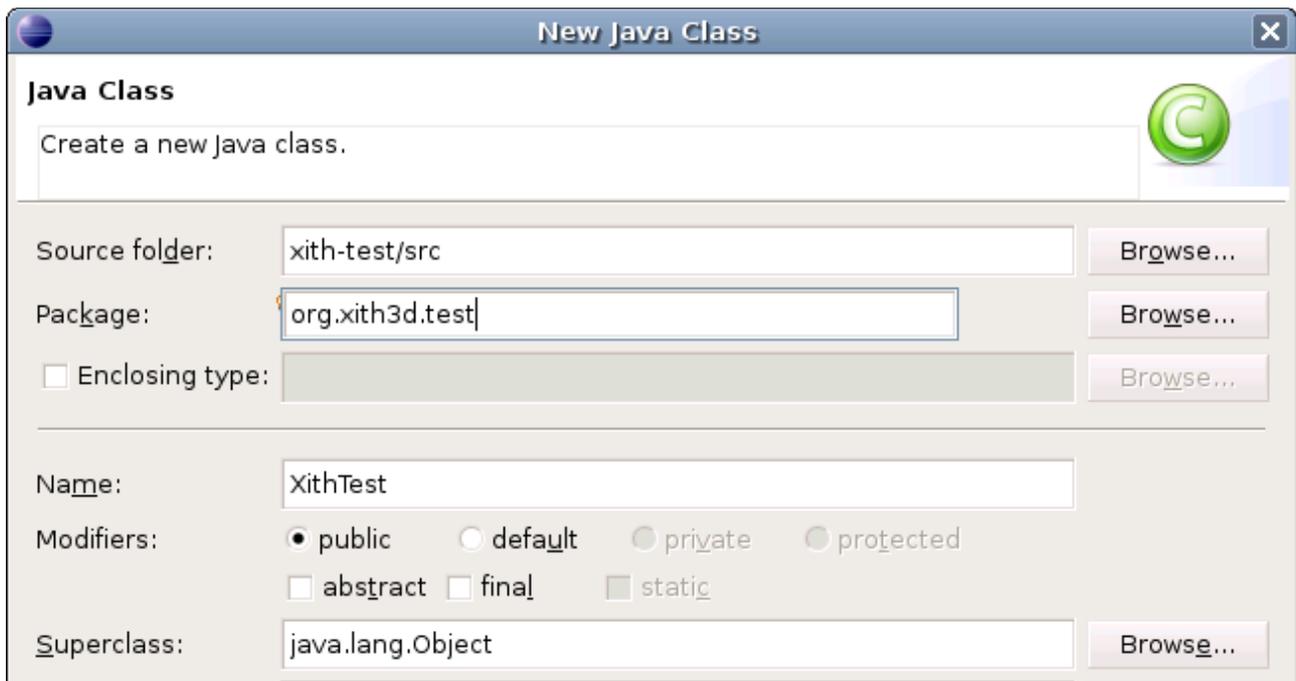
- New project, Java, Next.
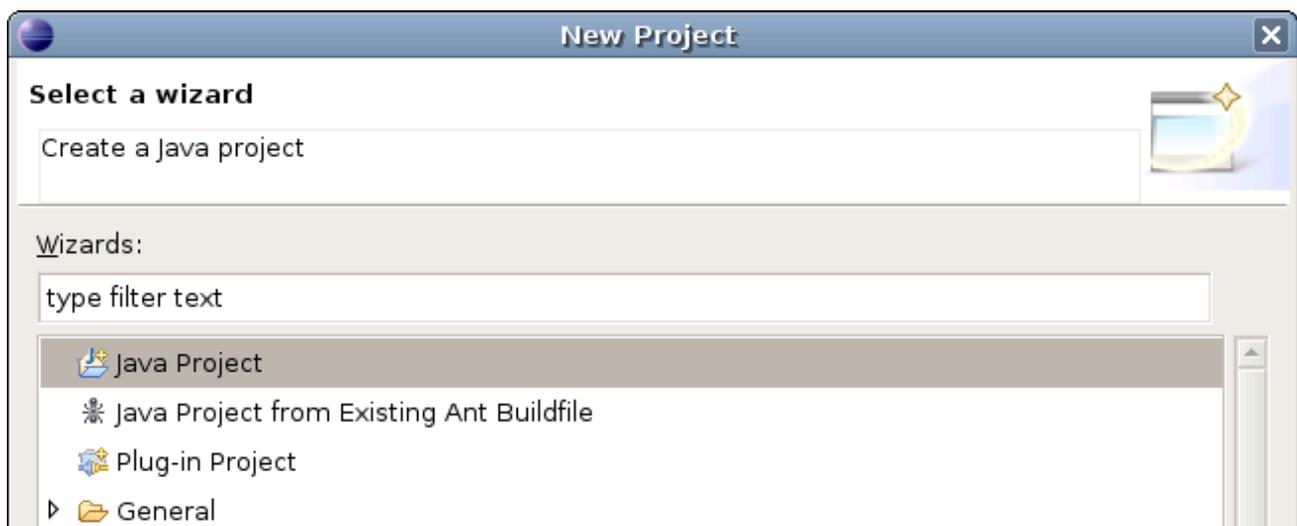
- Enter the name for the project e.g. "*xith-test*", Next.

- Under the Projects tab add both *xith3d* and *xith-tk*, Finish!

- Create a new class (in Package view : Right click "src" in xith-test, then "*New*", "*Class*") and call it "*XithTest*" and put it in the "org.xith3d.test" package.

- • Copy the following code into the file (modified from Xith In A Nutshell - 1st Edition):

```java
package org.xith3d.test;

import java.io.File;

import javax.vecmath.Vector3f;

import net.jtank.input.KeyCode;

import org.xith3d.geometry.Cube;
import org.xith3d.loaders.texture.TextureLoader;
import org.xith3d.loaders.texture.TextureStreamLocatorFile;
import org.xith3d.render.base.Xith3DEnvironment;
import org.xith3d.render.canvas.Canvas3DWrapper;
import org.xith3d.render.canvas.Canvas3DWrapper.Resolution;
import org.xith3d.render.loop.RenderLoop;
import org.xith3d.scenegraph.BranchGroup;

public class XithTest extends RenderLoop
{
    public void onKeyReleased(int key)
    {
        switch (key)
        {
            case KeyCode.VK_ESCAPE:
                this.end();
                break;
        }
    }

    private BranchGroup createScene()
    {
        Cube cube = new Cube( "stone.jpg", false, 3.0f );
```

```java
        BranchGroup bg = new BranchGroup();
        bg.addChild( cube );

        return( bg );
    }

    public XithTest()
    {
        super( 128L );

        Xith3DEnvironment env = new Xith3DEnvironment( new Vector3f( 3f, 2f, 5f ),
                                 new Vector3f( 0f, 0f, 0f ),
                                 new Vector3f( 0f, 1f, 0f ),
                                 this );

        Canvas3DWrapper canvas = Canvas3DWrapper.createStandalone(
                             Resolution.RES_800X600,
                             "My empty scene" );

        this.registerKeyboardAndMouse( canvas );



        env.addCanvas( canvas );

        TextureLoader.getInstance().addTextureStreamLocator(
                new TextureStreamLocatorFile( new File( "demo/textures") ) );

        env.addPerspectiveBranch( createScene() );

        this.begin();
    }

    public static void main(String[] args)
    {
        new XithTest();
    }
}
```

If *not* running on **Linux** a few project options need to be changed:

- Select properties on *xith3d*
- Select "*Java Build Path*"
- Choose the "*Libraries*" tab
- Select joal.jar
- Click "Edit..."
- Change the location from "xith3d/third-party/joal/linux" to "macosx" or "windows".
- Then for joal.jar, jogl.jar, and lwjgl.jar select the "Native library location", click "Edit..." and change this from "/linux" to your OS location.
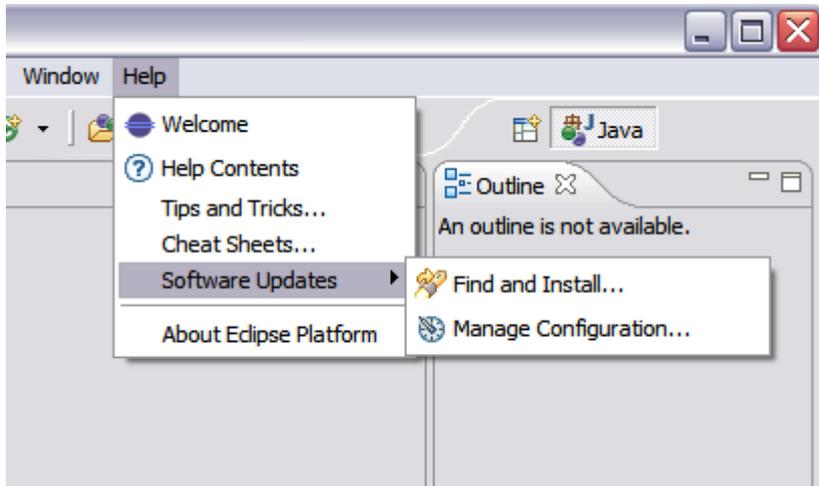
Now we can run XithTest!! (and get a cube, woo!)

Appendix A : Installing Subclipse

If you're not using the EasyEclipse distribution, you have to install Subclipse by yourself. But don't worry, it's really easy.
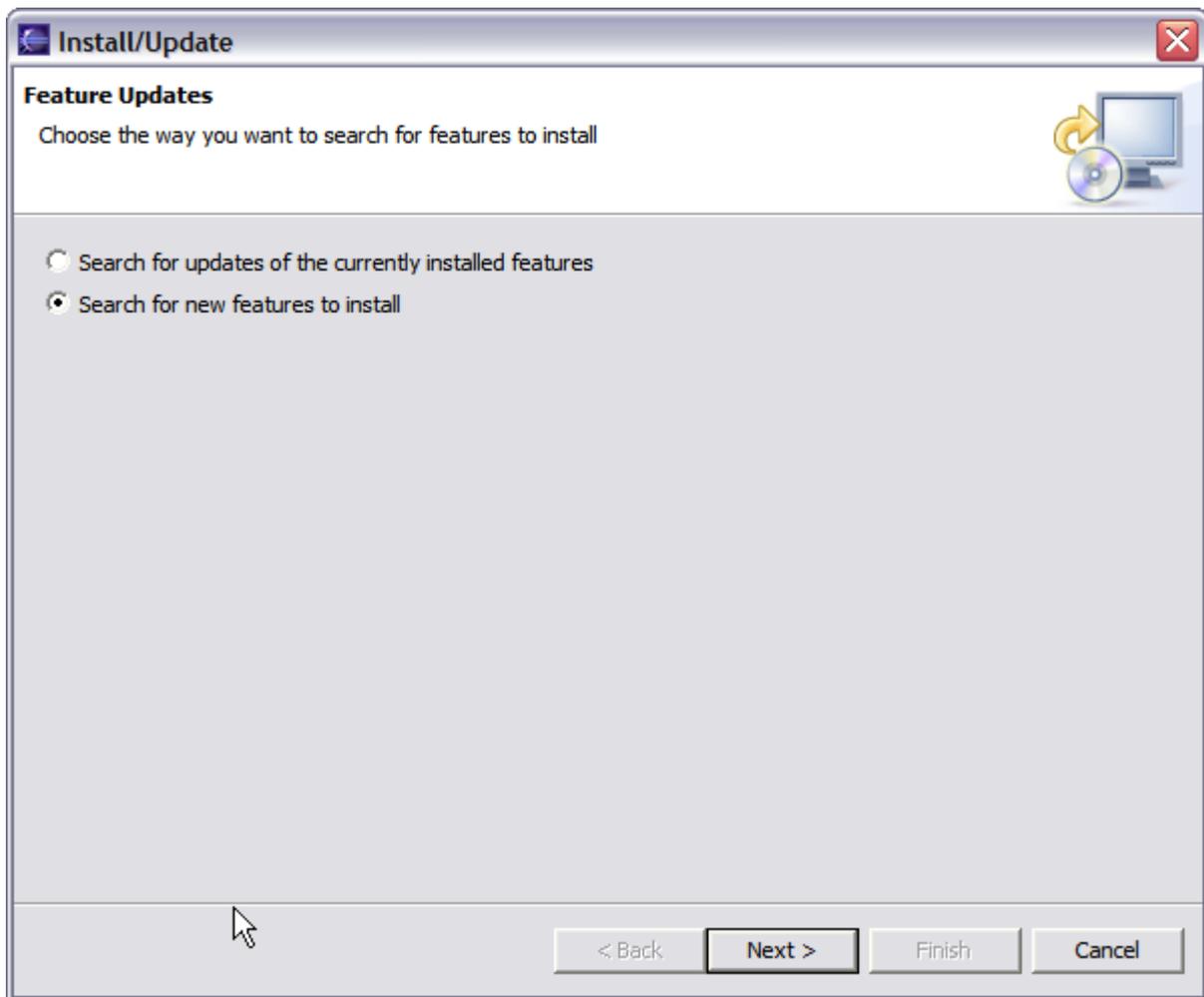
Step 1:

Begin the installation from the Eclipse Help menu item.


Step 2:

This screenshot show the screen as it initially comes up. In this case you will need to change the radio button to indicate that this is a new install.

Step 3:

This screen will vary depending on the features you have installed already. You want to click on the New Remote Site button. If you are behind a proxy and the Eclipse install mechanism does not work, then you can download a zipped version of the update site and then click the New Local Site button instead.
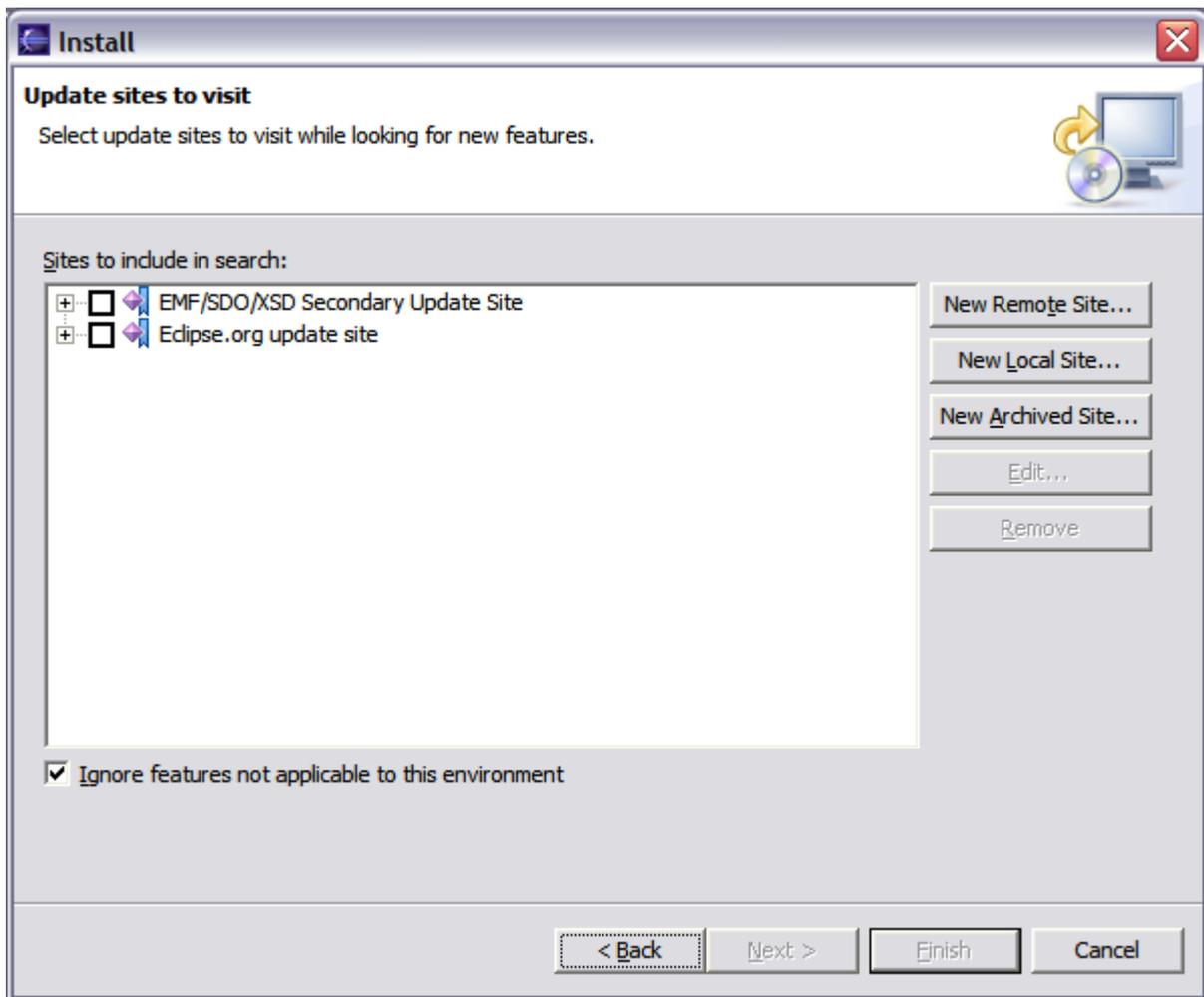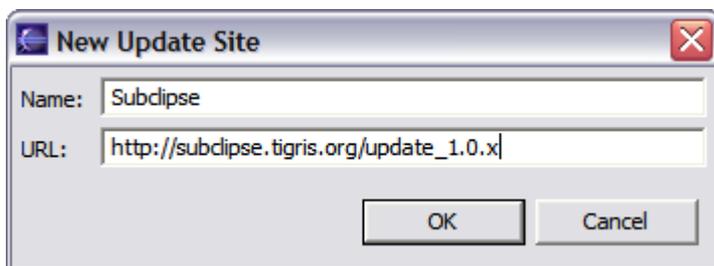
Step 4:

This screen is showing the New Remote Site dialog, filled in with the correct information to install Subclipse
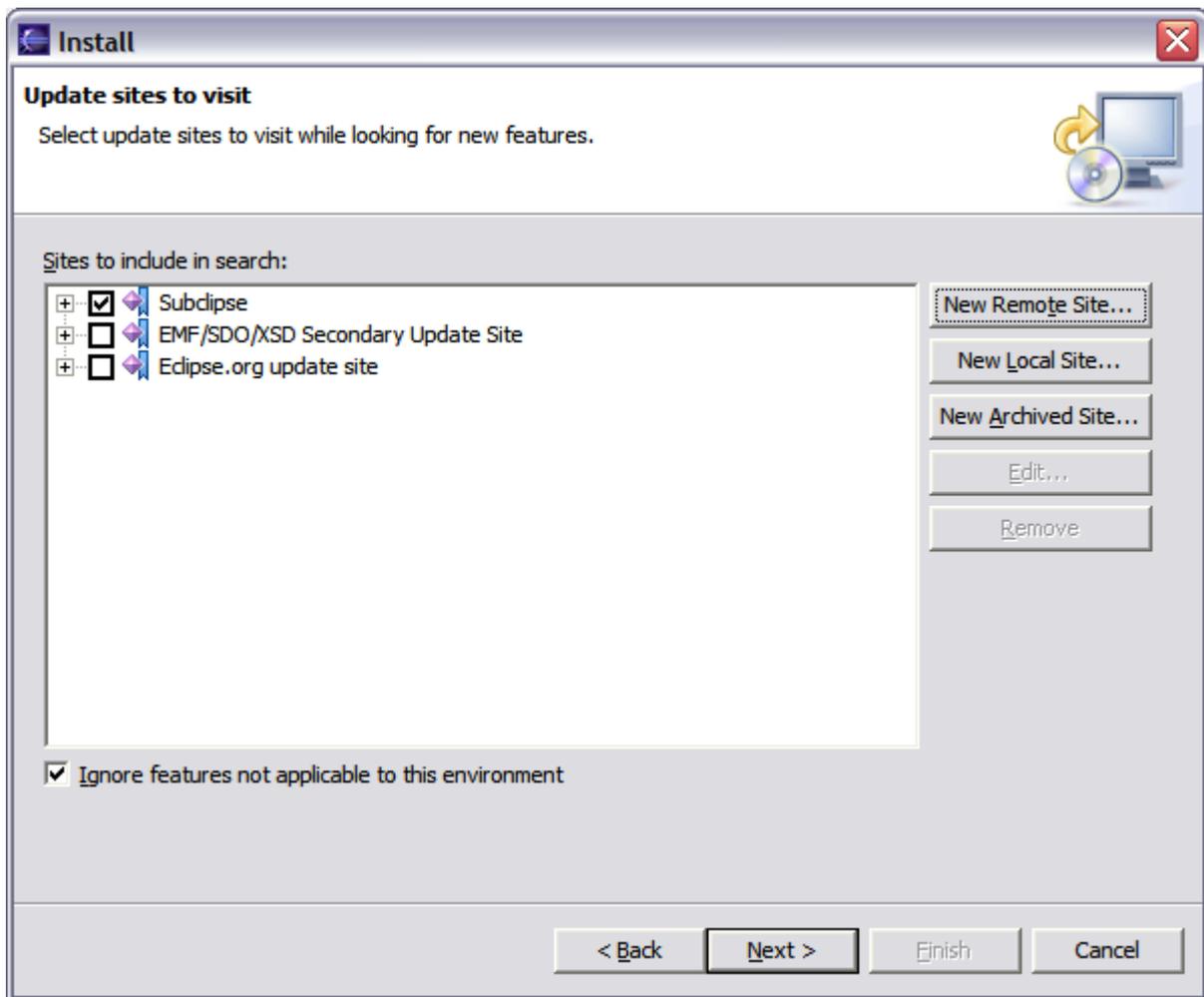
Name: Subclipse
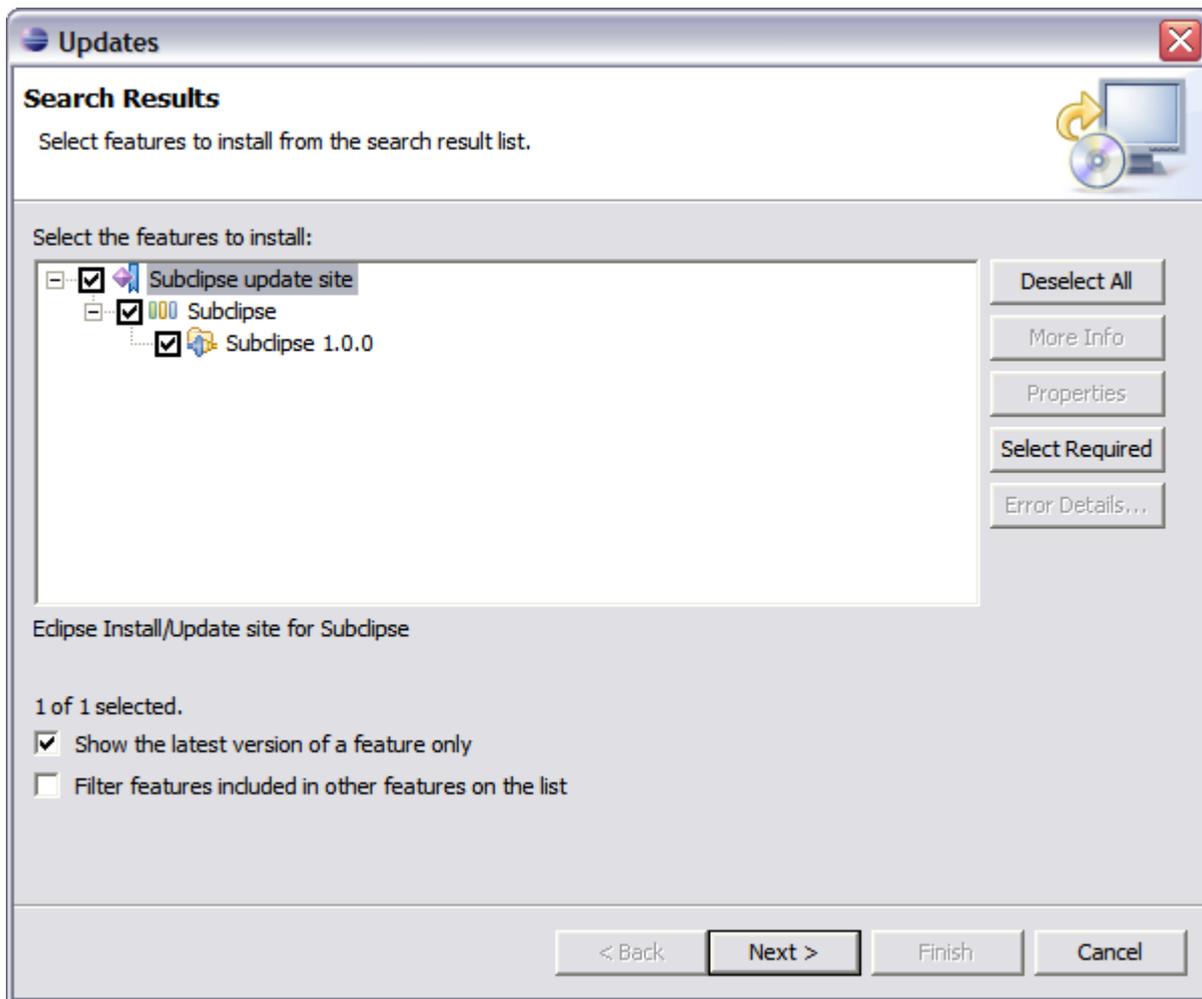URL: http://subclipse.tigris.org/update_1.0.x

Step 5:

When you first come back to this screen, the site you added will NOT be selected. Be sure to select it before clicking Next.
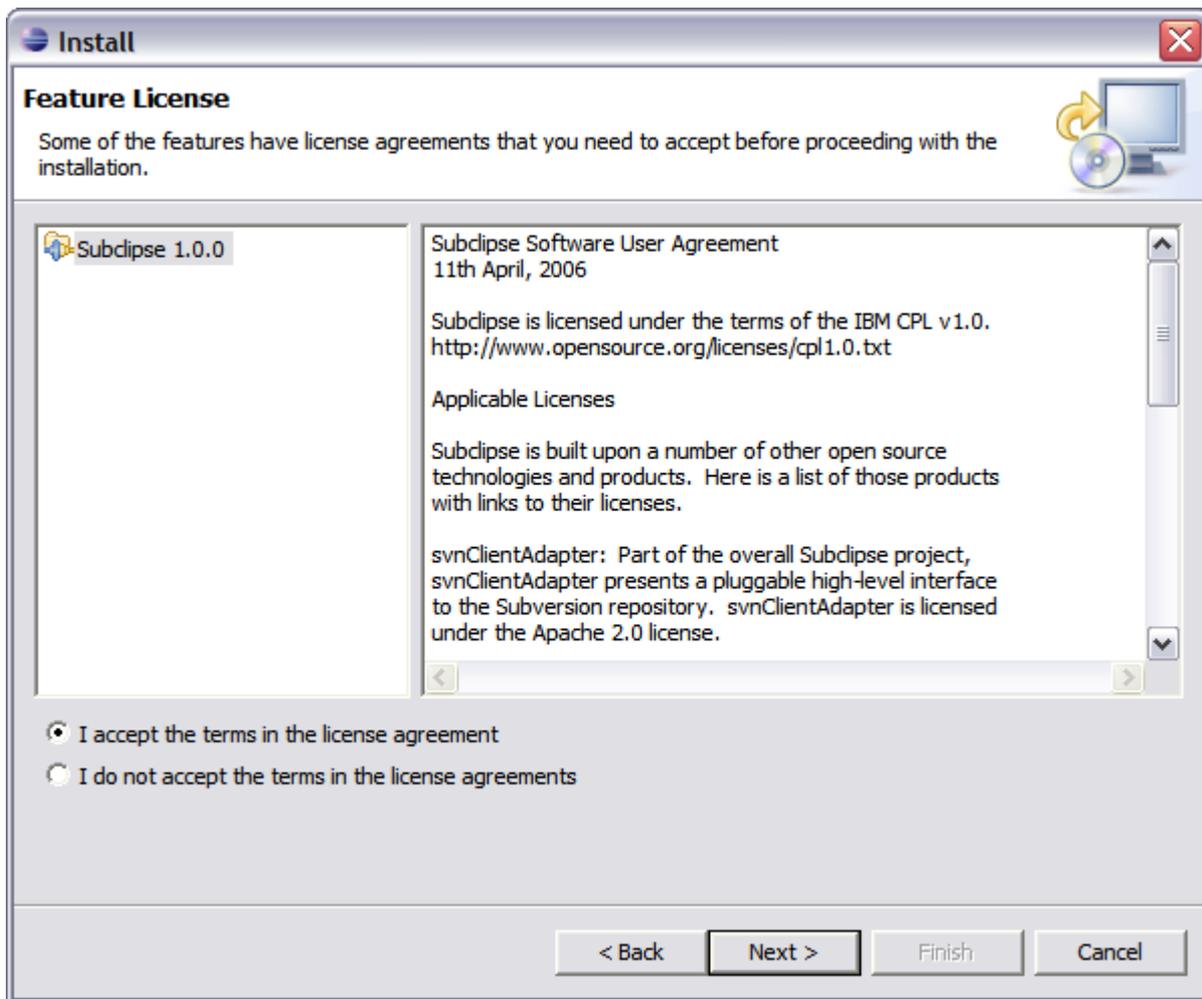
Step 6:

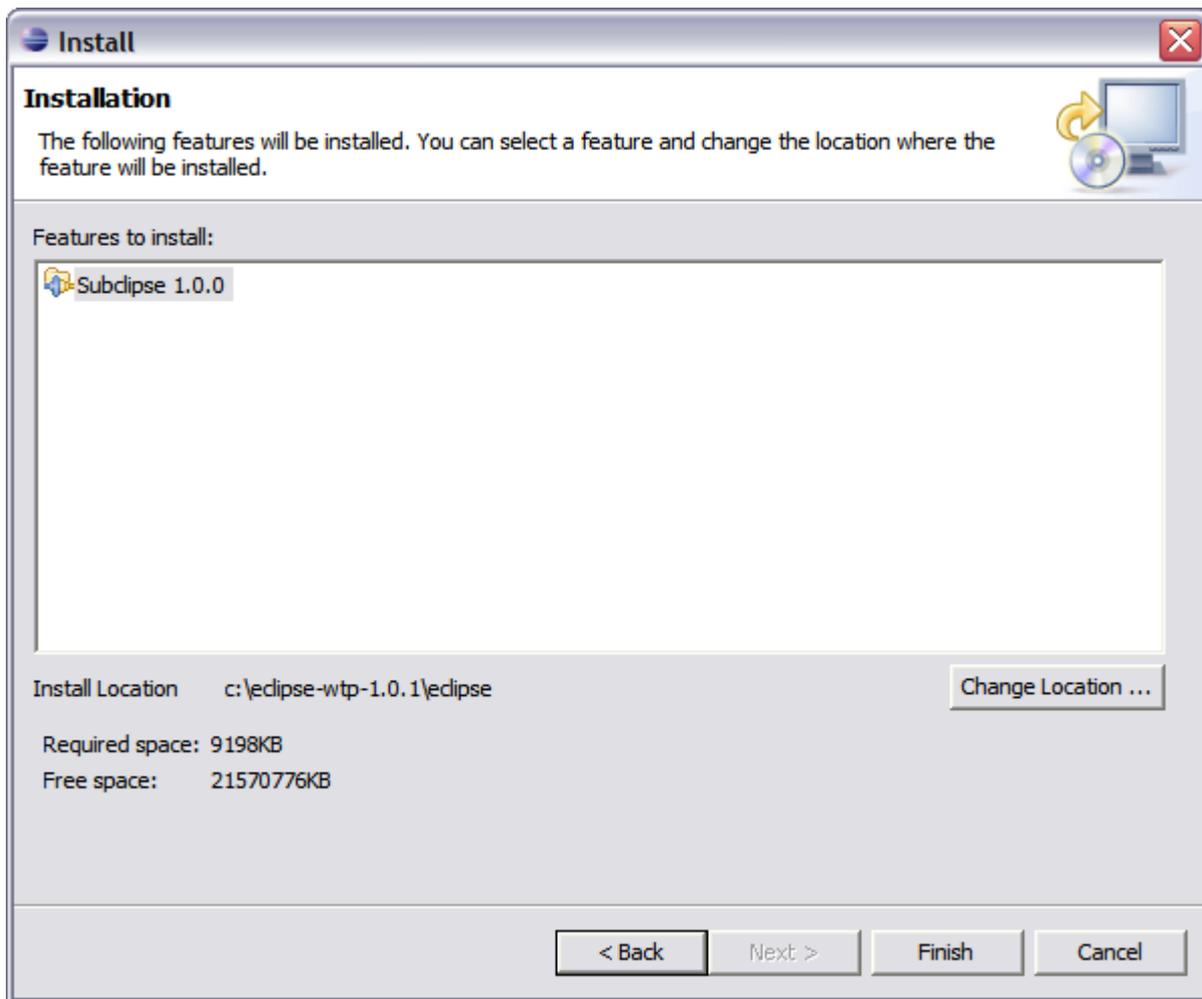This next screen shows all of the features that are available to install.

Step 7:

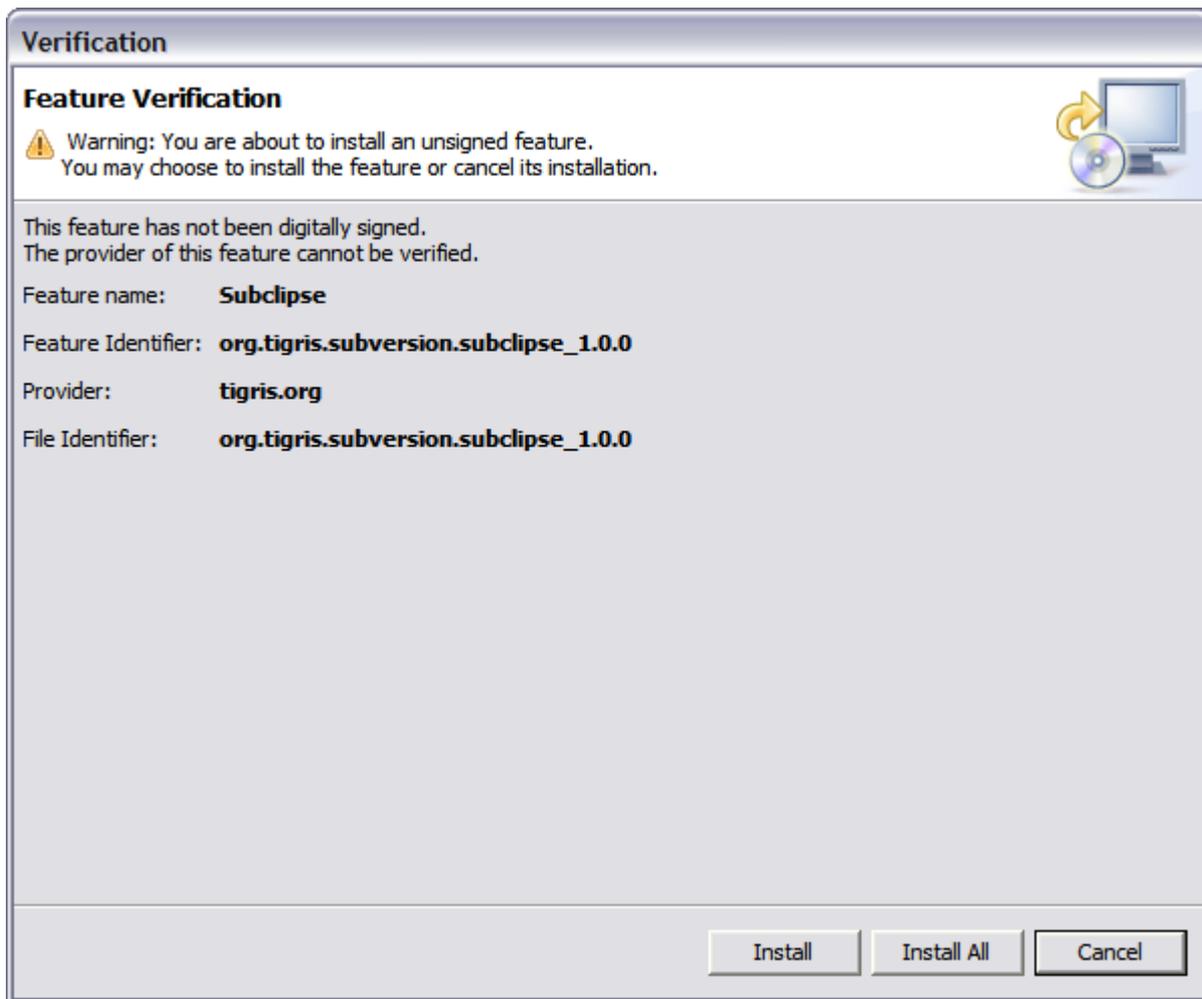Click the button to accept the license agreement.

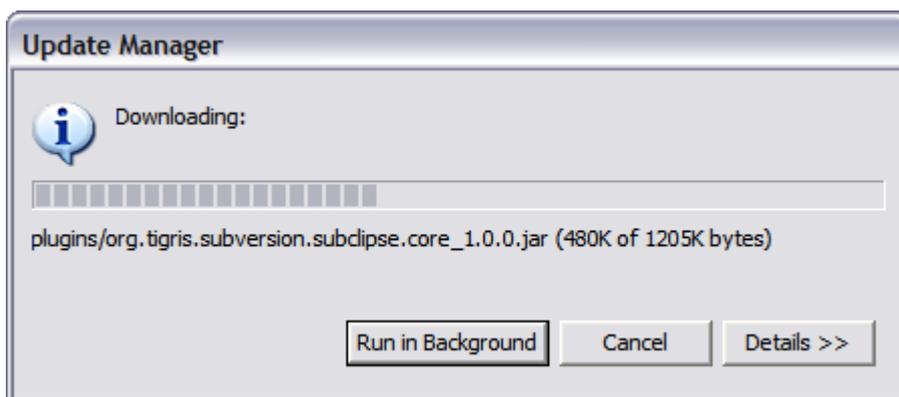## Step 8:

Confirm the install location

Step 9:

There is an Eclipse preference to turn off this next dialog. I have never seen a signed feature. Not even Eclipse.org nor IBM sign their features.
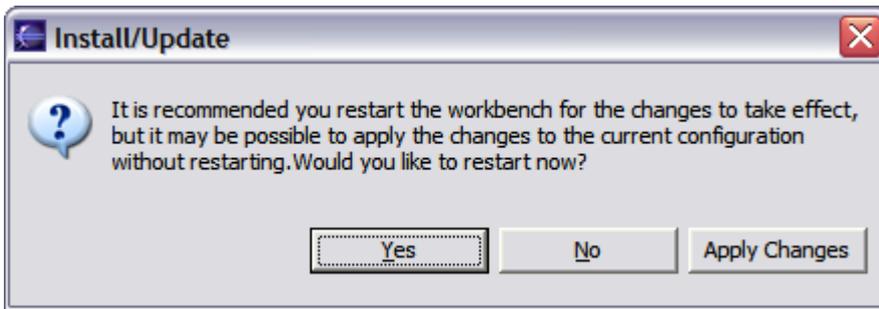
Step 10:

Just a screenshot of the in-process installation.

Step 11:

Eclipse needs to be restarted after installing Subclipse.

Install/Update

? It is recommended you restart the workbench for the changes to take effect, but it may be possible to apply the changes to the current configuration without restarting. Would you like to restart now?

[Yes]   [No]   [Apply Changes]

Step 12:

Finally, after restarting Eclipse, the first thing you will typically want to do is open the Subclipse Repository perspective where you can define your repositories. Be sure to also check the online help as well as the Subclipse preferences located under Team -> SVN.

Select Perspective

- CVS Repository Exploring
- Debug
- Java
- Java Browsing
- Java Type Hierarchy
- Plug-in Development
- Report Design
- Resource (default)
- SVN Repository Exploring
- Team Synchronizing

[OK]   [Cancel]