

Project Wonder Installation on Windows

Work in progress!

This page is designed to be a ground floor installation procedure for installing Project Wonder, including: Eclipse, WebObjects and the WOLips plugin. This guide assumes Windows 7 (procedure should be the same for Windows XP and Vista).

There are four sections to this installation process.

Step 1: Install a JDK 1.6

Before installing anything else, you need to install a JDK. We suggest that you install the official [JDK 1.6 from Oracle](#). Don't install a JRE, the tools really need the JDK.

Step 2: Install the latest version of Eclipse

If you already have Eclipse installed, you may skip this step.

Head over to the Eclipse Foundation and [download](#) the latest version of Eclipse 3.7 suitable for your Windows version. We suggest the "Classic" bundle of Eclipse.

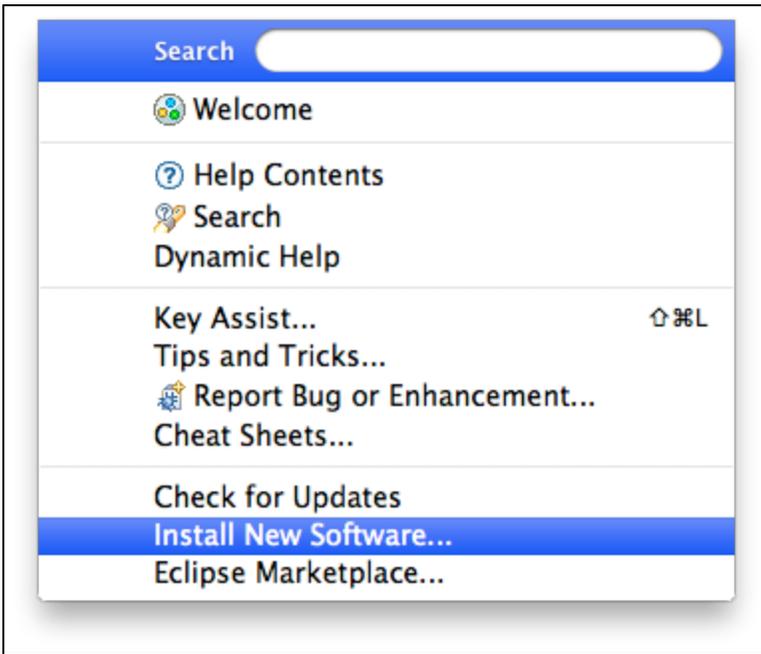
Eclipse does not install as a package, rather it is simply a compressed archive. Installation is as simple as extracting the Eclipse folder, entering it and running the Eclipse executable. In this context, it is important to remember where the folder resides on your system for future use.

Install Eclipse in a path with no spaces in it! Eclipse is sensitive to that and can be problematic. Same thing for the workspace, put the Eclipse workspace in a path without any spaces in it.

Step 3: Install WOLips plugin and the WebObjects Framework

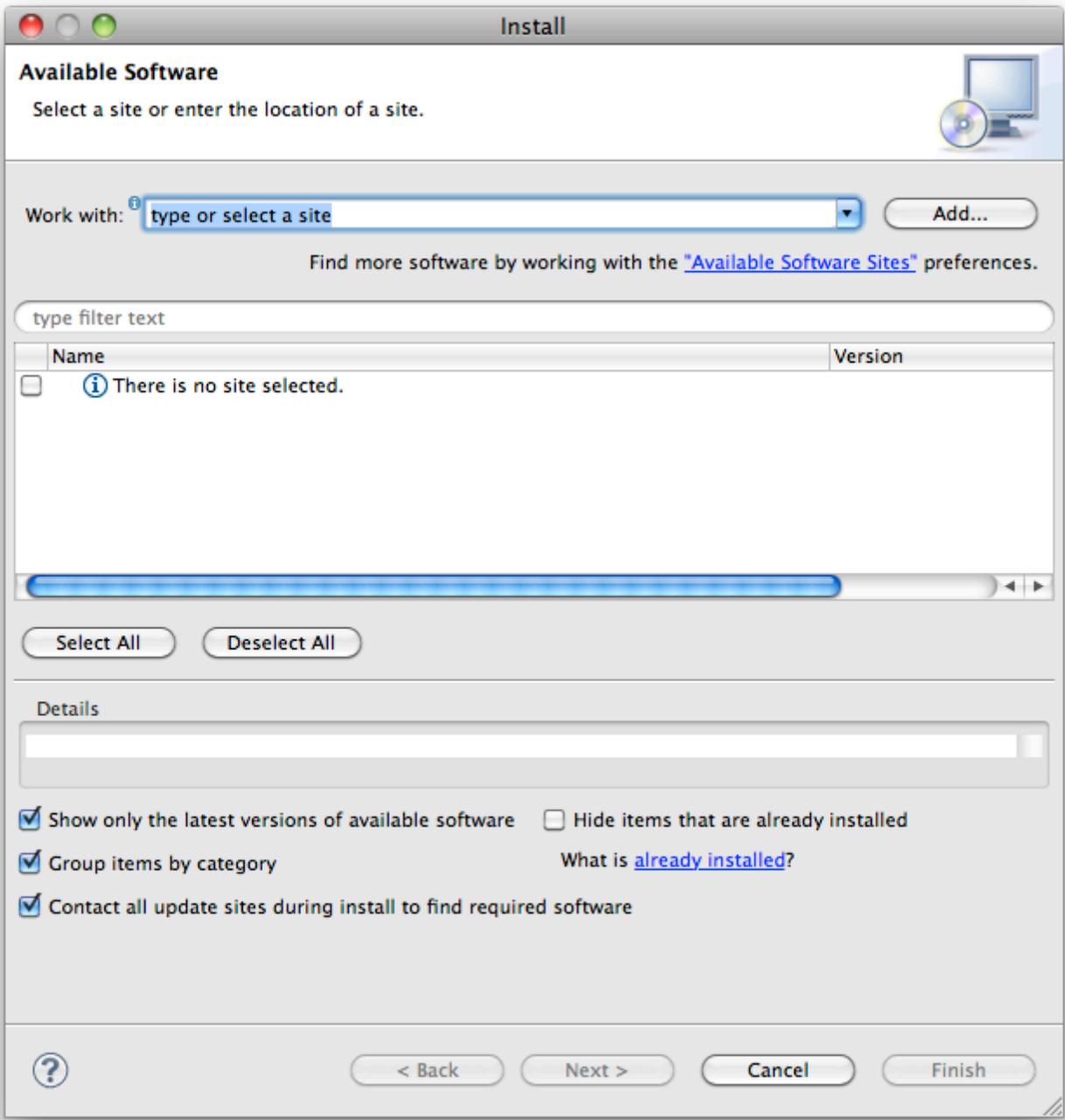
1. Install WOLips Plugin

Currently you have a stock Eclipse installation which will allow you to do normal Java development, but it does not contain any of the WOLips plugins, so you will not have any of the WebObjects development tools. To begin your WOLips installation, from within Eclipse, select **Help => Install New Software...**

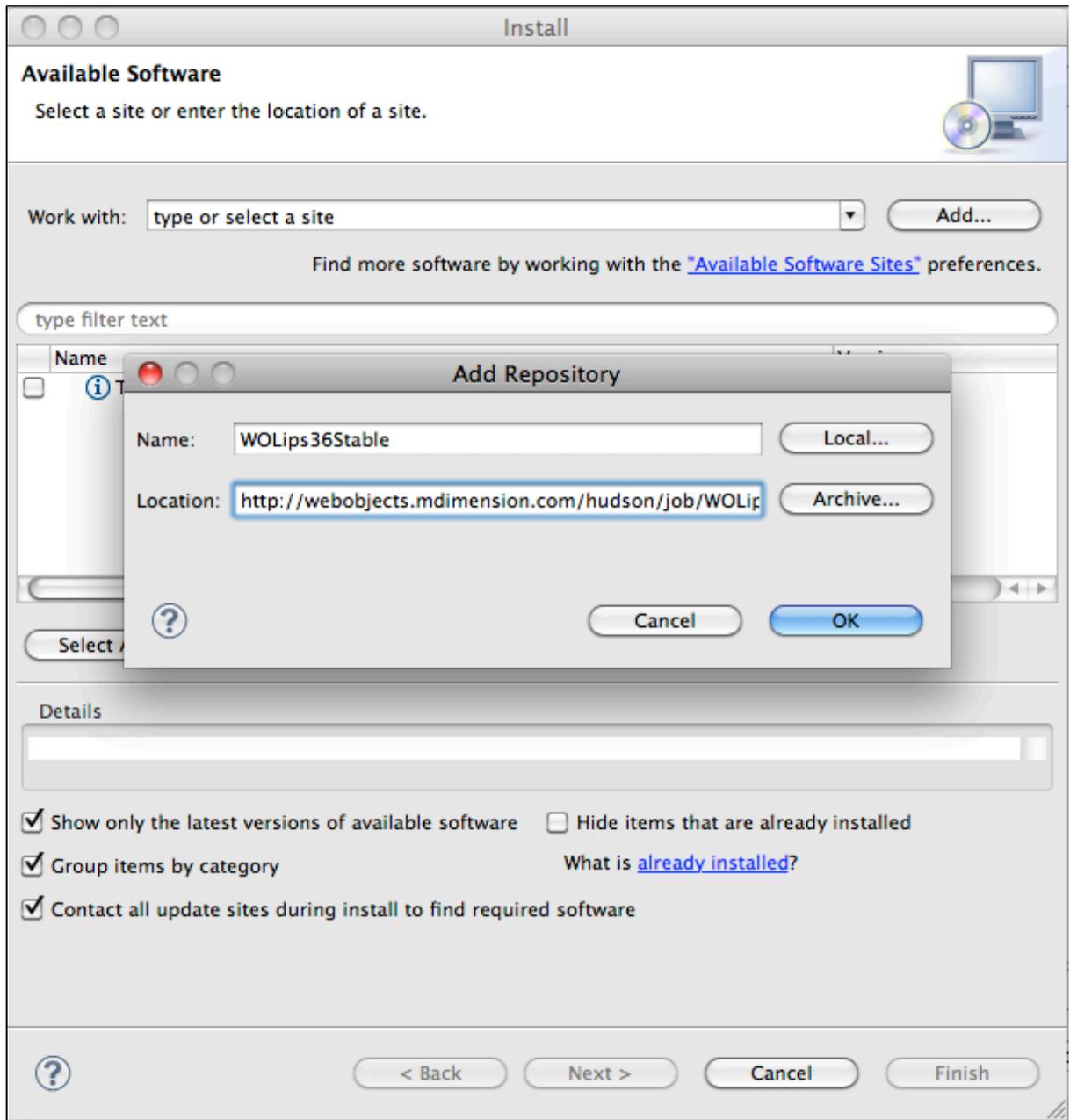


This opens the "Install" panel where you can specify sites for Eclipse to download new software features. This dialog allows you to add/inspect

any software features you have downloaded or were included in your default install of Eclipse. You can click "Available Software Sites" to see what was included by default.



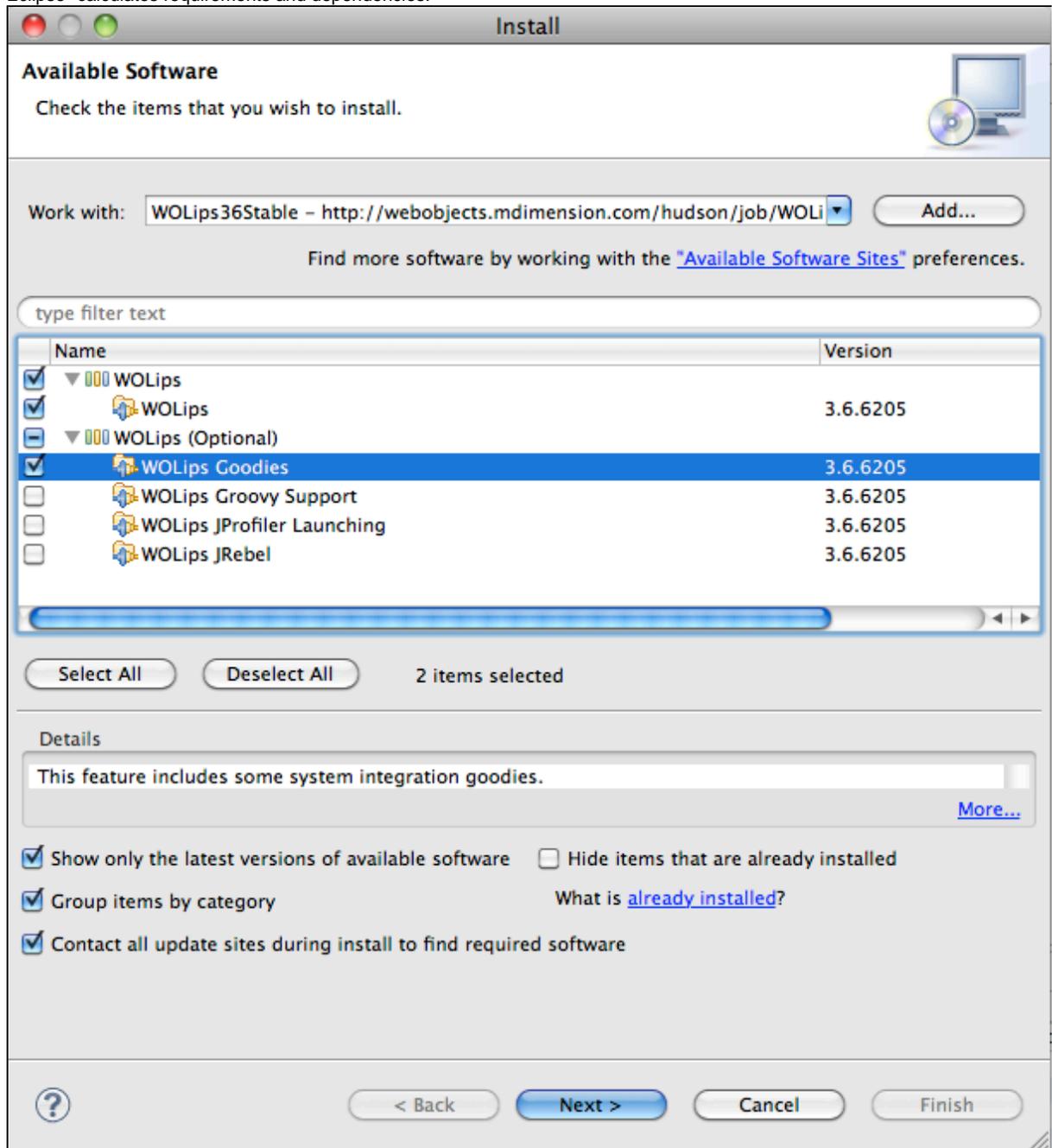
1. To install WOLips you will need to add the WOLips update site to Eclipse. You do this by clicking the "Add" button. In the Name field you can enter a name for your update site. The name is arbitrary but since you could have multiple similar update sites you should give it a name that leaves no doubt about what it is.



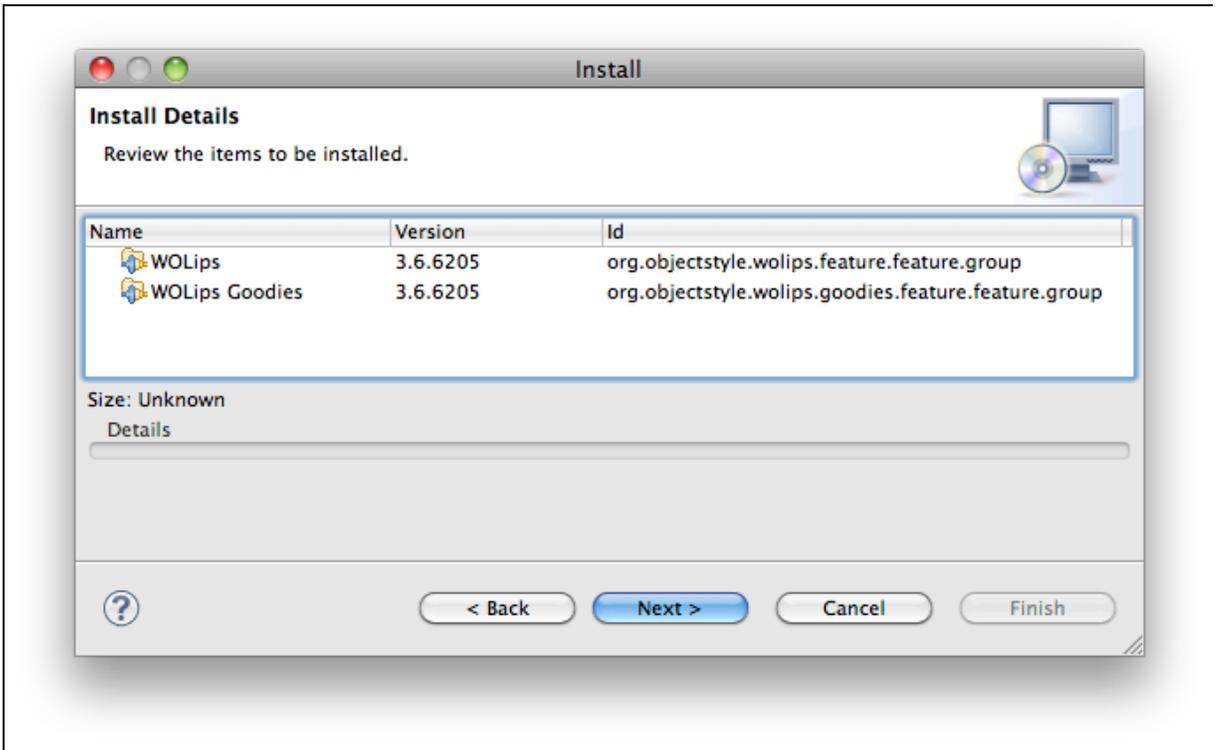
In the Location field, enter the following URL:

- **Current:** (For Eclipse 3.7) <http://jenkins.wocommunity.org/job/WOLips37Stable/lastSuccessfulBuild/artifact/temp/dist/>
2. You are returned to the "Install" panel, the WOLips site having been added to the list, and fully expanded to display all its offerings. Select features you want. If you're just getting started with WOLips you should select "WOLips" and "WOLips Goodies." Leave the other options until you are more familiar with Eclipse and WOLips and decide that you need them. (Here are

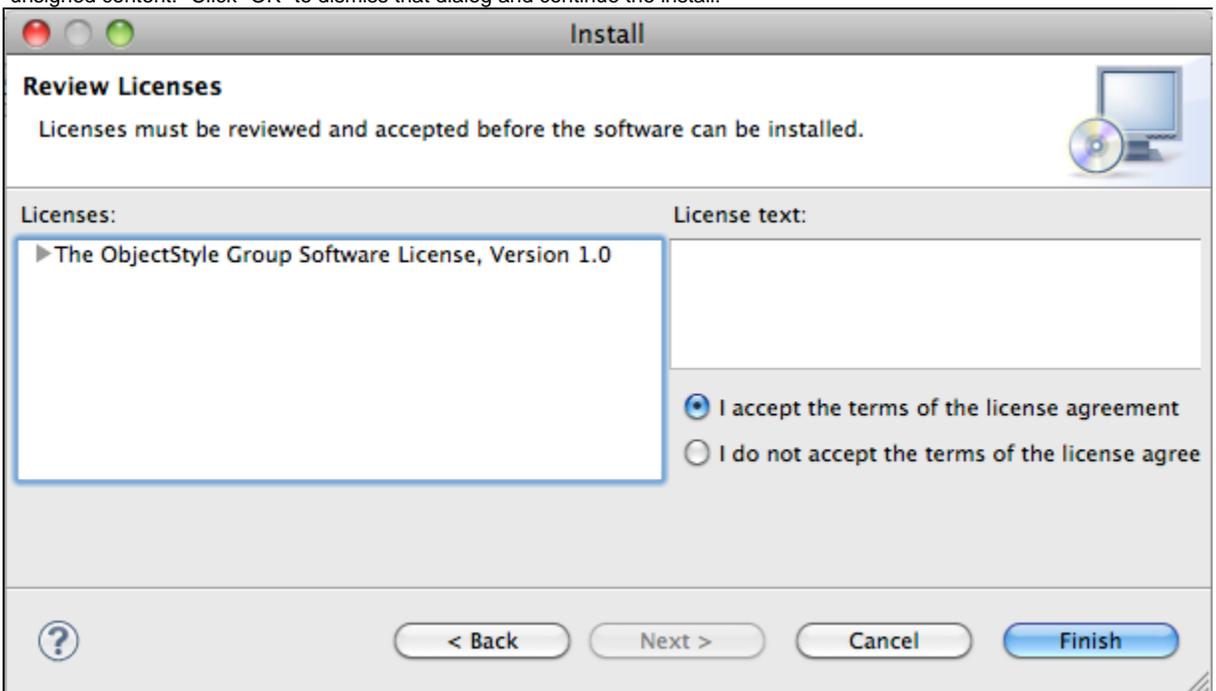
some Other WOLips Install Options.) After selecting the options you want, click "Next." You'll briefly see a progress bar while Eclipse "calculates requirements and dependencies."



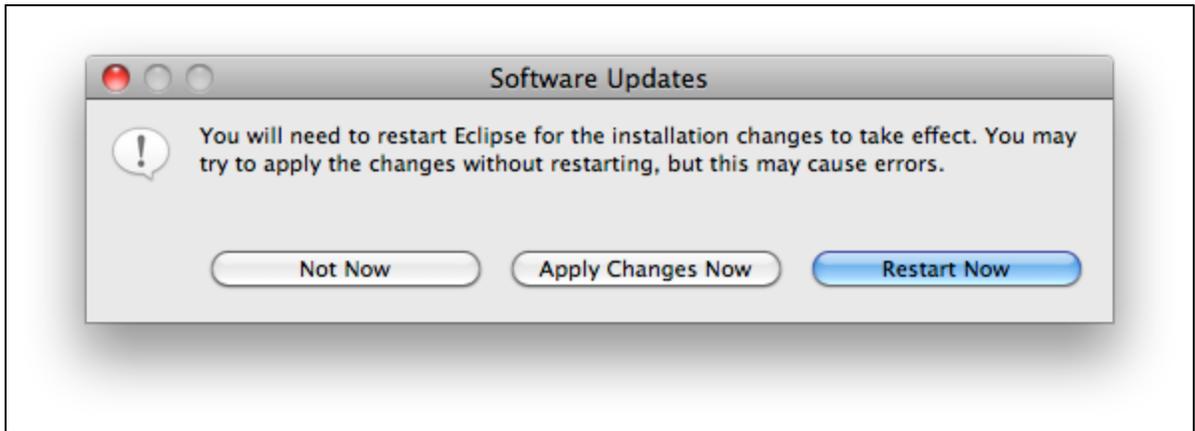
3. You'll see an "Install" panel, asking you to Review and Confirm selected items to be installed.



4. Click **Next >**. The next dialog contains the license agreements for the selected features. After reviewing the agreements (you DO review your license agreements, right?? 😊), select "I accept the terms of the license agreement." Selecting this radio button will accept the license agreements for **all** of the features listed. Click **Finish** to continue. You may also get a warning about "unsigned content." Click "OK" to dismiss that dialog and continue the install.



5. Eclipse will now download and install all the selected features and plugins. When it has completed, it will prompt you to restart Eclipse. Select "Restart Now."



2. Install the WebObjects Framework

Once the WOlips plugin is installed, you will need to install the core frameworks with the help of WOInstaller.jar, available from the wocommunity.org site. After, open a Command Prompt, and enter the following command:

```
java -jar \path\to\WOInstaller.jar 5.4.3 c:\Apple
```

You might get an error about a missing folder in C:\Apple\Library, you can ignore this error.

NOTE: Above WOInstaller.jar was built in 2008. Inside the jar file, the path statement for the dmg file is incorrect. Apple has moved this file. This jar file [WOInstaller2013_08_10.jar](#) has been updated for the correct path. Only tested on WinXP system. Don't know about other versions.

Step 4: Link the Framework to WOlips through the properties configuration file

Prerequisite: Eclipse with the WOlips plugin installed, and the WebObjects framework via step 2.

This is the most intricate step in the installation process. The configuration file found in the WOlips preferences must correctly reference the framework directory structure. Failure to successfully complete this step will result in broken libraries.

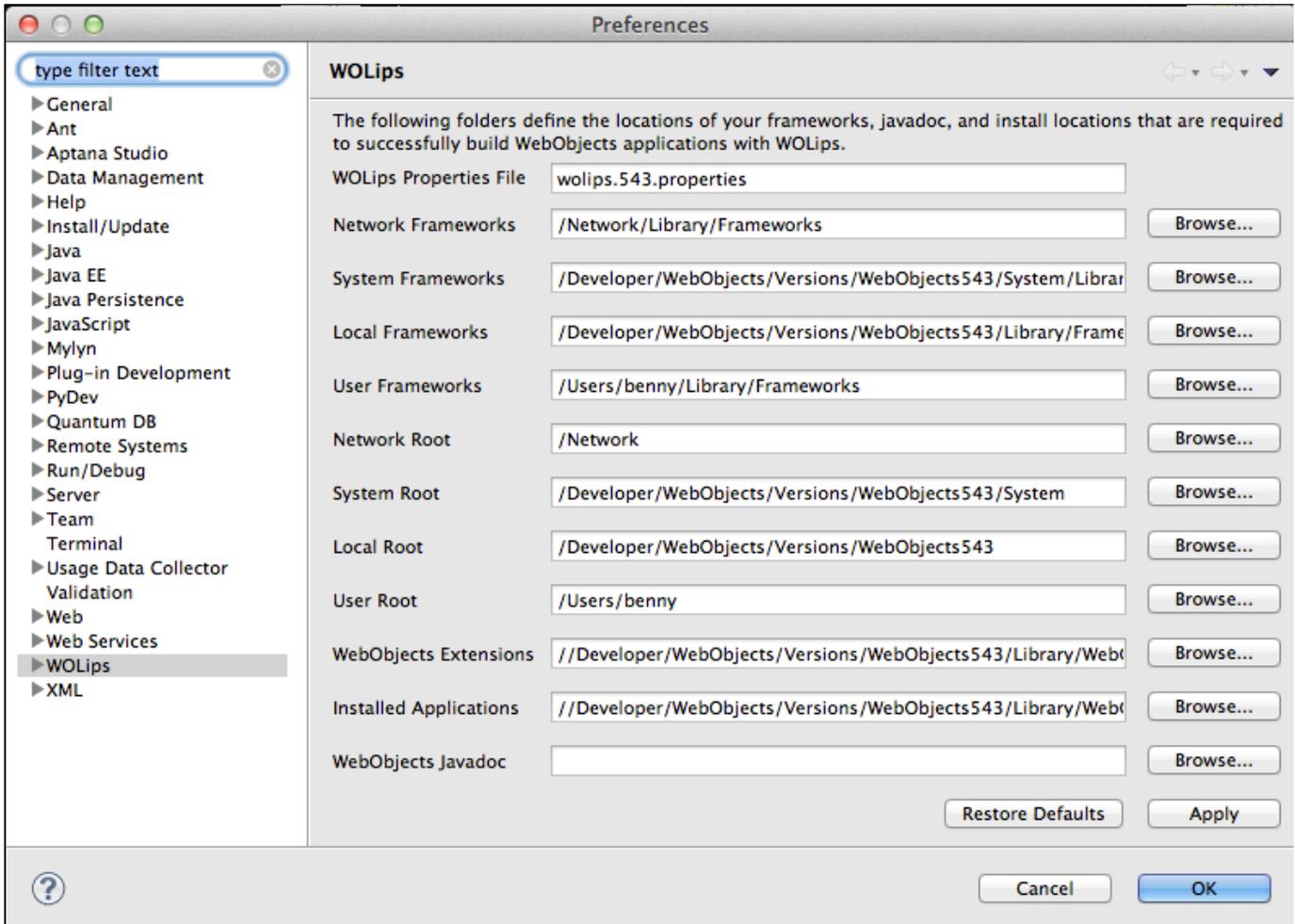
Running the script from step 2 installs the WO framework to the /Developer/WebObjects543 directory on your system. Below is the contents for a configuration file that matches this setup. Create a new file called wolips.543.properties. Copy the code below (after modification) into that file.

You must replace <YOUR_USER_NAME_HERE!> with your the username which installed the framework.

```
wo.system.root=/Developer/Versions/WebObjects543/System
wo.user.frameworks=/Users/<YOUR_USER_NAME_HERE!>/Library/Frameworks
wo.system.frameworks=/Developer/Versions/WebObjects543/System/Library/Frameworks
wo.bootstrapjar=/Developer/Versions/WebObjects543/System/Library/WebObjects/JavaApplications/wotaskd.woa/WOBootstrap.jar
wo.network.frameworks=/Network/Library/Frameworks
wo.api.root=/Developer/ADC%20Reference%20Library/documentation/WebObjects/Reference/API
wo.network.root=/Network
wo.extensions=/Developer/Versions/WebObjects543/Library/WebObjects/Extensions
wo.user.root=/Users/<YOUR_USER_NAME_HERE!>
wo.local.frameworks=/Developer/Versions/WebObjects543/Library/Frameworks
wo.dir.local.library.frameworks=/Developer/Versions/WebObjects543/Library/Frameworks
wo.apps.root=/Developer/Versions/WebObjects543/Library/WebObjects/Applications
wo.local.root=/Developer/Versions/WebObjects543
wo.dir.user.home.library.frameworks=/Users/<YOUR_USER_NAME_HERE!>/Library/Frameworks
```

Place this file in ~/Library/Application Support/WOLips, this directory should already exist and contain the default wolips.properties file.

The second part of this step is to link the WOLips plugin with the WO framework installed on your system. This is done through the tab found at Eclipse menu -> Preferences -> WOLips.



It is important to realize that if you change the WOLips Properties File to "wolips.543.properties" and then hit "apply" it will write out the (now stale) field values to that file. This could potentially clobber your new wolips.543.properties file with the default values. If you suspect this has happened, close the WOLips preferences and inspect the wolips.543.properties file.

Change the "WOLips Properties File" to wolips.543.properties and click OK.

At this point, you should have a functioning WebObjects platform to work with, visit [hello world link](#) to test your first WebObjects application.

Video of Step 3:

Your browser does not support the HTML5 video element

Step 5: Import Project Wonder into your Eclipse workspace

Please follow this guide: [Getting the Wonder Source Code](#), which includes the page for [importing the Project Wonder source](#) into your Eclipse workspace.

Project Wonder is a significant part of the WebObjects experience. Where the previous step seems to give new users the most grief now, this step will give you the most pleasure when working with WebObjects.

Video of Step 4:

Your browser does not support the HTML5 video element

At the end of this step, you will have Project Wonder imported into Eclipse under its own working set. If you are unfamiliar with working sets in

Eclipse, the following video will help clarify. This video also demonstrates a "Hello World" project creation to validate proper installation.

Your browser does not support the HTML5 video element

Fixing Auto Launch

Q. Hi everyone. I've seen several people on the list with the same problem as this, but haven't seen it resolved yet. When launching an app on WO5.2.2 developer on Windows XP, this error message is displayed:

```
Your application is not running on a supported development platform. AutoLaunch will not work.
```

I can cut and paste the URL into a browser manually, but I'm too lazy to do all that 'work' all day long 😊 I've tried the WOAutoOpenInBrowser setting with no luck. I've also tried setting W2K compatibility mode for project builder and the WOOpenUrl application with no luck.

A. use the following methods in your Application class:

```
/**
 * Calls _isAdditionalForeignSupportedDevelopmentPlatform
 *
 * @see
 com.webobjects.appserver.WOApplication#_isForeignSupportedDevelopmentPlatform()
 */
public boolean _isForeignSupportedDevelopmentPlatform()
{
    return (super._isForeignSupportedDevelopmentPlatform() ||
        _isAdditionalForeignSupportedDevelopmentPlatform());
}

/**
 * Check for Windows XP
 * @return true if runs on XP
 */
public boolean _isAdditionalForeignSupportedDevelopmentPlatform()
{
    String s = System.getProperty("os.name");
    return ( s != null && s.equals("Windows XP") );
}
```

- StefanKlein

Don't Use White Spaces for Path

Q. Can someone explain why NSBundle seems have broken path on Windows, but work fine on other platforms?

Example of an exception:

```
java.lang.IllegalArgumentException: Attempt to insert null into an
com.webobjects.foundation.NSMutableArray.
at com.webobjects.foundation.NSMutableArray.addObject(NSMutableArray.java:239)
at com.webobjects.eoaccess.EOModelGroup.modelGroupForLoadedBundles(EOModelGroup.java:700)
at com.webobjects.eoaccess.EOModelGroup.globalModelGroup(EOModelGroup.java:306)
at com.webobjects.eoaccess.EOModelGroup.defaultGroup(EOModelGroup.java:333)
```

A. When using white spaces in the project, framework path or WAR deployment etc on Windows, the white space will turn into "%20" in the path. This will cause NSBundle mainBundle returns null when it unable to load any bundles. For example:

Example path will not work:

C:\Documents and Settings

will turn into this wrong path:

C:\Documents%20and%20Settings

Therefore, this path will work:

C:\DocumentsAndSettings

Keep your files in path without white space seemed to be the first rule of thumb when development on Windows. Default location "C:\Documents and Setting" should not be used.