

# ICW-IDE

## Installation Manual

| Version: 1.0 | Status: Approved  
Security Level Public



# Imprint

InterComponentWare AG Industriestraße 4169190 Walldorf, Germany Tel.: +49 (0) 6227 385 0 Fax: +49 (0) 6227 385 199

© Copyright 2009 InterComponentWare AG. All rights reserved.  
LifeSensor is a trademark of the InterComponentWare AG.

Document Version: 2.1  
Document Language: en (US)  
Security Level: Public  
Product Name: eHF  
Product Release: 2.7  
Last Change: 07.09.2009  
Editorial Staff: S. Geier

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>5</b>
1.1	Document Overview .....	5
1.2	Document Conventions .....	6
<b>2</b>	<b>Installation Requirements .....</b>	<b>7</b>
2.1	General Requirements .....	7
2.1.1	Minimum Requirements.....	7
2.1.2	Recommended Requirements.....	7
<b>3</b>	<b>Installing the ICW-IDE.....</b>	<b>8</b>
3.1	Installation Files.....	8
3.1.1	About the ICW-IDE Installer .....	8
3.1.2	About the Plugin Sites .....	8
3.2	Preparing for your Development Environment Setup.....	9
3.2.1	Choosing which Components to Install .....	9
3.2.2	Uninstall Previous ICW-IDE Versions .....	9
3.3	Installing the ICW-IDE .....	11
<b>4</b>	<b>Installing the Update Site Plug-ins.....</b>	<b>22</b>
<b>5</b>	<b>Migrating existing projects into the ICW-IDE.....</b>	<b>32</b>
<b>6</b>	<b>Working with the ICW-IDE.....</b>	<b>34</b>
<b>A</b>	<b>Troubleshooting.....</b>	<b>35</b>
A 1	Checking the Install Logs .....	35
A 2	Uninstalling the ICW-IDE.....	35
A 3	Increasing HSQL database memory .....	35
<b>B</b>	<b>Checking Environment Settings.....</b>	<b>37</b>
B 1	Checking the JAVA Environment .....	37
B 1.1.1	Check JAVA is correctly installed.....	37
B 1.1.2	Removing older versions of Java .....	38
B 1.1.3	Removing obsolete Java Environment Settings.....	39
B 1.2	Check Maven Environment .....	39
B 1.2.1	Check MAVEN 1 is correctly installed .....	40
B 1.2.2	Removing obsolete Maven Environment Settings .....	41
B 1.2.3	Configuring the Maven build.properties file .....	41

<b>C</b>	<b>References.....</b>	<b>43</b>
C 1.1	Maven .....	43
C 1.2	Eclipse .....	43
C 1.3	Subversion.....	43
C 1.4	Jira .....	43
<b>D</b>	<b>List of Current Eclipse Plug-Ins .....</b>	<b>44</b>
D 1	Core Plugin Updates .....	45
D 2	Optional Plugin Updates.....	45
<b>E</b>	<b>List of Figures .....</b>	<b>51</b>

# 1 Introduction

## 1.1 Document Overview

This Installation Manual provides you with the necessary steps for setting up the ICW-IDE.

The ICW-IDE is an Integrated Development Environment based on Eclipse and customized for working with ICW applications. It contains versions of

- Sun Java SE 6
- Apache HTTP Web Server 2.2
- Apache Ant 1.7
- Apache Tomcat 6.0
- Apache Maven 1.1
- Apache Maven 2.2
- Hypersonic SQL Database

In this document the user will be guided through the process of installing the ICW-IDE.

## 1.2 Document Conventions






Mark up	Explanation	Example
italic	Windows, Dialogs, Sections, Forms	Open the window:: <i>Options</i>
bold	Buttons, Fields, Tabs, Check boxes, Options, Drop-down-Menus	The Button <b>next</b> lets you navigate through the wizard.
bold and underlined	Link	Click on <a href="http://www.java.com">http://www.java.com</a> . The website will be opened.
bold and italic	Emphasis	You will find these information <b><i>only</i></b> on the vendors page.
	Caution	<b>Warning:</b> Save your inputs!
	Note	<b>Note:</b> This view shows only data according to the installation process.
	Example	<b>Example:</b>
	Path	<b>Path:</b> Click File > save as
	Code	<b>Code:</b> <code>chmod -R g+wx /home/esb</code> <code># chgrp -R icw /home/esb</code>
<<Variable>>	Variable	The installation directory will be shown << INSTALL DIRECTORY>>
[strg+alt]	Shortcut	

Table 1: Document Conventions

## 2 Installation Requirements

### 2.1 General Requirements

In order to be able to fully utilize all the functions of the ICW-IDE you should already have experience with the Eclipse IDE and development in a Java programming environment using the Maven build tool.

You can find help how to use Eclipse on their homepage: <http://www.eclipse.org/>.

#### 2.1.1 Minimum Requirements

The minimum requirements for running the ICW-IDE are

- 500 MB Hard Drive space
- 1 GB RAM
- Windows XP SP2

#### 2.1.2 Recommended Requirements

The recommended requirements for developing ICW applications depend on the application itself. Please consult the application's reference documentation. In general the following is recommended:

- > 1 GB Hard Drive space
- > 2 GBs RAM
- Windows XP SP2 or later service pack.

The ICW-IDE will also run on Microsoft Vista.

## 3 Installing the ICW-IDE

### 3.1 Installation Files

The following files are delivered for the ICW-IDE installation process

- `ICW-IDE-3.0.6.exe`
- `Update_Site(core).zip`
- `Update_Site(optional).zip`

#### 3.1.1 About the ICW-IDE Installer

The ICW-IDE is based on the Eclipse software development platform. The `ICW-IDE-3.0.6.exe` contains:

- Eclipse 3.4 SR2 (“Ganymede”)
- Sun Java SE 1.6
- *Apache HTTP Webserver 2.2.8*
- *Apache Tomcat Servlet Container 6.0.20*
- *Apache Maven 1.1* and the *Maven Runner* plugin.
- *Apache Maven 2.2*
- *Apache Ant 1.7.1*
- Hypersonic SQL database

#### 3.1.2 About the Plugin Sites

Eclipse employs a plug-in mechanism in order to extend the core functionality of their development platform. The version of Eclipse bundled with the ICW-IDE is already customized for Java development and includes support for CVS, Mylyn, and XML Editing.

Provided in addition to the ICW-IDE are two Update Sites containing plug-in updates for working with ICW development.

- The `Update_Site(core).zip` contains the plug-ins required for basic development
- The `Update_Site(optional).zip` contains plug-ins which developers may find useful.

A complete list of the plugins is presented in Appendix D.

## 3.2 Preparing for your Development Environment Setup

### 3.2.1 Choosing which Components to Install

As already mentioned, the ICW-IDE comes with the following additional components:

- *Sun Java SE 1.6*
- *Apache HTTP Webserver 2.2.8*
- *Apache Tomcat 6.0.20*
- *Apache Maven 1.1 and Apache Maven 2.2*
- *Eclipse*

If you already have any of these components installed on your machine, you may decide to omit these during the installation process of the ICW-IDE. Please be sure that the versions of the software you currently have installed are the correct compatible versions required. (Refer to section 3.1.1)

If you wish to use a previously installed Maven 1 already on your system, make sure your MAVEN\_HOME path is correct. Otherwise, for Maven 2, make sure you have a correctly configured M2\_HOME. Both settings should also be available on the system path. If you wish to use a preconfigured environment, please read the Check Environment Settings Appendix (Appendix B) on how to correctly configure your environment before beginning the next section.

If you do not have any of these components installed on your machine, or you wish to install the latest components, you should select the default options during the installation process.

### 3.2.2 Uninstall Previous ICW-IDE Versions

If you already use an earlier version of the ICW-IDE it is recommended that you uninstall this version first before installing any newer version.

From the Start menu select

- Programs -> ICW-IDE -> Uninstall ICW IDE



**Warning:**

If you do not backup your current project workspace to a new location before uninstall all your project information may be deleted. It is recommended to always backup your workspace and later re-import your projects into the new ICW IDE Eclipse version.

---

### 3.3 Installing the ICW-IDE

To start the process, *double click* on the **ICW-IDE-3.0.6.exe** installer executable.

1. First, you'll be greeted with the start screen. Click *Next* to proceed to the 3<sup>rd</sup> party License Agreement screen of the ICW-IDE installation wizard.



Figure 1: Welcome Dialog

2. To accept the license, please make sure that the *checkbox* for the acceptance of the license agreement has been checked. Then click *Next*.

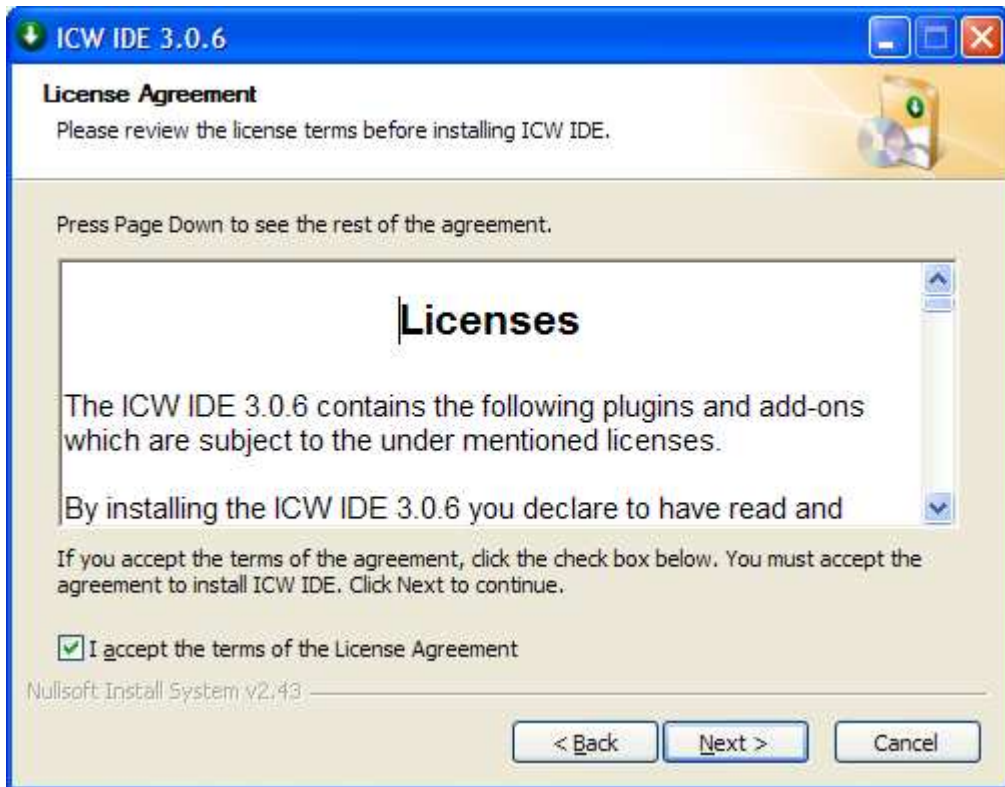


Figure 2: License Agreement Dialog



**Note:**

The License Agreement contains all the license agreements for the 3<sup>rd</sup> Party components bundled with the ICW-IDE. Please be sure to read the agreement before accepting.

3. Choose the components you wish to install, and then click next. The components can be chosen by Install Type or custom selected.
  - eHF Components: Recommended components for development with eHF.
  - IPF Components: Recommended components for development with IPF

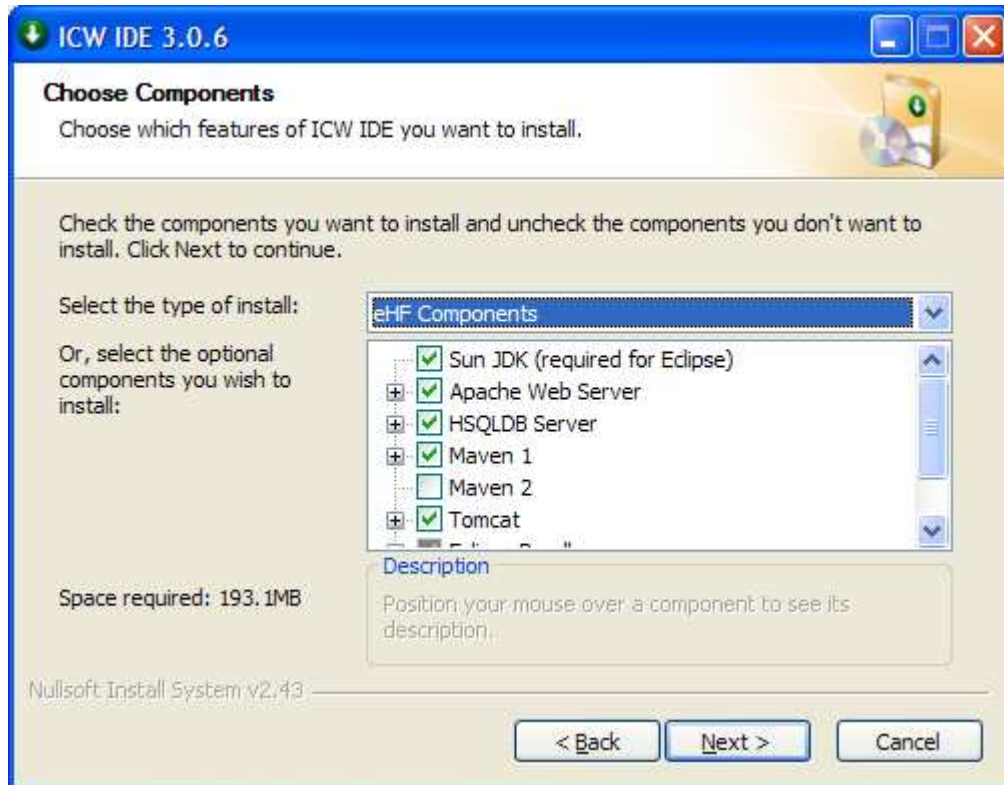
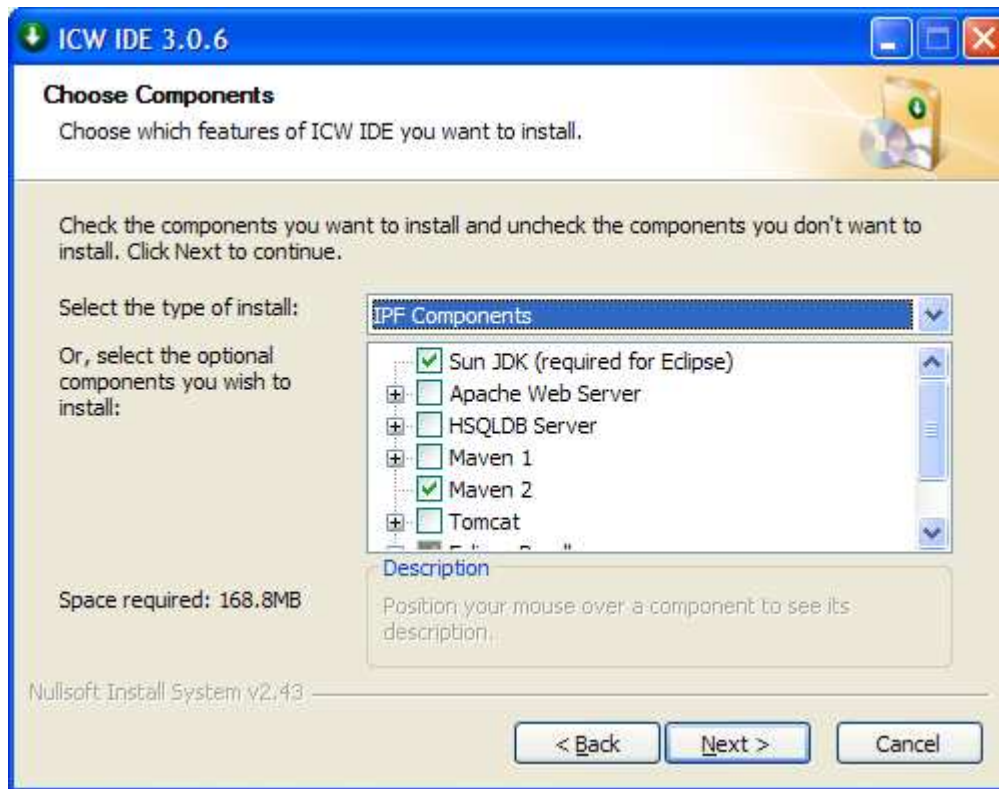


Figure 3: Selection of Components to install – eHF Components



**Figure 4: Selection of Components to install – IPF Components**



**Note:**

The main difference between EHF and IPF development is the usage of different Maven version as it's build tool. It is possible to install both versions of Maven in the same environment.

### **About the Components:**

The *Sun JDK* is required to run the Eclipse development environment. Even if you select this component your environment will be checked for compatible Java installation, and will not be installed if a correct version is detected. It is recommended that this option is always selected.

The *Apache Web Server* is used as an SSL endpoint and load balancer for web application hosted by Tomcat. You may choose to deselect this if you only want to develop or test with Tomcat.

The HSQLDB Server is a lightweight database used for development and testing.

The Maven components are used as build tools. If you decide to develop EHF applications then you should choose Maven 1. Otherwise, for IPF development, choose Maven 2. Both Maven versions can be selected and both can be used in the same environment. In the case, you already have Maven 1 or Maven 2 installed on your system you can ignore these options. In this case, you will need to ensure that the Maven environment is setup correctly. Please see the Troubleshooting Appendix for further information.

Tomcat is used to deploy and run a web application..

Finally, *the Eclipse Bundle* contains the Eclipse Development Platform. This is recommended.

4. Choose the installation folder for the infrastructure components. The ICW-IDE will be installed at this location. You may use the Browse button to locate a destination directory. Click Next to continue.

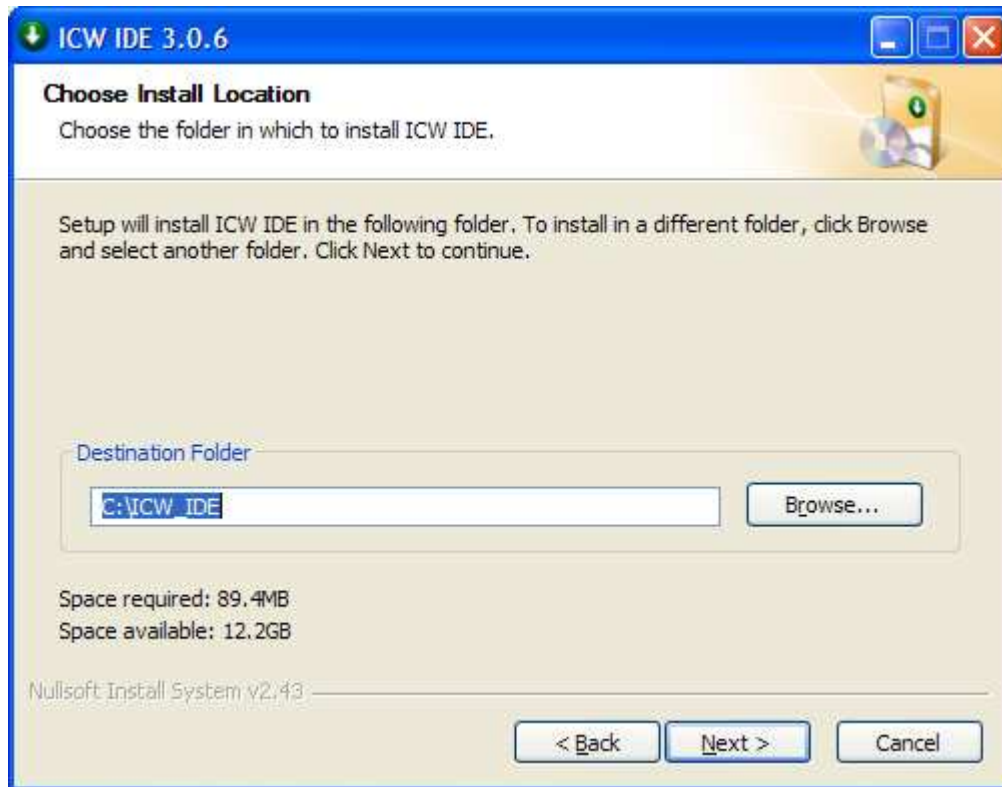
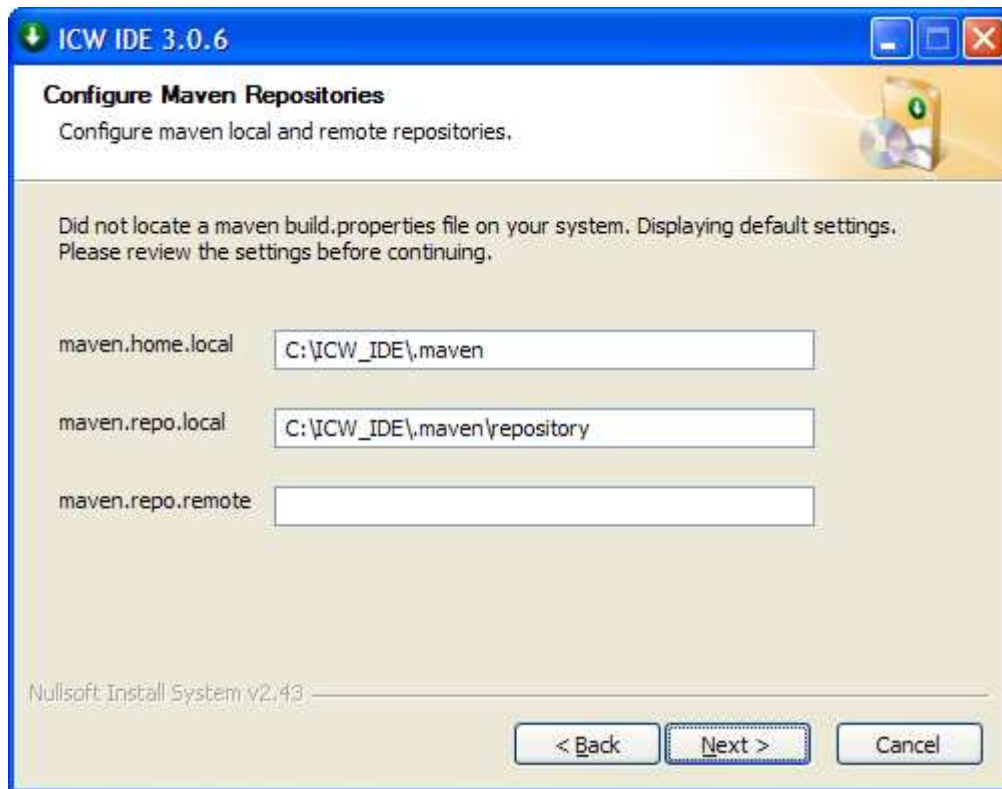


Figure 5: Choose the Installation Folder

5. If you choose to install Maven 1 as part of the ICW-IDE, you will be presented with the configuration screen for the Maven Repositories. Review the settings and then select Next.



**Figure 6: Configure Maven Repositories**

The ICW-IDE installer searches for existing Maven 1 repositories on your system. If you already have Maven 1 installed, you will be presented with the settings used. It is recommended you use existing Maven settings.

If you do not have Maven installed on your system, you will be presented with default options as shown in Figure 6: Configure Maven Repositories. You can change the settings by configuring:

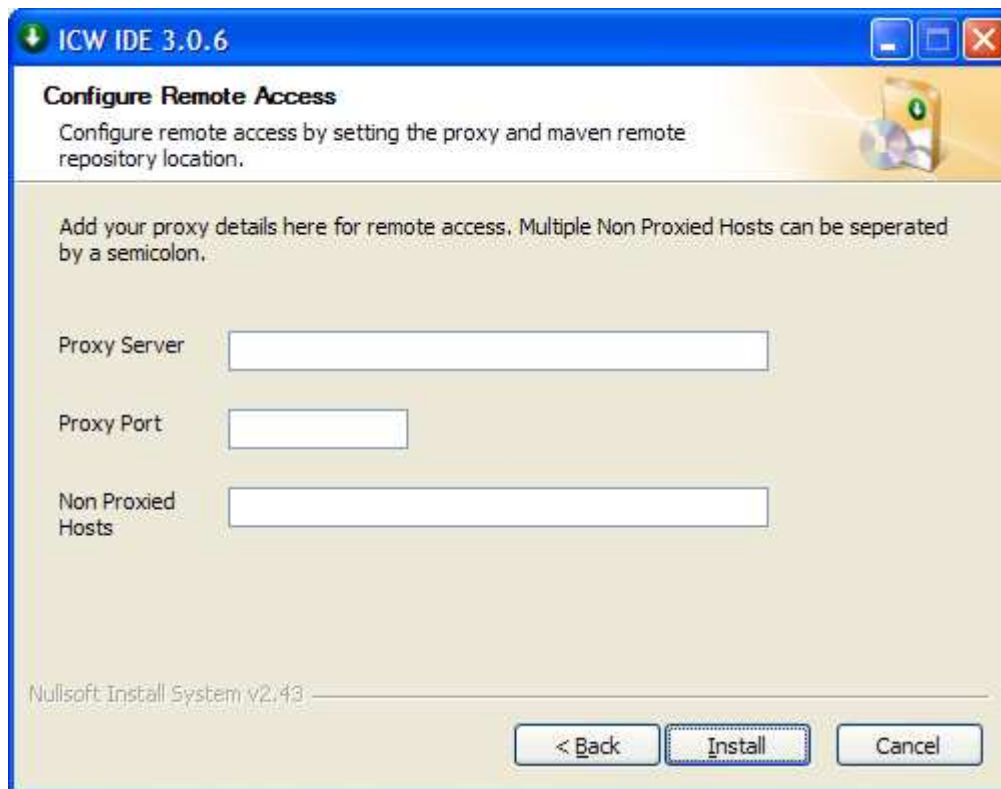
- `maven.home.local`: The location of your Maven cache. The Maven cache contains the plugin artifacts used by Maven.
- `maven.repo.local`: The location of your Maven repository. The Maven repository contains the artifact dependencies for your projects.
- `maven.repo.remote`: The remote Maven repository. This is a centrally located repository that can be shared between several developers. The artifacts are retrieved from this remote repository to your local repository.



**Note:**

No default value is presented for `maven.repo.remote`. Users do not need to configure a remote repository (`maven.repo.remote`) if they have access to copy of a repository. They can reuse this repository by copying it to the location configured by `maven.repo.local`.

6. Finally enter the settings for configuring Remote Access to the internet, and click Install to start the installation process.



**Figure 7: Configuring Maven for remote access**

Configuring Remote Access allows Eclipse to access sites accessible only via a proxy. You can choose not to enter any information, if you do not have a proxy. If you are not sure what to enter, please contact your IT administrator. You can also leave these fields empty and configure this later in Eclipse.

The following information can be entered:

- `Proxy Server` is the server address of the proxy needed to connect to the internet.
- `Proxy Port` is the port used by the proxy server to connect to the internet.

- Non Proxied Hosts are addresses that exclude proxy use when connecting to them. Multiple hosts can be separated using a semicolon.



**Note:**

Proxy Server, Proxy Port, and Non Proxied Hosts can be omitted and left blank if you are not using a proxy server.

Non Proxied Hosts include the current machine (i.e. localhost) as a default. You only need to enter other hosts that you intend to access on you network e.g. \*.mynetwork.com.

7. The Installation process now begins. All the components chosen earlier are installed.

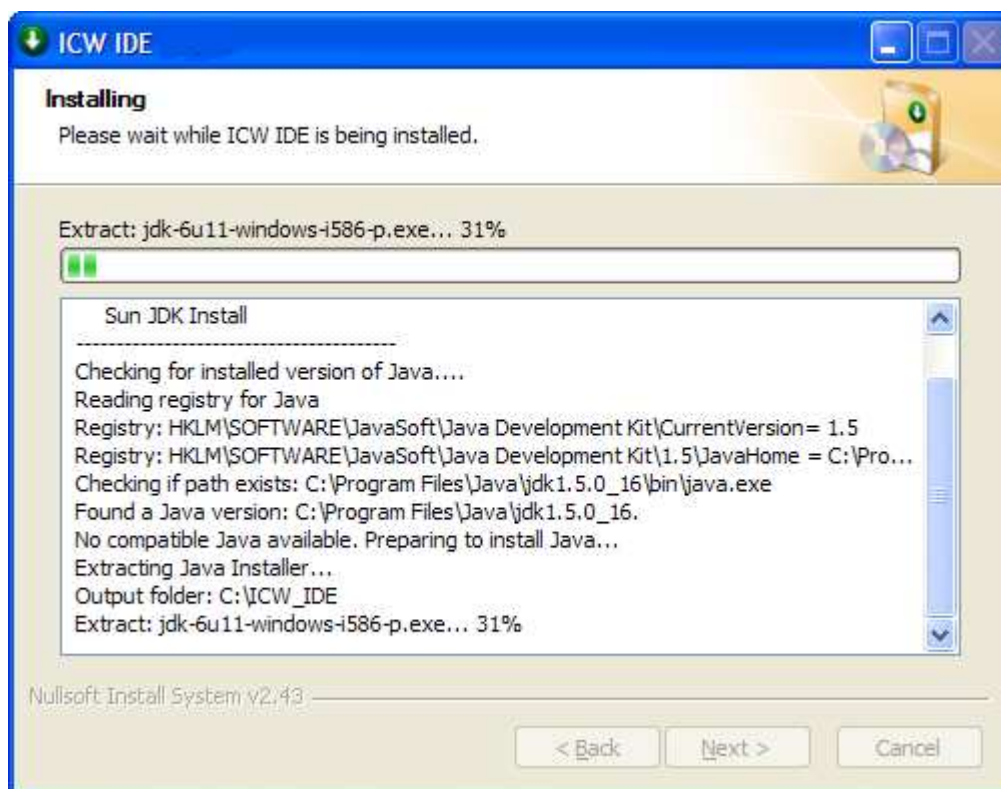


Figure 8: Installing the ICW-IDE...



**Note:**

As part of the installation process the ICW-IDE installer checks your environment for compatible Java versions.

---

8. If the IDE has been installed and configured successfully (instead of the message box mentioned before), you will see the following dialog:



**Figure 9: Successful install message**

---



**Warning:**

If you do not see the above dialog, then something went wrong during installation. Please see the Troubleshooting section in Appendix 0 for further details.

---

9. After completing the ICW-IDE installation, you will be prompted to reboot your computer. It is recommended you reboot your computer before starting any component installed by the ICE-IDE installer. You may choose to reboot later if wish to close any open application or save remaining work. Click Finish to end the installation process.

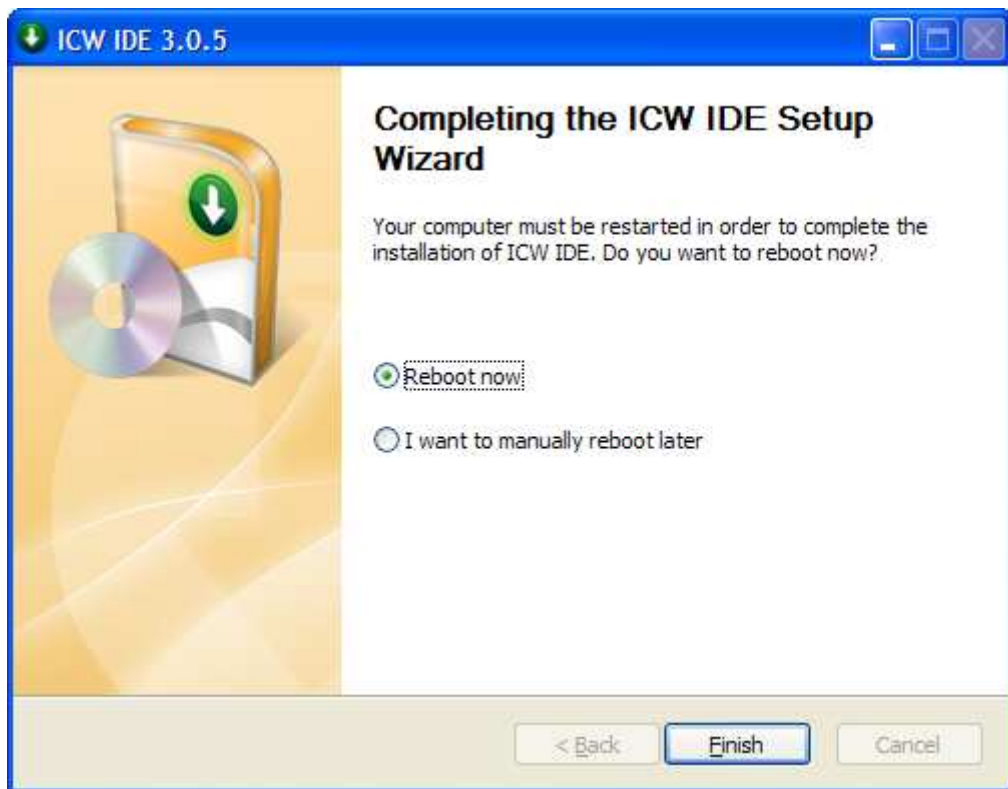


Figure 10: Installation finished

After the installation has finished, you have the option to reboot your machine. A reboot is recommended to set up the environment variables which will be used by Eclipse. In order to do so, make sure that *Reboot now* is selected and afterwards click on *Finish*.



**Note:**

You can find the installed ICW-IDE components in your program menu by selecting *Start->programs->ICW-IDE*.

After finishing, continue to the next section (Installing the Update Site Plug-ins) to configure Eclipse with the required plug-ins.

## 4 Installing the Update Site Plug-ins

There are two separate Update Site plug-ins for ICW\_IDE-3.0.4:

- `Update_Site(core).zip`
- `Update_Site(optional).zip`

The `Update_Site(core).zip` contains the plug-ins required for basic development, while the `Update_Site(optional).zip` contains plug-ins which developers may find useful.



### Note:

It is recommended that all users of the ICW-IDE install at least the `Update_Site(core)` plug-ins. **For a complete list of what is contained in both Update Sites see Appendix D.**

---

The following steps explain the process of installing the `Update_Site(core)`. Follow the same steps a second time to add the `Update_Site(optional)` if you wish to install these extra plug-in.

1. Unzip the set of plug-ins you wish to install.

Windows contains an Extraction Wizard utility to uncompress zip files. To install the `Update_Site(core)`:

- Right Click on the `Update_Site(core).zip` select 'Extract All'
- The Extraction Wizard dialog is displayed. Select Next.

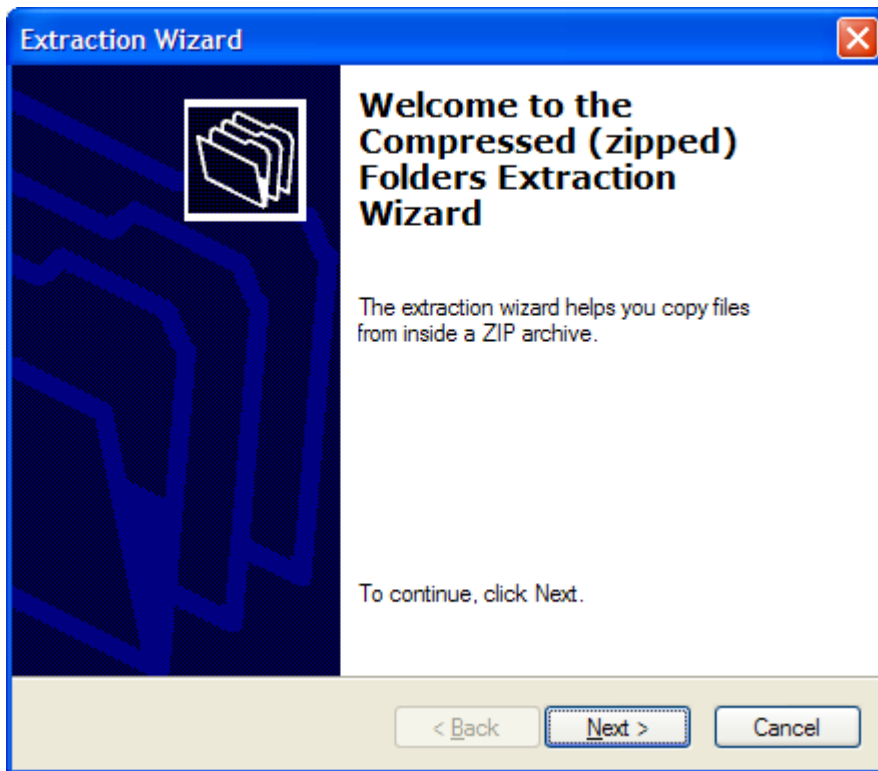


Figure 11: The Extraction Wizard

- Select the destination you wish to extract the Update Site too. You can use the Browse to select a location. Click Next.

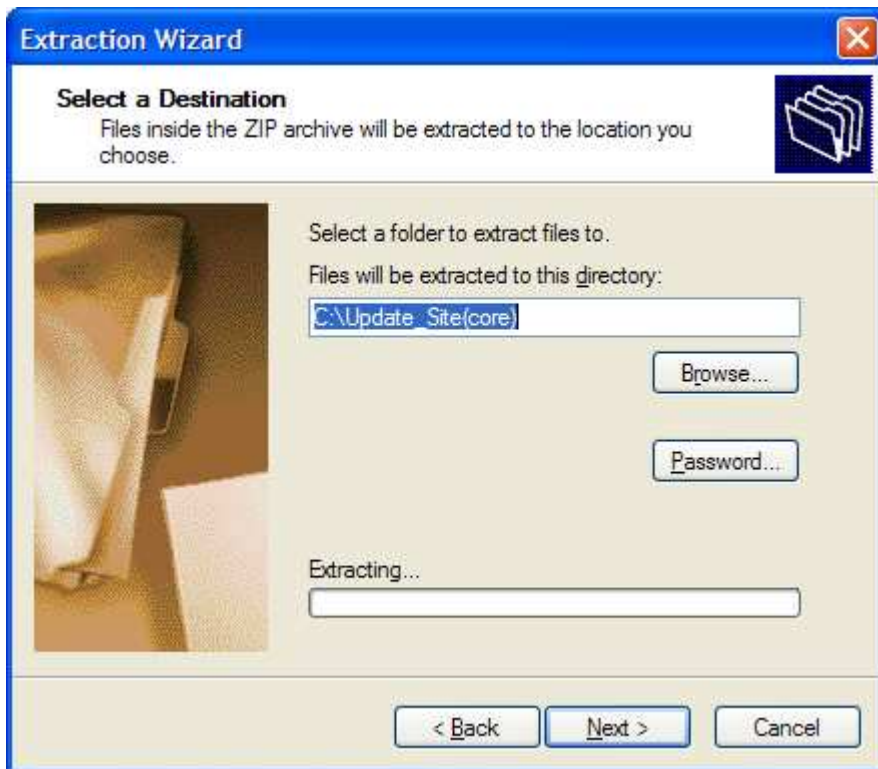


Figure 12: Select a Destination for the ZIP Archive

- Check the 'Show extracted files' checkbox and click Finish.

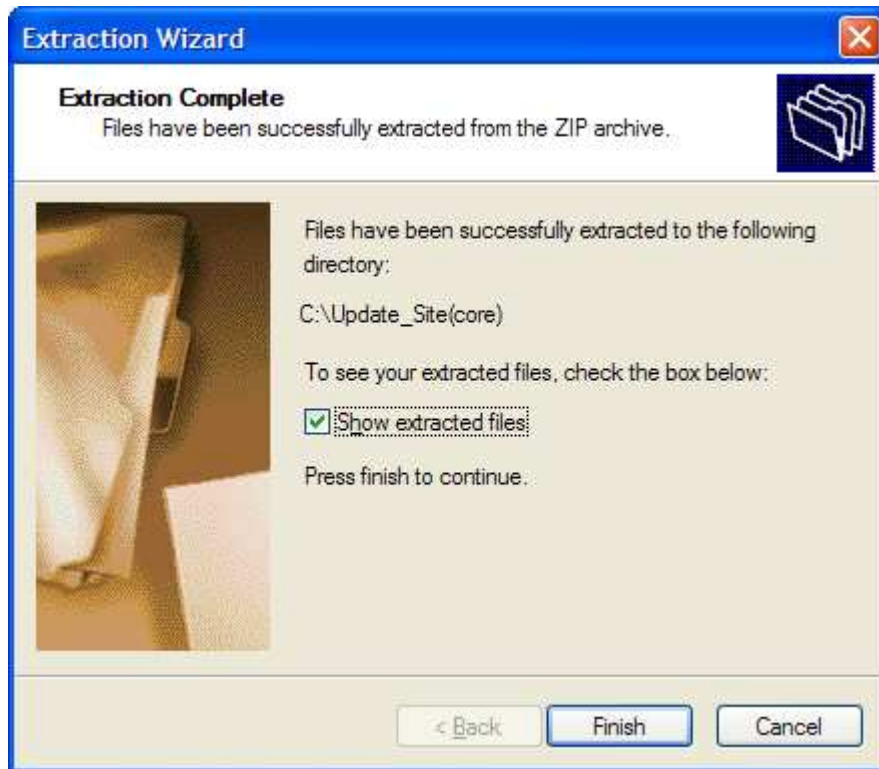


Figure 13: Extraction is complete

- Make a note of the path of the extract Update Site. You will need this later.

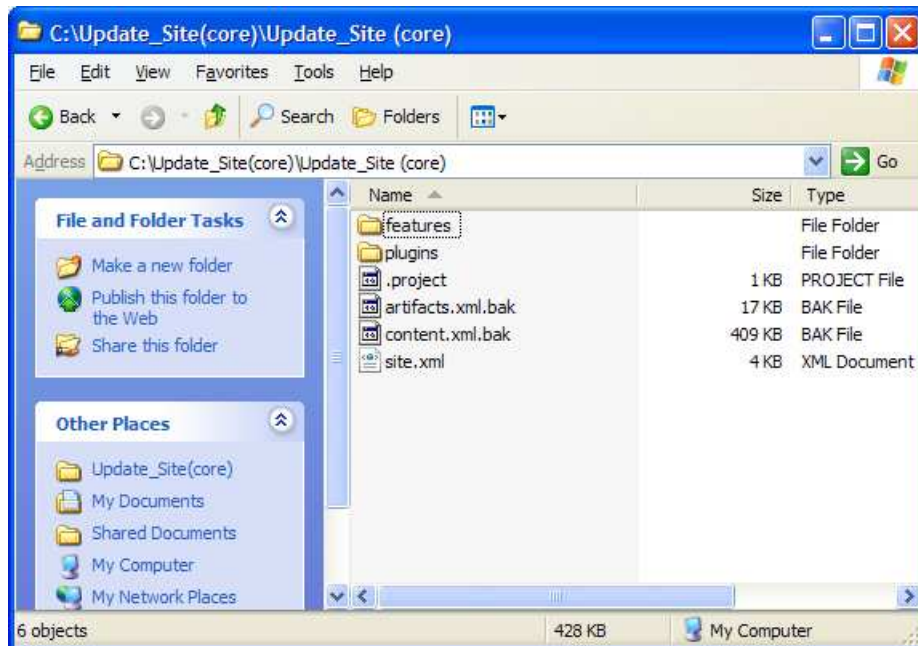


Figure 14: Path to the contents on the Update Site.



**Note:**

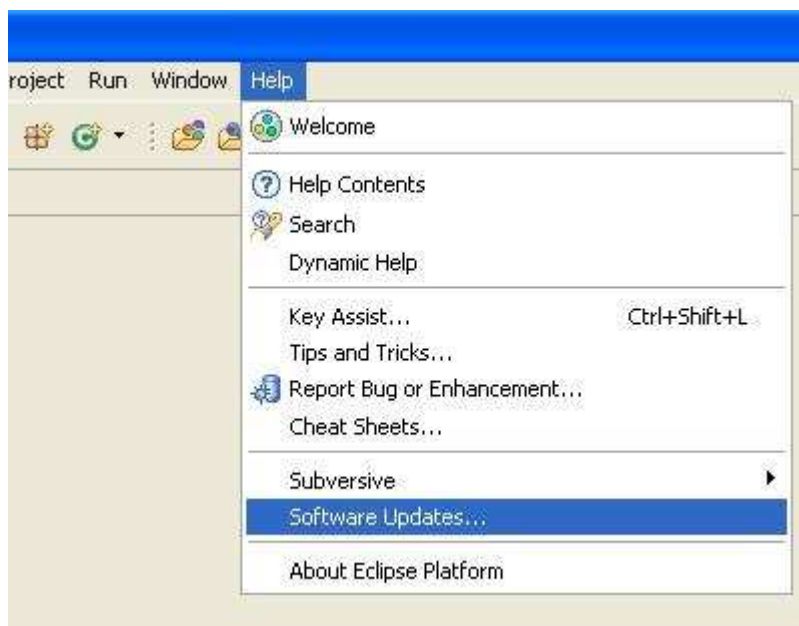
You should follow the same process to extract the `Update_Site(optional).zip`. Extract the contents of `Update_Site(optional).zip` to a separate directory from the `Update_Site(core)`.

2. Next start the Eclipse development platform.

There are two approaches to start Eclipse:

- Use the Desktop Icon label 'Eclipse'
- Open Start -> Programs -> ICW\_IDE -> Eclipse

3. Click on *Menu->Help* and choose *Software Updates*.



**Figure 15: The Software Update dialog in Eclipse**

4. Select the tab *Available Software* from the dialog window *Software Updates and Add-ons* opening up.

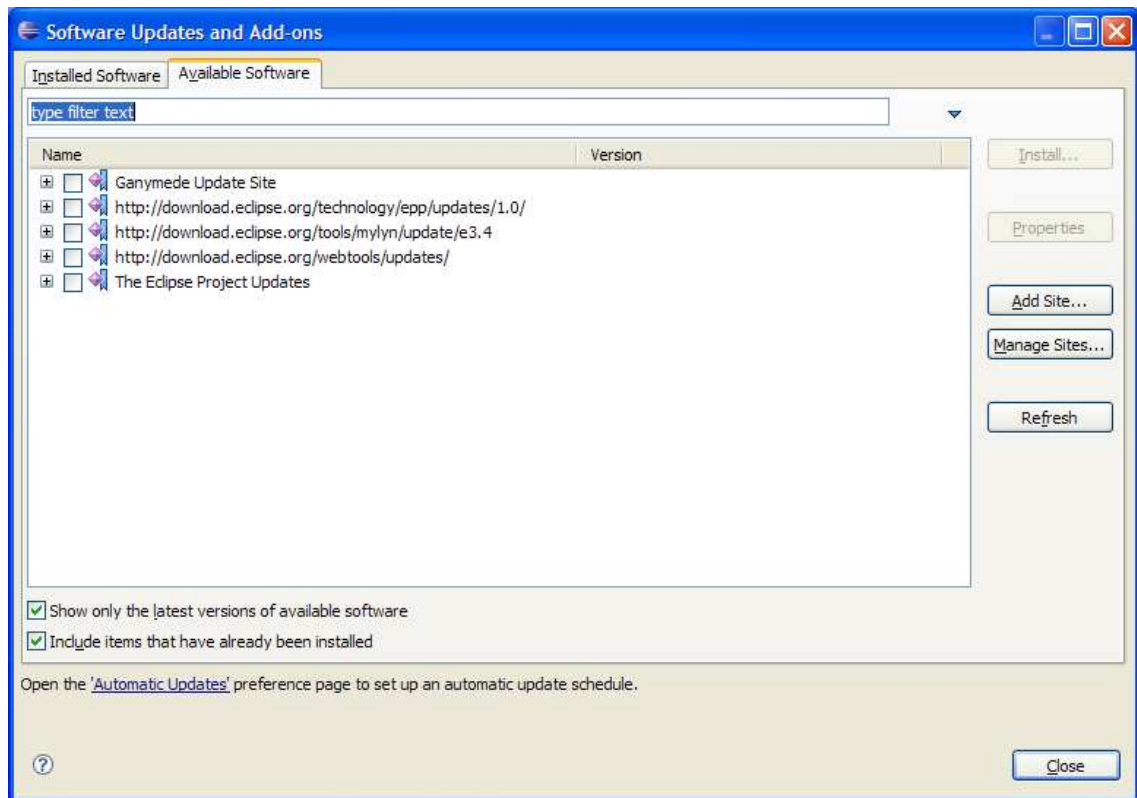


Figure 16: Software Updates and Add-ons

5. Click on the *Add Site* button. Then, click on the button *Local*.

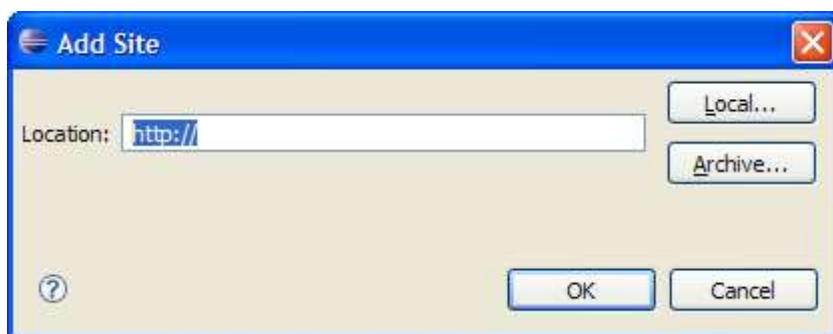


Figure 17: Choose *Local*

6. A Popup dialog is displayed, prompting you to browse for your folder. Select the Update\_Site you extracted in step 2.

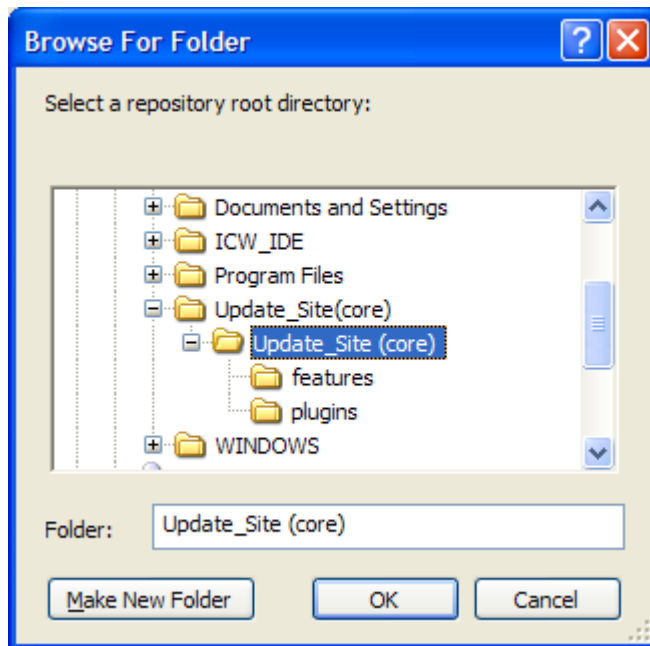


Figure 18: Browse For Folder – selecting the Update\_Site(core) folder

7. The *Add Site* dialog will appear with the selected folder. Once you are done, press the *Ok* button.

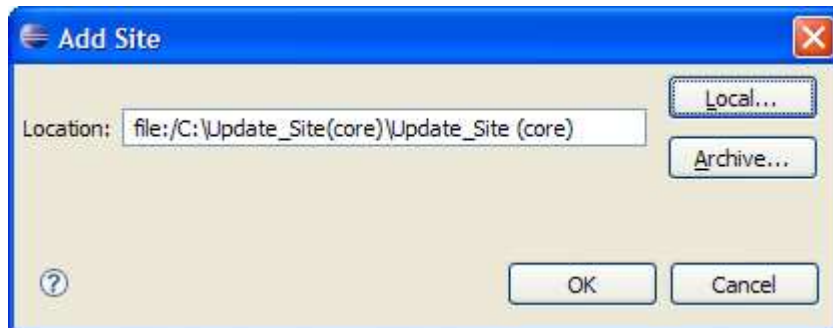
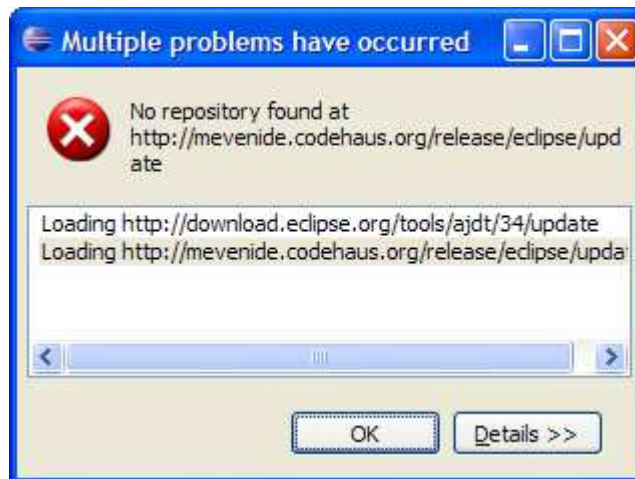


Figure 19: Add Site

When Eclipse adds an Update Site, it checks the contents and tries to access any third party URLs contained in the site. You may encounter something like the following pop up dialog:



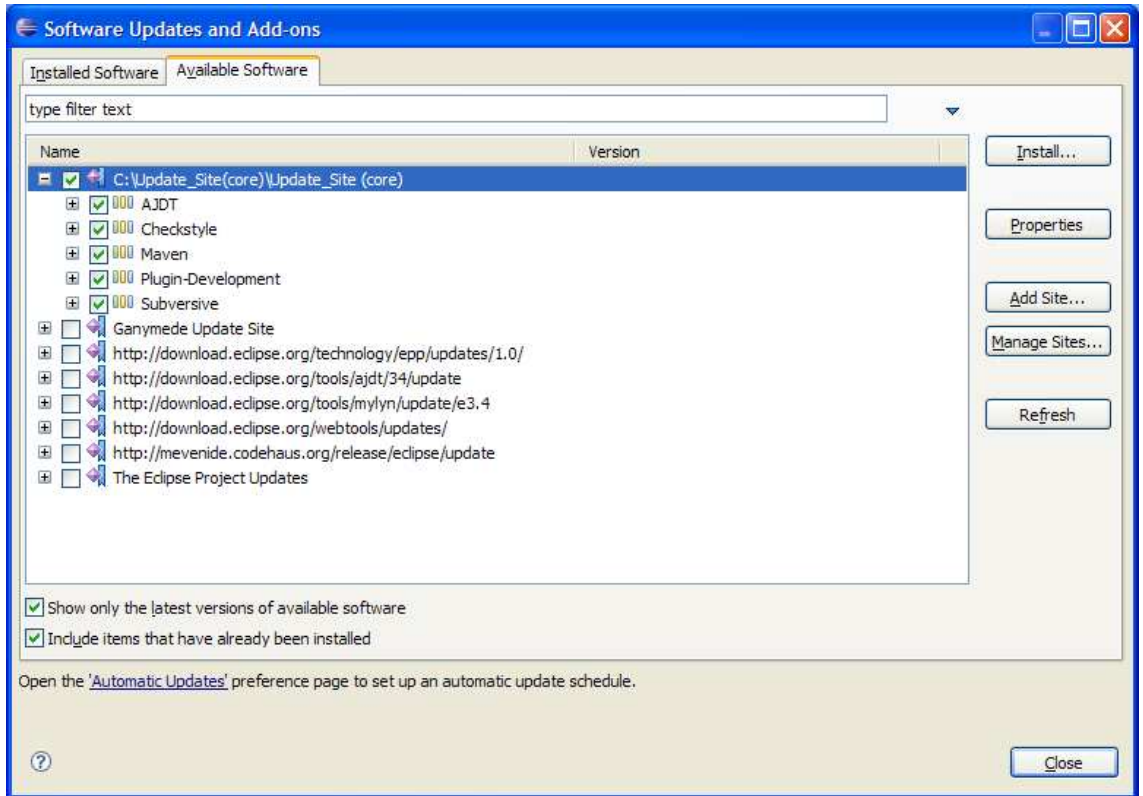
**Figure 20: Multiple problems have occurred dialog**

If you do encounter such a dialog, do not worry. There are two reasons why this occurs:

- Eclipse cannot access the internet. (Proxy settings may be incorrect)
- A third party plug-in may contain an incorrect URL

Click OK and continue.

8. Select the newly added Update\_Site checkbox in the Software *Updates and Add-ons* window, and click *Install*.



**Figure 21: Selecting the Update\_Site(core)**

9. A dialog with a list of the plug-in you wish to install is displayed. Click Finish.

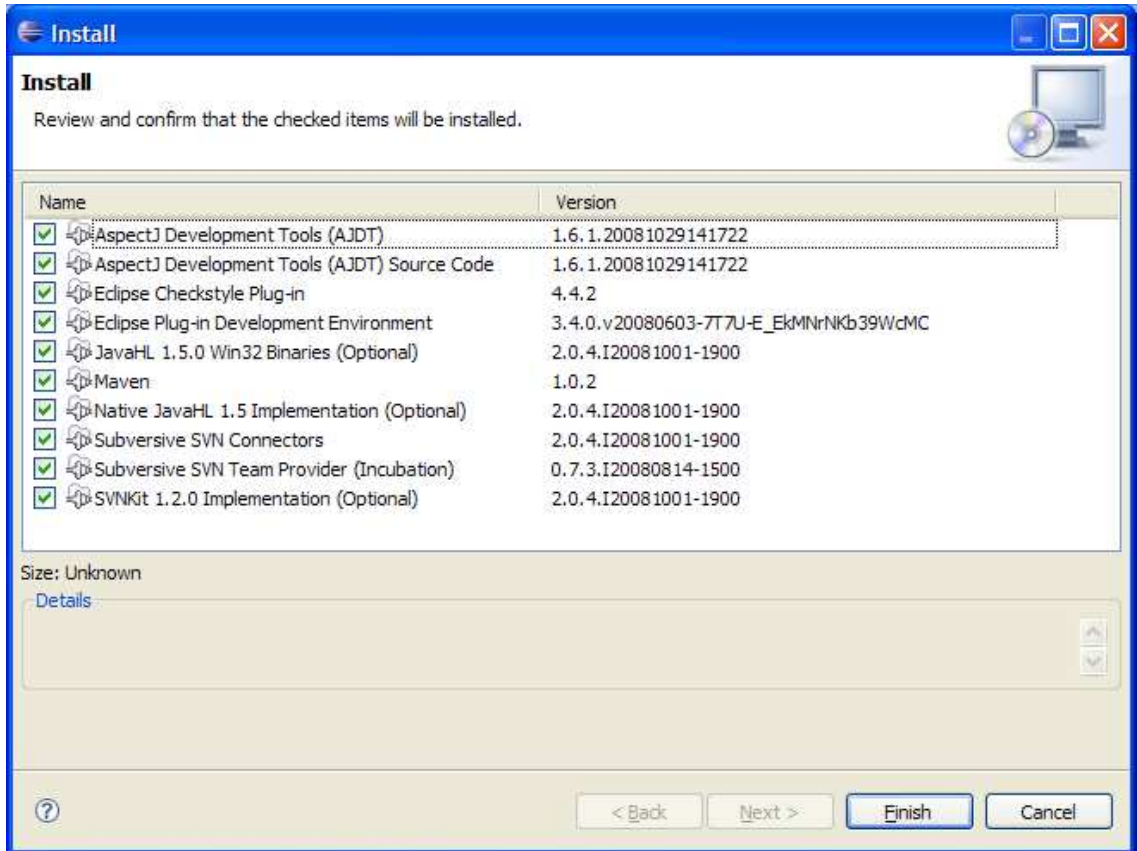


Figure 22. Previewing plug-ins for installation

10. The Update\_Site plug-ins are installed and after the process ends a dialog prompting you a restart of Eclipse is displayed. Click Yes.

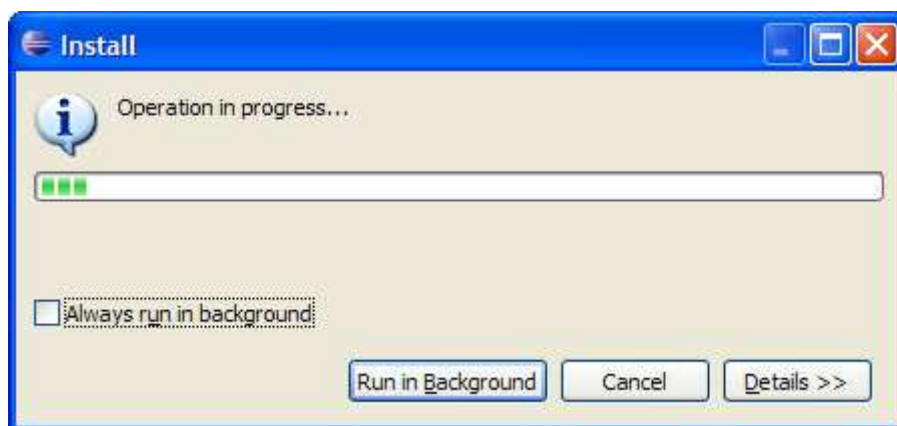


Figure 23: Plug-in Installation in progress



**Figure 24: Prompting for restart of Eclipse**

11. Eclipse will restart.

If you require further plug-in, you can choose to add further plug-ins at this stage such as the Update\_Site(optional) or additional third party plug-in from the internet. For further information on what is contained in the Update\_Site(optional).zip, please read Appendix D.

To migrate existing projects to your new ICW\_IDE Eclipse continue to the next section. Otherwise you may skip this chapter and continue to

## 5 Migrating existing projects into the ICW-IDE

You may already have had an existing version of Eclipse on your machine before installing the newer ICW-IDE Eclipse. You can migrate your older projects to the new ICW-IDE Eclipse by following the steps below:



### Note:

Please do not configure your old workspace with the new ICW-IDE Eclipse version installed. This is because the new ICW-IDE Eclipse contains configurations that are not compatible with older versions of Eclipse. To migrate your projects follow these following steps and you may delete your old workspace when completed.

1. In eclipse, select File -> Import
2. Select General -> Existing Projects into Workspace

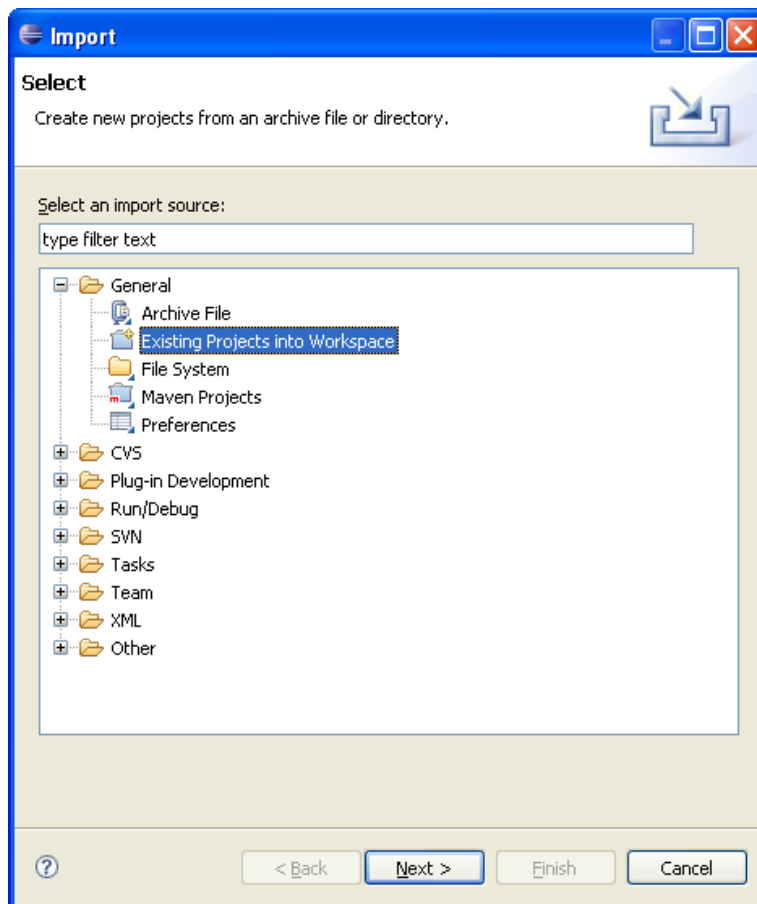
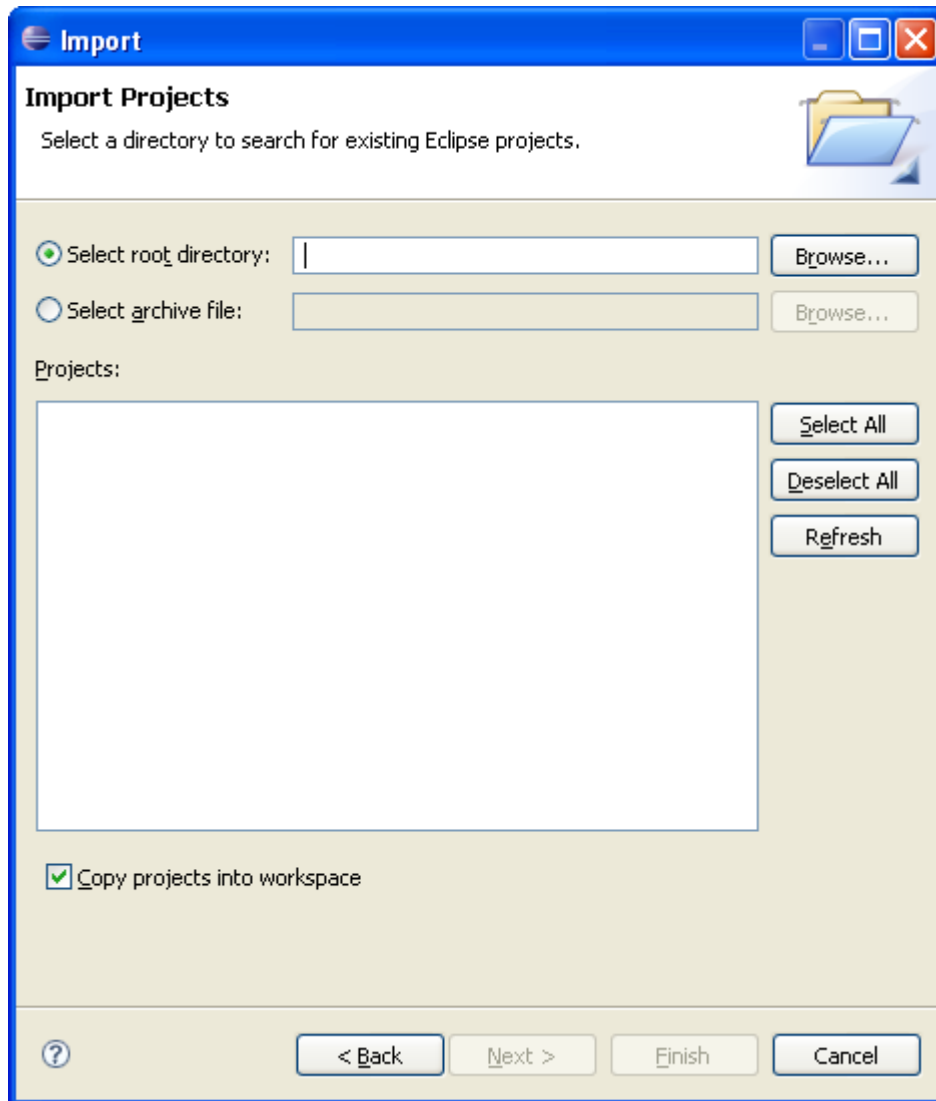


Figure 25: Selecting Existing Projects into Workspace

3. Add your projects by clicking Browse



**Figure 26: Selecting projects to migrate**

4. When all the projects are selected, click Finish and the migration will start.
5. When the migration is completed you can safely remove your old Eclipse and workspace directory. Before doing this check that your projects have successfully migrated to the ICW-IDE Eclipse.

## 6 Working with the ICW-IDE

Congratulations you have now installed the ICW-IDE and you are ready to start working on ICW based components.

In order to help you get started, we have provided training material such as:

- Tutorials
- References
- Howtos
- Release Notes

You can find more details about these on our Partner Guide DVD or directly at our ICW Developer Network website: <http://idn.icw-global.com/>

## A Troubleshooting

The most common issues that occur during the installation process are related to incorrect Java or Maven environment setup. Follow the next steps to detect what may be wrong:

- Check the Log files and see if this gives further information about your problem
- Check your environment and fix any issues
- Uninstall and Reinstall the ICW-IDE

### A 1 Checking the Install Logs

The ICW-IDE installer produces two log files contained in the directory where the ICW-IDE was installed:

- `INSTALL-IDE.log` contains the steps installed by the ICW-IDE installer. During the installation, the installer checks the environment for Java versions. If Java errors occur then they will be documented in this file.
- `Eclipse-install.log` is the output produced during the configuration of the Eclipse IDE. During this process Maven 1 is configured. This file will show any issues related to the Eclipse or Maven configuration.
- `Maven2-ide-configurator.log` is the output from the Maven 2 configuration process. During this process the Maven 2 environment is setup and Eclipse is configured.

### A 2 Uninstalling the ICW-IDE

Uninstall the ICW-IDE by selecting:

- Start -> Programs -> ICW\_IDE -> Uninstall ICW IDE 3.0.5

### A 3 Increasing HSQL database memory

If you encountered issues with HSQL DB such as the following:



```
"Exception in thread HSQLDB Connection @26d4f1"  
java.lang.OutOfMemoryError: Java heap space
```

---

increase the memory for the database.

This can be done by:

- Navigating to your ICW-IDE Installation directory
- In the *database\hsqldb\bin* open *server-start.bat*
- Increase the memory (highlight by bold font):

```
"%JAVA_HOME%\bin\java" -Xmx512m -cp hsqldb-1.8.0.2.jar org.hsqldb.Server
```

## B Checking Environment Settings

Some users may wish to use existing versions of software included with the ICW\_IDE. This sections explains how to use pre-installed versions of Java and Maven.

### B 1 Checking the JAVA Environment

#### B 1.1.1 Check JAVA is correctly installed

If you already have Java installed in your machine, check that it is a working version. The easiest way to check if Java is installed correctly is to check the java version and its output:



**Command:**

```
> java -version
```

**Output:**

```
java version "1.6.0_11"
```

```
Java(TM) SE Runtime Environment (build 1.6.0_11-b33)
```

---

**Table 2: java -version and an expected output**

Next check that your JAVA\_HOME variable is correctly setup. Your PATH variable should also contain a reference to the bin/ directory of your correct Java installation.



**Command:**

```
> echo %JAVA_HOME%
```

**Output e.g:**

```
C:\JAVA\JAVA1.6
```

---

**Table 3: Checking JAVA\_HOME variable**



**Command:**

```
> echo %PATH%
```

**Output e.g:**

```
..... C:\JAVA\JAVA1.6\bin ....
```

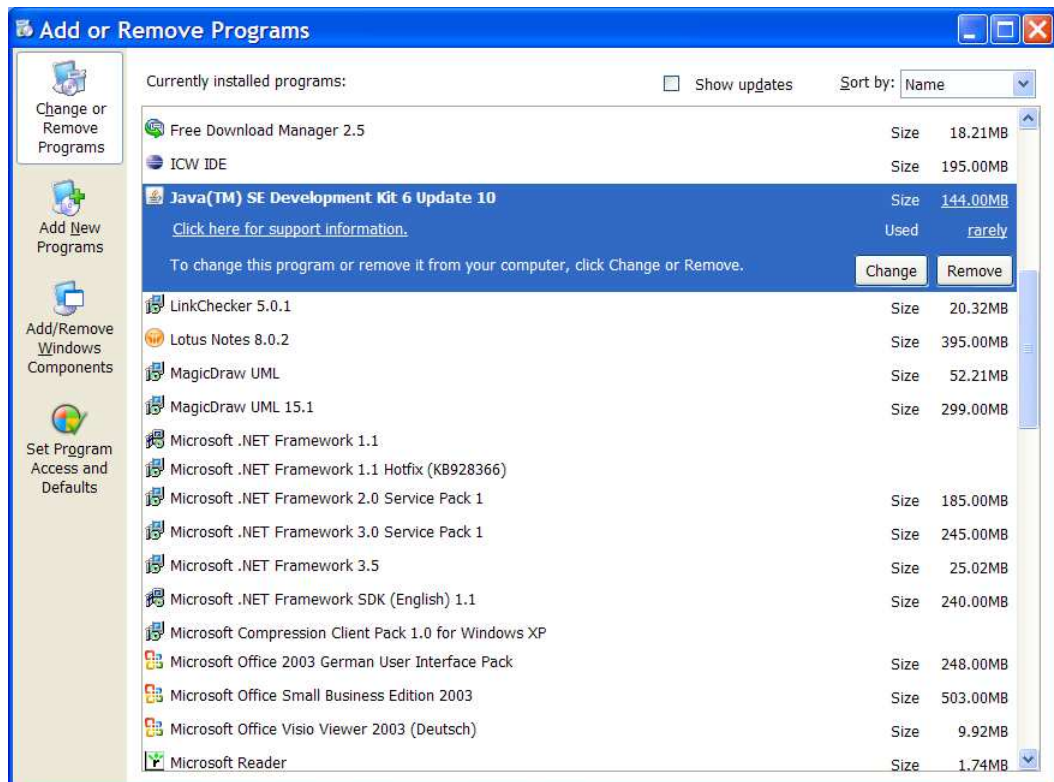
**Table 4: Check path variable for Java references and make sure it references the bin directory**

If you encounter problems with running java or your variables (JAVA\_HOME, PATH) continue with the following sections, otherwise you can go directly to the checking Maven environment section.

**B 1.1.2 Removing older versions of Java**

Only remove your Java version if you had issues with the previous steps

- Open 'Start->Settings->Control Panel->Add/Remove Programs'
- Select your version of Java
- Follow the uninstall process



### B 1.1.3 Removing obsolete Java Environment Settings

Remove all references to Java from your environment

- Open 'Start->Settings->System'
- Click on the Advanced tab
- Select Environment Variables
- Remove JAVA\_HOME variable from user and system variables sections by selecting Delete.
- Remove java references from PATH in user and system variable sections by selecting Edit.

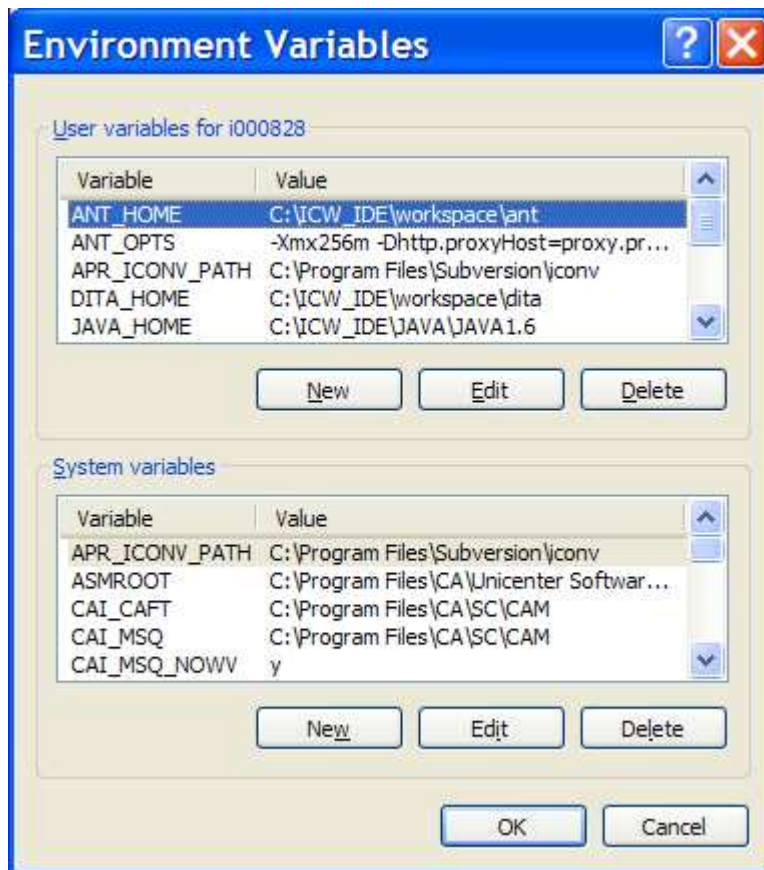


Figure 27: Environment Variables

### B 1.2 Check Maven Environment

The following provides instructions for the setup for Maven 1. Maven 2 has a similar environment. To test a Maven 2 environment replace *MAVEN\_HOME* with *M2\_HOME*, and the command *maven* with *mvn*.

In addition, Maven 2 does not use a `build.properties` file. Instead a `settings.xml` file is used to configure Maven 2 and this can be located at:

- %USERPROFILE%\.m2/settings.xml.

The following sub sections detail how to check Maven 1:

### B 1.2.1 Check MAVEN 1 is correctly installed

If you already have Maven installed in your machine, check that it is a working version. The easiest way to check if Maven is installed correctly is to check that Maven runs on a command:



#### Command:

```
> maven
```

#### Output (e.g.):

```

    ____
   /  __ \   \ Apache   /
  /| | | |   \ v / -_) ' \ ~ intelligent projects ~
 | | | | |   \| \ /   | | |   v. 1.1

WARNING: No pom file was found, assuming default settings!
-----

BUILD SUCCESSFUL

-----

Total time   : 3 seconds
Finished at  : 18 February 2009 12:37:19 CET
Final Memory : 1M/4M
-----

```

**Table 5: java -version and an expected output**

Next check that your MAVEN\_HOME variable is correctly setup. Your PATH variable should also contain a reference to the bin/ directory of your correct Maven installation.



#### Command:

```
> echo %MAVEN_HOME%
```

#### Output e.g:

```
C:\Maven\maven
```

**Table 6: Checking MAVEN\_HOME variable**

**Command:**

```
> echo %PATH%
```

**Output e.g:**

```
..... C:\Maven\maven\bin ....
```

---

**Table 7: Check path variable for Java references and make sure it references the bin directory**

If you encounter problems with running maven or your variables (MAVEN\_HOME, PATH) continue with the following section. Unlike Java, Maven does not need to be uninstalled. You can remove obsolete settings in your environment.

## B 1.2.2 Removing obsolete Maven Environment Settings

Remove all references to Java from your environment

- Open 'Start->Settings->System'
- Click on the Advanced tab
- Select Environment Variables
- Remove MAVEN\_HOME variable from user and system variables sections by selecting Delete
- Remove java references from PATH in user and system variable sections by selecting Edit.

## B 1.2.3 Configuring the Maven build.properties file

Maven reads settings from the `build.properties` file. This is located in the `%USERPROFILE%` directory. If you are not sure where this resides, open a command prompt and use the following command:

**Command:**

```
> echo %USERPROFILE%
```

---

**Table 8: Displaying %USERPROFILE% settings**

Your `build.properties` should look something like the following example:



### Command:

```
### Remote repository  
  
maven.repo.remote=  
  
maven.repo.local=C:/ICW_IDE/.maven/repository  
  
maven.home.local=C:/ICW_IDE/.maven  
  
maven.eclipse.workspace=C:/ICW_IDE/workspace
```

**Table 9**

Make sure the `maven.repo.local`, `maven.home.local` and `maven.eclipse.workspace` values are correct.

- `maven.repo.local`: This should reference your local maven repository.
- `maven.home.local`: This should reference your local `.maven` directory
- `maven.eclipse.workspace`: This should reference your Eclipse workspace



### Note:

If you wish to use another Maven installation and older repository than the one provided with the ICW-IDE then you should configure the `build.properties` file accordingly. Please note that the ICW-IDE comes with a version of the `eHealthFramework`. Using your older repository may not support some development of ICW application within the ICW-IDE.

## C References

### C 1.1 Maven

For more information about Maven please consult the main Maven project site on [apache.org](http://maven.apache.org):

<http://maven.apache.org>

Information about the Maven concepts can be downloaded as a book (mavenbook) at the following location:

**<http://mavenbook.xwiki.com/xwiki/bin/download/Main/Source/mavenbook-1.0.zip>**

More information about the Maven book is available at **<http://www.mavenbook.org/>**.

Alternatively, for Maven 2, Sonatype provide an excellent online book. For further details see: **<http://www.sonatype.com/books/m2eclipse-book/reference/>**

### C 1.2 Eclipse

The main website of the project contains more detailed information on Eclipse:

**<http://www.eclipse.org/>**

At the following location you can find the Eclipse documentation:

**<http://www.eclipse.org/documentation/main.html>**

### C 1.3 Subversion

The project site of Subversion offers more information on Subversion:

**<http://subversion.tigris.org>**

### C 1.4 Jira

More information on the ehf-assembly application can be found here:

**<http://www.atlassian.com/software/jira/docs/v3.5.1/index.html?clicked=jirahelp>**

## D List of Current Eclipse Plug-Ins

The Eclipse version included as part of the IDE\_ICW\_3.0.4 includes the following pre-installed plug-ins:

- Eclipse CVS Client
- Eclipse Java Development Tools
- Eclipse XML Editors and Tools
- EMF
- Graphical Editing Framework
- Mylyn
- Usage Data Collector
- WST
- XSD

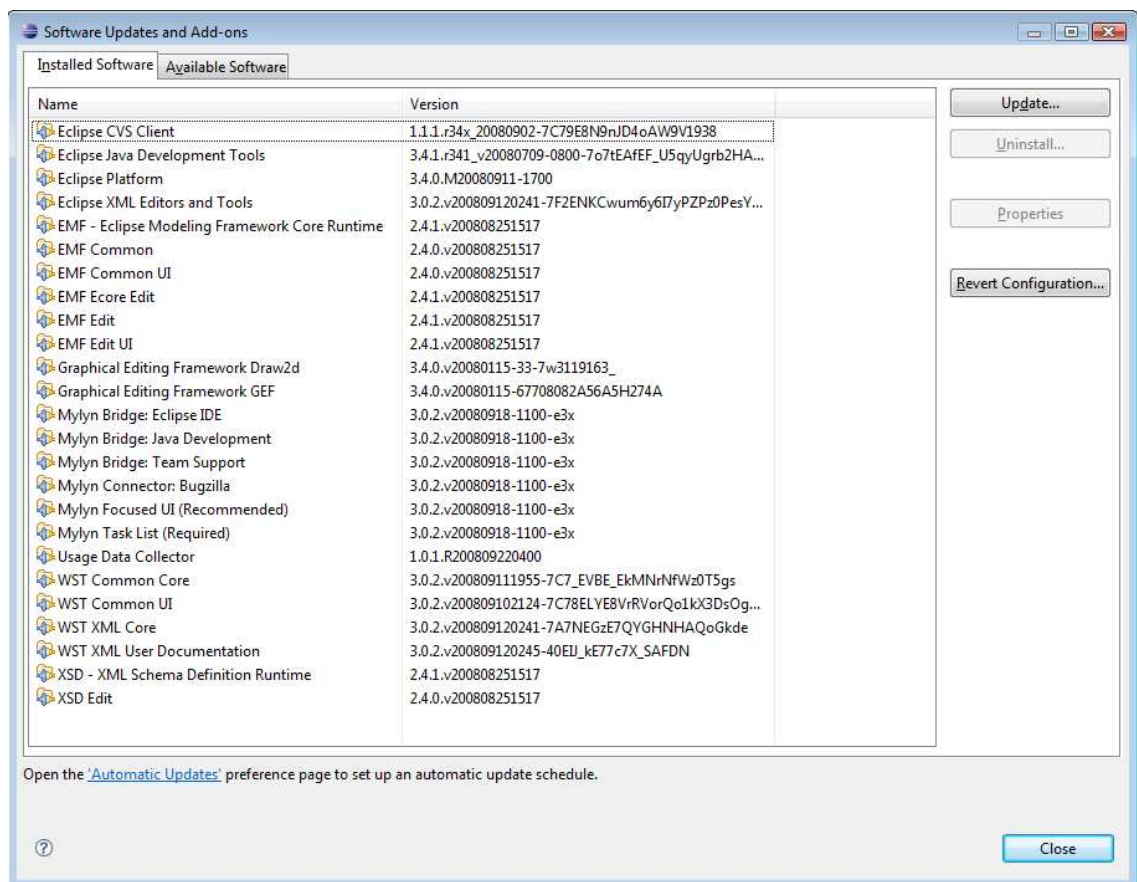


Figure 28: Pre-installed Plugins and version that are provided with Eclipse

## D 1 Core Plugin Updates

Contained in the Update\_Site(core) are the core updates recommended for the ICW\_IDE. This includes:

- ADJT
- Checkstyle
- FindBugs
- Maven
- Plugin-development (required for ADJT)
- Subversive

## D 2 Optional Plugin Updates

Contained in the Update\_Site(optional) are the optional updates for the ICW\_IDE. This includes:

- EMF (required by Topcased and OAW)
- EMF Model (required by Topcased)
- GMF (required by Topcased and OAW)
- HibernateTools
- Mevenide
- Oaw
- OCL (required by Topcased)
- Orangevolt XSLT
- Quantum DB
- Spring IDE
- Topcased
- UML2 (required by Topcased, and oAW)
- W3C

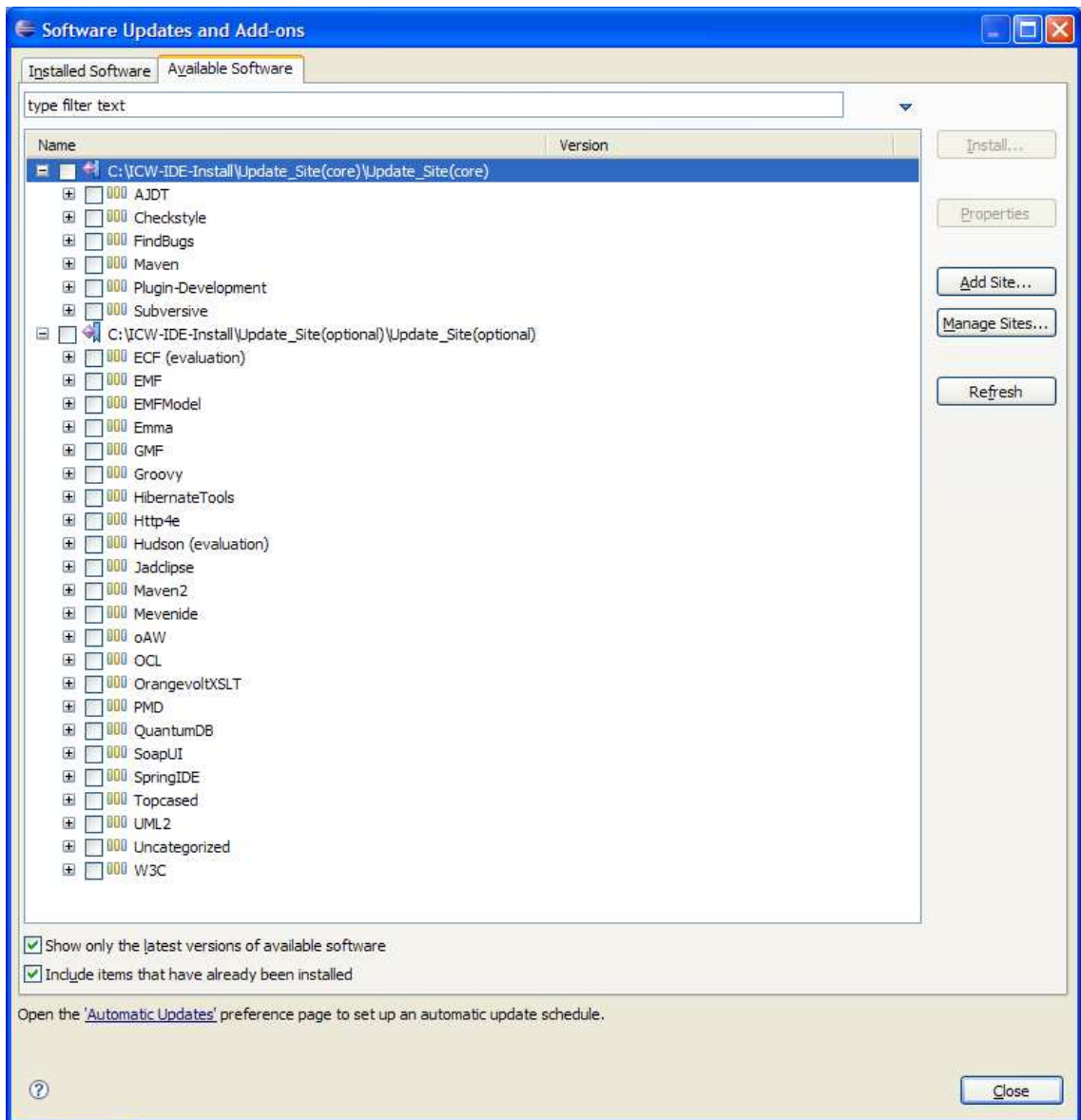


Figure 29: Overview of the contents of the Core and Optional Plugins

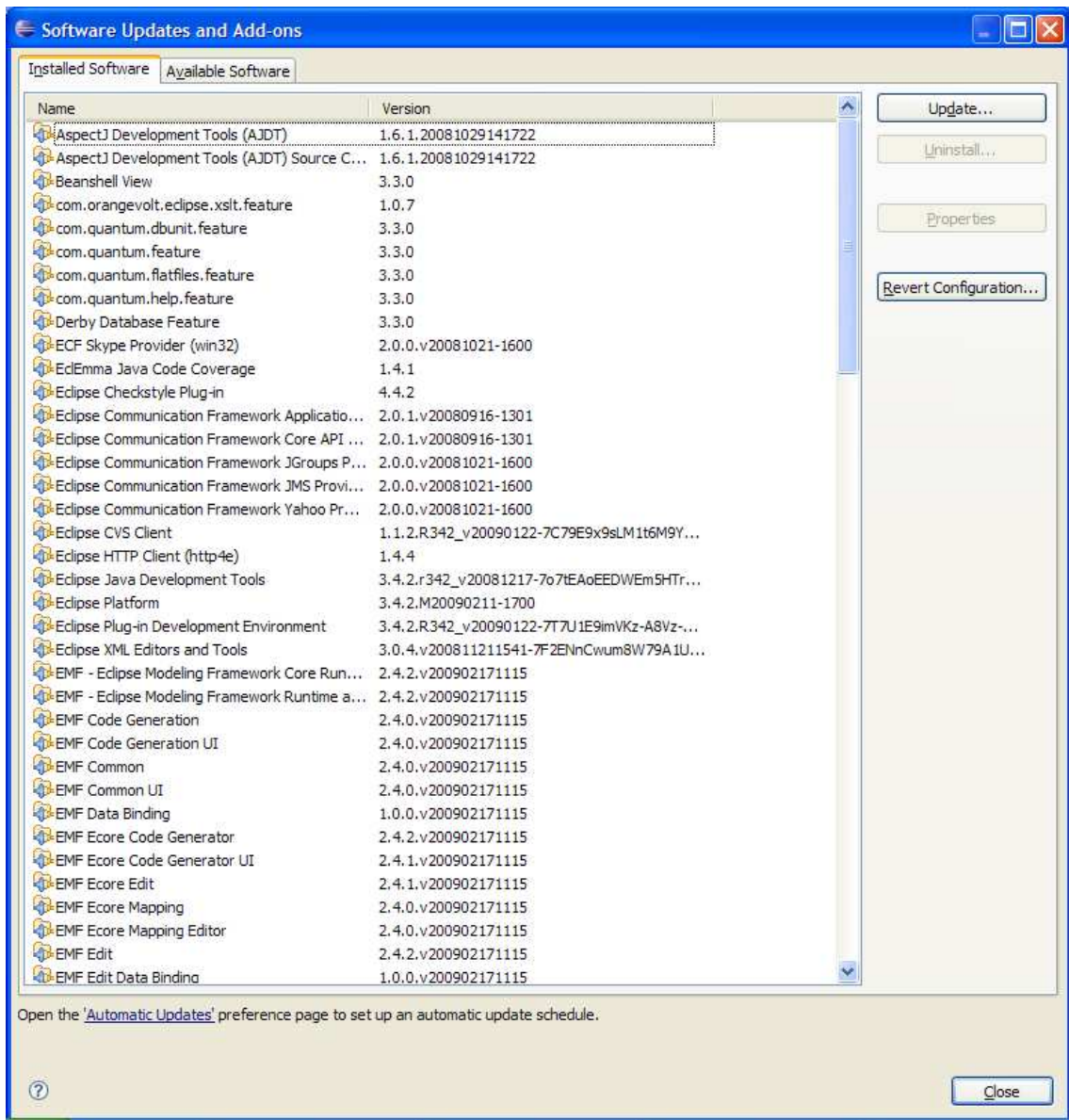


Figure 30: Core and Optional Plugins installed for ICW\_IDE, part 1

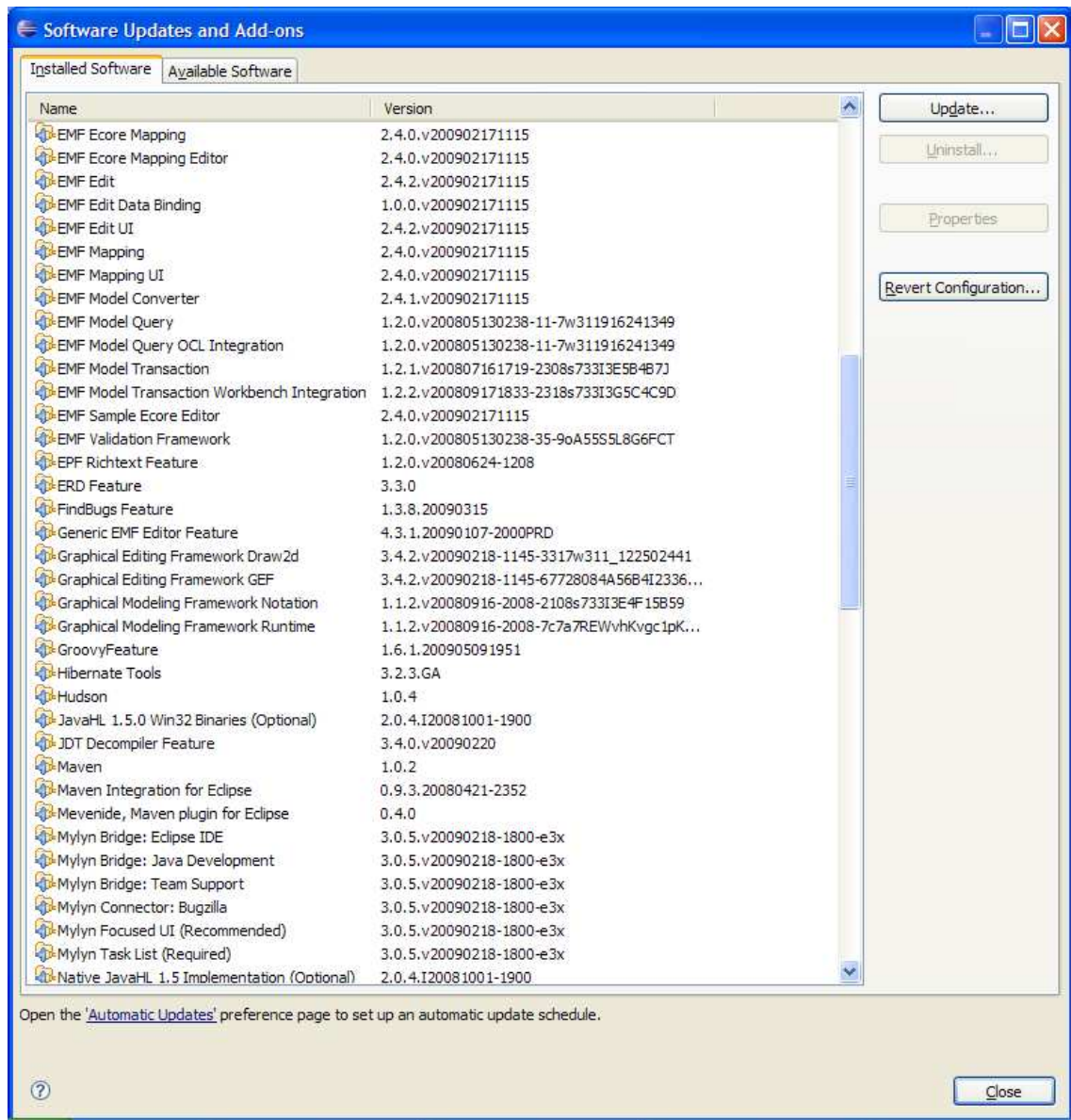


Figure 31 Core and Optional Plugins installed for ICW\_IDE, part 2

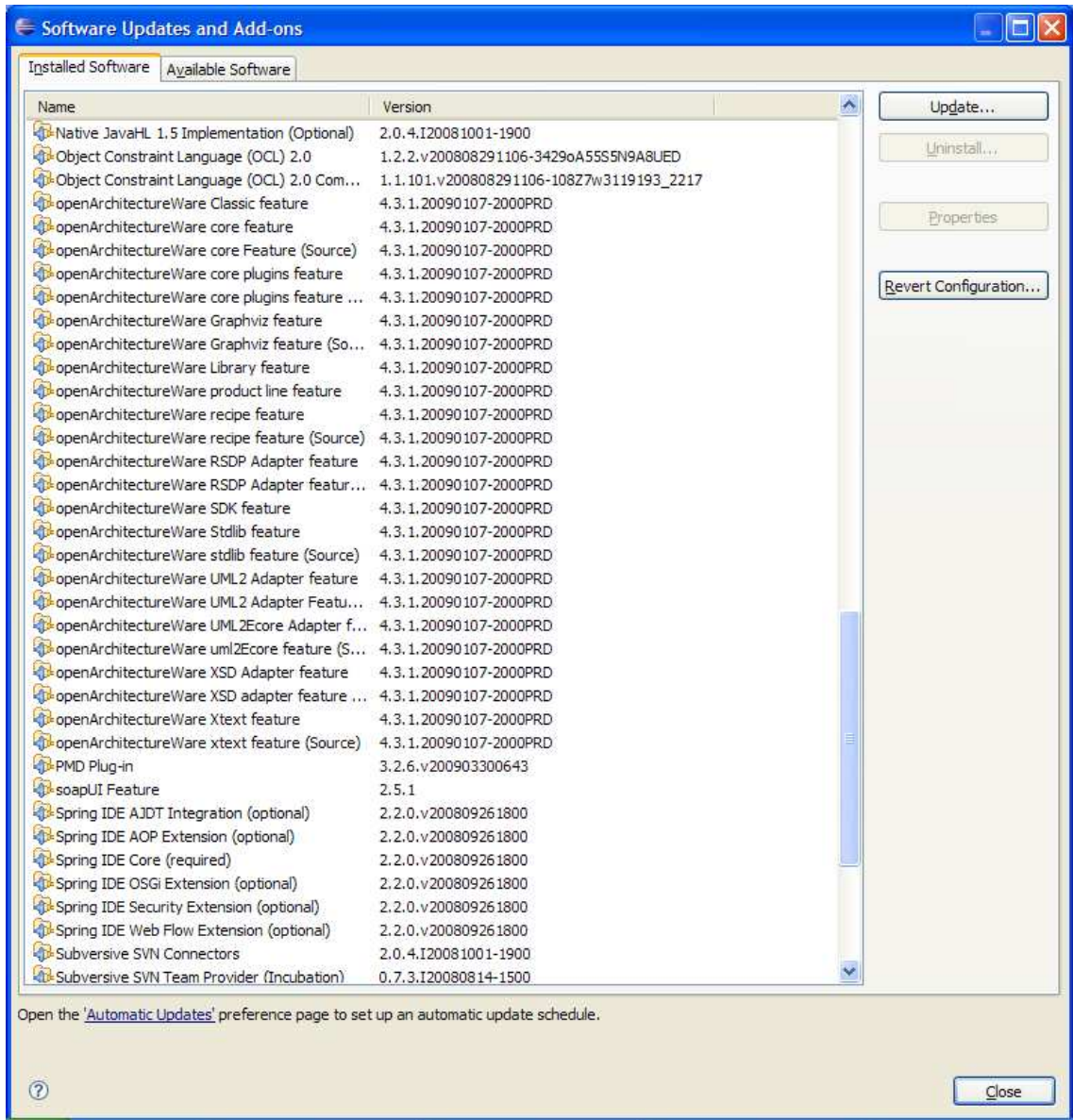


Figure 32 Core and Optional Plugins installed for ICW\_IDE, part 3

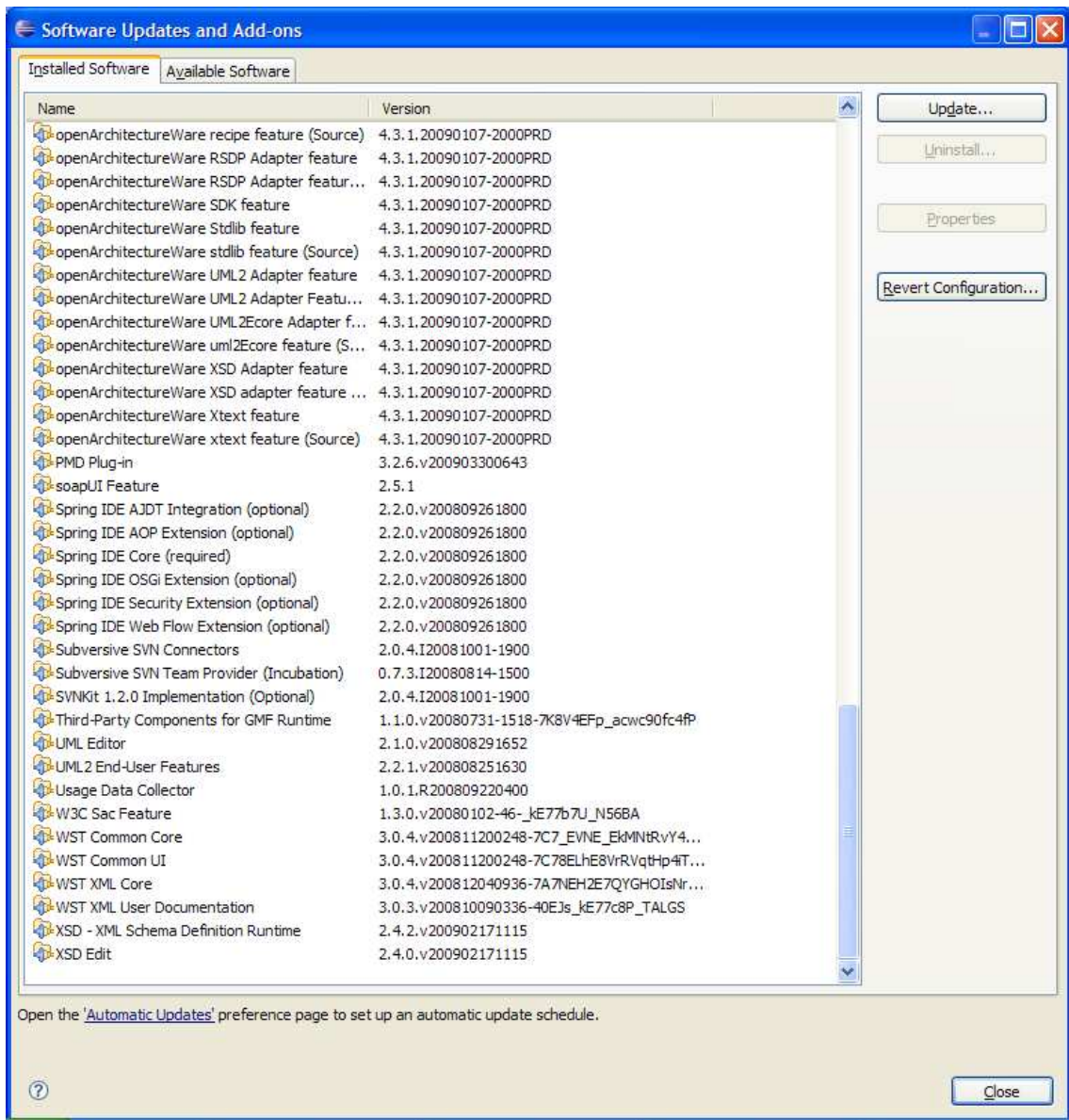


Figure 33 Core and Optional Plugins installed for ICW\_IDE, part 4

## E List of Figures

Figure 1: Welcome Dialog.....	11
Figure 2: License Agreement Dialog.....	12
Figure 3: Selection of Components to install – eHF Components.....	13
Figure 4: Selection of Components to install – IPF Components .....	14
Figure 5: Choose the Installation Folder .....	16
Figure 6: Configure Maven Repositories .....	17
Figure 7: Configuring Maven for remote access .....	18
Figure 8: Installing the ICW-IDE.....	19
Figure 9: Successful install message.....	20
Figure 10: Installation finished .....	21
Figure 11: The Extraction Wizard .....	23
Figure 12: Select a Destination for the ZIP Archive .....	23
Figure 13: Extraction is complete.....	24
Figure 14: Path to the contents on the Update Site.....	24
Figure 15: The Software Update dialog in Eclipse.....	25
Figure 16: Software Updates and Add-ons.....	26
Figure 17: Choose <i>Local</i> .....	26
Figure 18: Browse For Folder – selecting the Update_Site(core) folder .....	27
Figure 19: Add Site .....	27
Figure 20: Multiple problems have occurred dialog .....	28
Figure 21: Selecting the Update_Site(core).....	29
Figure 22. Previewing plug-ins for installation .....	30
Figure 23: Plug-in Installation in progress.....	30
Figure 24: Prompting for restart of Eclipse .....	31
Figure 25: Selecting Existing Projects into Workspace .....	32
Figure 26: Selecting projects to migrate .....	33
Figure 27: Environment Variables.....	39
Figure 28: Pre-installed Plugins and version that are provided with Eclipse .....	44
Figure 29: Overview of the contents of the Core and Optional Plugins .....	46
Figure 30: Core and Optional Plugins installed for ICW_IDE, part 1 .....	47
Figure 31 Core and Optional Plugins installed for ICW_IDE, part 2.....	48

Figure 32 Core and Optional Plugins installed for ICW\_IDE, part 3.....49

Figure 33 Core and Optional Plugins installed for ICW\_IDE, part 4.....50