

Eclipse, WildFly and SwitchYard installation guide

Balázs Simon (sbalazs@iit.bme.hu), BME IIT, 2015

1 Introduction

This document describes how to install and configure the Eclipse IDE, the WildFly server and the SwitchYard environment to develop and host ESB applications.

2 Downloads

Download the **JDK** from:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

The current version is: **JDK 8u31**

Download **Maven** from:

<https://maven.apache.org/download.cgi>

The current version is: **3.3.1**

Download **Eclipse IDE for JavaEE Developers** from:

<https://eclipse.org/downloads/>

The current version is: **Luna SR1a (4.4.1)**

Download **SwitchYard (switchyard-...-WildFly.zip)** from:

<http://switchyard.jboss.org/downloads>

The current version is: **2.0.0.CR1**

Download **WildFly** from:

<http://wildfly.org/downloads/>

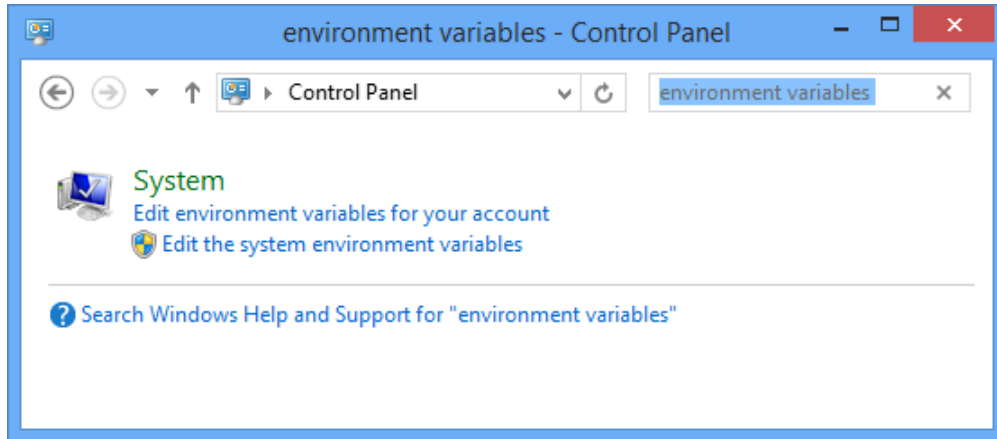
The version compatible with the current SwitchYard version is: **8.1.0.Final**

3 Installation

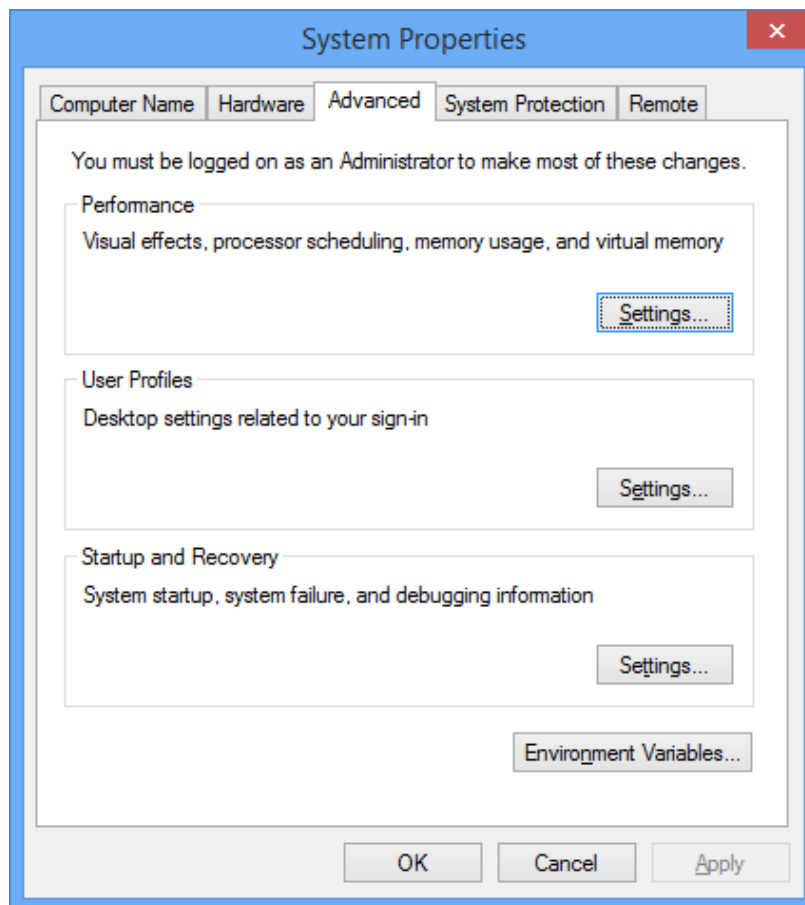
3.1 Install JDK 8 and set the environment variables

Install the JDK 8 using the installation wizard.

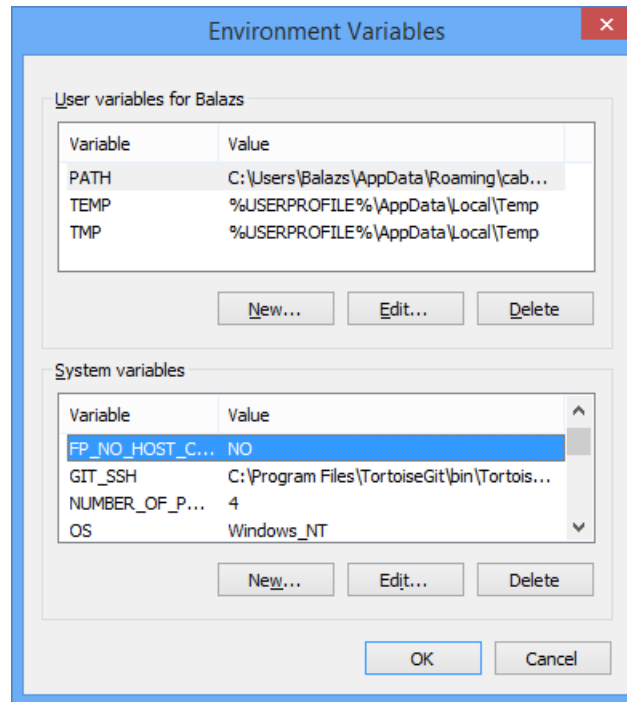
Open **Control Panel** and type „**environment variables**” in the search box in the upper right corner:



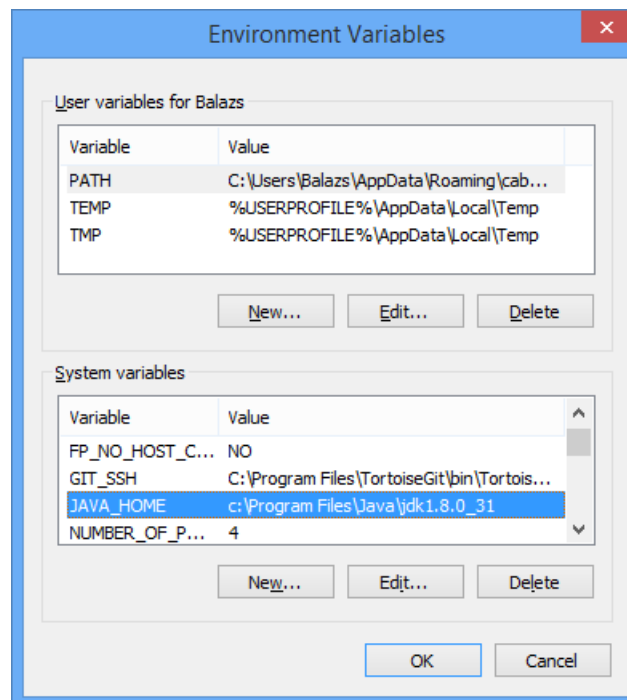
Click on the **Edit the system environment variables** link and the following window will open:



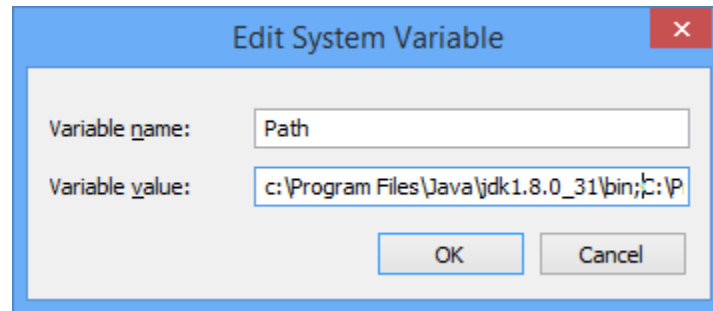
Click on **Environment Variables...**:



If there is no **JAVA_HOME** in the list under **System variables** then add it to the list with the **New...** button! If it is already set use the **Edit...** button to correct its value if necessary! Make sure to include the appropriate version number in the path, e.g.:



Look for the **PATH** environment variable in the list under **System variables**, and prepend the path of the JDK **bin** folder to the beginning separated by a semicolon, e.g.:



To apply these changes you may have to restart your computer, but it is usually enough to close and reopen all the applications.

3.2 Install Maven

Extract the downloaded **Maven zip** file to a folder that does not contain any spaces or special characters. For example extract the zip to:

```
c:\Programs\apache-maven-3.3.1
```

Set the **PATH** environment variable as described in the previous section so that it also includes the **Maven bin** folder:

```
c:\Programs\apache-maven-3.3.1\bin
```

3.3 Install the WildFly server

Extract the downloaded **WildFly zip** file to a folder that does not contain any spaces or special characters. For example extract the zip to:

```
c:\Programs\wildfly-8.1.0.Final
```

To test the server installation, start it with the following command:

```
c:\Programs\wildfly-8.1.0.Final\bin\standalone.bat
```

If the installation is correct the server is started without any errors.

Finally, stop the server with **Ctrl+C**.

3.4 Install SwitchYard

Extract the contents of the **SwitchYard-...-WildFly zip** so that the files and folders overwrite everything inside the WildFly installation folder:

c:\Programs\wildfly-8.1.0.Final

If the server should run side-by-side of another server (e.g. another WildFly server, Tomcat server, GlassFish server) you can change the port numbers of the server inside the following configuration file:

c:\Programs\wildfly-8.1.0.Final\standalone\configuration\standalone.xml

Find the following entries at the end of the file and if necessary, change the port numbers to free ports:

```
...
    <socket-binding-group name="standard-sockets" default-interface="public"
        port-offset="{jboss.socket.binding.port-offset:0}">
        <socket-binding name="management-http" interface="management"
            port="{jboss.management.http.port:19990}" />
        <socket-binding name="management-https" interface="management"
            port="{jboss.management.https.port:19993}" />
        <socket-binding name="ajp" port="{jboss.ajp.port:18009}" />
        <socket-binding name="http" port="{jboss.http.port:18080}" />
        <socket-binding name="https" port="{jboss.https.port:18443}" />
        <socket-binding name="txn-recovery-environment" port="14712" />
        <socket-binding name="txn-status-manager" port="14713" />
        <outbound-socket-binding name="mail-smtp">
            <remote-destination host="localhost" port="25" />
        </outbound-socket-binding>
    </socket-binding-group>
</server>
```

To test the server installation, start it with the following command:

c:\Programs\wildfly-8.1.0.Final\bin\standalone.bat

If the installation is correct the server is started without any errors. Finally, stop the server with **Ctrl+C**.

Edit the following file:

c:\Programs\wildfly-8.1.0.Final\bin\jboss-cli.xml

And change the management port number to match the port number defined in the standalone.xml:

```
<?xml version='1.0' encoding='UTF-8'?>
<jboss-cli xmlns="urn:jboss:cli:2.0">
    <default-protocol use-legacy-override="true">http-remoting</default-protocol>
    <!-- The default controller to connect to when 'connect' command is executed w/o
arguments -->
    <default-controller>
        <protocol>http-remoting</protocol>
        <host>localhost</host>
        <port>19990</port>
    </default-controller>
    ...
</jboss-cli>
```

3.5 Enable JMS

Find the **standalone.xml** configuration file of WildFly:

c:\Programs\wildfly-8.1.0.Final\standalone\configuration\standalone.xml

Next to this file there is a **standalone-full.xml** file, too. The modifications of the **standalone.xml** described below can be copied from this **standalone-full.xml** file.

Insert the messaging module at the beginning of the **standalone.xml**:

```
<?xml version="1.0" ?>
<server xmlns="urn:jboss:domain:2.1">
  <extensions>
    ...
    <extension module="org.jboss.as.logging"/>
    <extension module="org.jboss.as.mail"/>
    <extension module="org.jboss.as.messaging"/>
    <extension module="org.jboss.as.naming"/>
    ...
  </extensions>
  ...
</server>
```

Add the **messaging** subsystem configuration between the **mail** and **naming** subsystems:

```
<?xml version="1.0" ?>
<server xmlns="urn:jboss:domain:2.1">
  ...
  <profile>
    ...
    <subsystem xmlns="urn:jboss:domain:mail:2.0">
      <mail-session name="default" jndi-name="java:jboss/mail/Default">
        <smtp-server outbound-socket-binding-ref="mail-smtp"/>
      </mail-session>
    </subsystem>
    <subsystem xmlns="urn:jboss:domain:messaging:2.0">
      <hornetq-server>
        <persistence-enabled>true</persistence-enabled>
        <journal-file-size>102400</journal-file-size>
        <journal-min-files>2</journal-min-files>
        <connectors>
          <http-connector name="http-connector" socket-binding="http">
            <param key="http-upgrade-endpoint" value="http-acceptor"/>
          </http-connector>
          <http-connector name="http-connector-throughput" socket-binding="http">
            <param key="http-upgrade-endpoint" value="http-acceptor-throughput"/>
            <param key="batch-delay" value="50"/>
          </http-connector>
          <in-vm-connector name="in-vm" server-id="0"/>
        </connectors>
        <acceptors>
          <http-acceptor name="http-acceptor" http-listener="default"/>
          <http-acceptor name="http-acceptor-throughput" http-listener="default">
            <param key="batch-delay" value="50"/>
            <param key="direct-deliver" value="false"/>
          </http-acceptor>
          <in-vm-acceptor name="in-vm" server-id="0"/>
        </acceptors>
      </hornetq-server>
    </subsystem>
  </profile>
  ...
</server>
```

```

    <security-settings>
      <security-setting match="#">
        <permission type="send" roles="guest"/>
        <permission type="consume" roles="guest"/>
        <permission type="createNonDurableQueue" roles="guest"/>
        <permission type="deleteNonDurableQueue" roles="guest"/>
      </security-setting>
    </security-settings>
    <address-settings>
      <!--default for catch all-->
      <address-setting match="#">
        <dead-letter-address>jms.queue.DLQ</dead-letter-address>
        <expiry-address>jms.queue.ExpiryQueue</expiry-address>
        <redelivery-delay>0</redelivery-delay>
        <max-size-bytes>10485760</max-size-bytes>
        <address-full-policy>PAGE</address-full-policy>
        <page-size-bytes>2097152</page-size-bytes>
        <message-counter-history-day-limit>10</message-counter-history-day-limit>
      </address-setting>
    </address-settings>
    <jms-connection-factories>
      <connection-factory name="InVmConnectionFactory">
        <connectors>
          <connector-ref connector-name="in-vm"/>
        </connectors>
        <entries>
          <entry name="java:/ConnectionFactory"/>
        </entries>
      </connection-factory>
      <connection-factory name="RemoteConnectionFactory">
        <connectors>
          <connector-ref connector-name="http-connector"/>
        </connectors>
        <entries>
          <entry name="java:jboss/exported/jms/RemoteConnectionFactory"/>
        </entries>
      </connection-factory>
      <pooled-connection-factory name="hornetq-ra">
        <transaction mode="xa"/>
        <connectors>
          <connector-ref connector-name="in-vm"/>
        </connectors>
        <entries>
          <entry name="java:/JmsXA"/>
          <!-- Global JNDI entry used to provide a default JMS Connection factory
to EE application -->
          <entry name="java:jboss/DefaultJMSConnectionFactory"/>
        </entries>
      </pooled-connection-factory>
    </jms-connection-factories>
  </hornetq-server>
</subsystem>
<subsystem xmlns="urn:jboss:domain:naming:2.0">
  <remote-naming/>
</subsystem>
...
</profile>
...
</server>

```

Add the messaging-group socket-binding at the end of the file, and also change the port number if necessary:

```
<?xml version="1.0" ?>
<server xmlns="urn:jboss:domain:2.1">
    ...
    <socket-binding-group name="standard-sockets" default-interface="public" port-
offset="${jboss.socket.binding.port-offset:0}">
        <socket-binding name="management-http" interface="management"
port="${jboss.management.http.port:19990}"/>
        <socket-binding name="management-https" interface="management"
port="${jboss.management.https.port:19993}"/>
        <socket-binding name="ajp" port="${jboss.ajp.port:18009}"/>
        <socket-binding name="http" port="${jboss.http.port:18080}"/>
        <socket-binding name="https" port="${jboss.https.port:18443}"/>
        <socket-binding name="messaging-group" port="0" multicast-
address="${jboss.messaging.group.address:231.7.7.7}" multicast-
port="${jboss.messaging.group.port:19876}"/>
        <socket-binding name="txn-recovery-environment" port="14712"/>
        <socket-binding name="txn-status-manager" port="14713"/>
        <outbound-socket-binding name="mail-smtp">
            <remote-destination host="localhost" port="25"/>
        </outbound-socket-binding>
    </socket-binding-group>
</server>
```

The final **standalone.xml** is attached to this installation guide.

To test the server configuration, start the server with the following command:

```
c:\Programs\wildfly-8.1.0.Final\bin\standalone.bat
```

If the configuration is correct the server is started without any errors. Finally, stop the server with **Ctrl+C**.

3.6 Adding users to the server

Users can be added to the server using the **add-user** command in the **bin** folder of WildFly:

```
c:\Programs\wildfly-8.1.0.Final\bin\add-user.bat
```

Execute this command and add a user with the following data (this is an admin user):

- **a) Management User**
- Username: **admin**
- Password: **admin**
- No groups (leave it empty)
- AS process to AS process connection: **no**

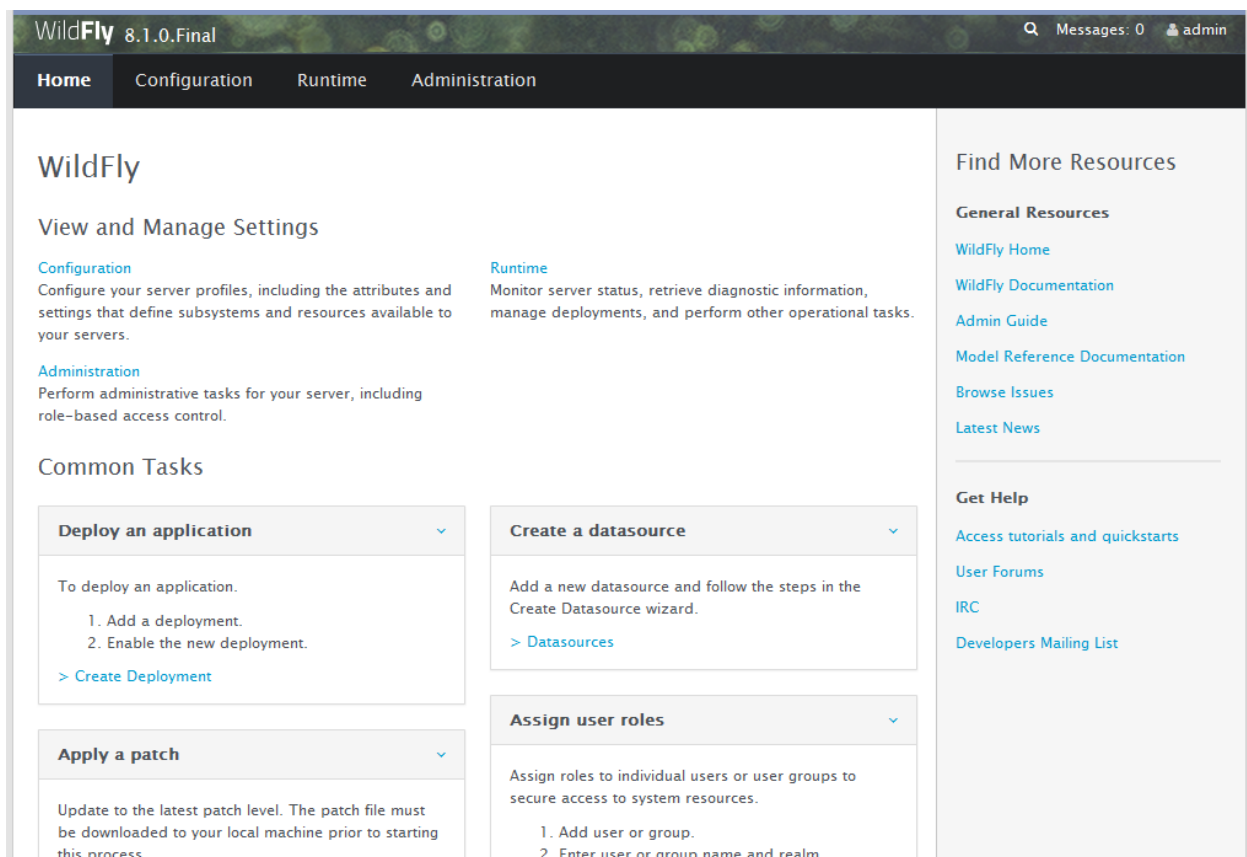
Execute this command again and add a user with the following data:

- **b) Application User**
- Username: **guest**
- Password: **guest**
- Groups: **guest**
- AS process to AS process connection: **no**

Start the server and type the following URL in a browser (make sure that the port number matches the management port number in the **standalone.xml** file):

<http://localhost:19990/console>

To log in, use the **admin** username and the **admin** password. The web page shows the admin console of WildFly:



3.7 Install Eclipse

Extract the downloaded Eclipse zip file to a folder that does not contain any spaces or special characters. For example, extract the contents of the **eclipse** directory from the zip to:

c:\Programs\eclipse_soi

Find the **eclipse.ini** file in this folder, and add or change the following lines shown in bold in the ini file (make sure that the path to the JRE server folder is correct):

```
-vm
c:\Program Files\Java\jdk1.8.0_31\jre\bin\server
-startup
plugins/org.eclipse.equinox.launcher_1.3.0.v20140415-2008.jar
--launcher.library
plugins/org.eclipse.equinox.launcher.win32.win32.x86_1.1.200.v20140603
-1326
-product
org.eclipse.epp.package.jee.product
--launcher.defaultAction
openFile
--launcher.XXMaxPermSize
256M
-showsplash
org.eclipse.platform
--launcher.XXMaxPermSize
256m
--launcher.defaultAction
openFile
--launcher.appendVmargs
-vmargs
-Dosgi.requiredJavaVersion=1.6
-Xms256m
-Xmx1024m
-XX:PermSize=128M
-XX:MaxPermSize=256M
```

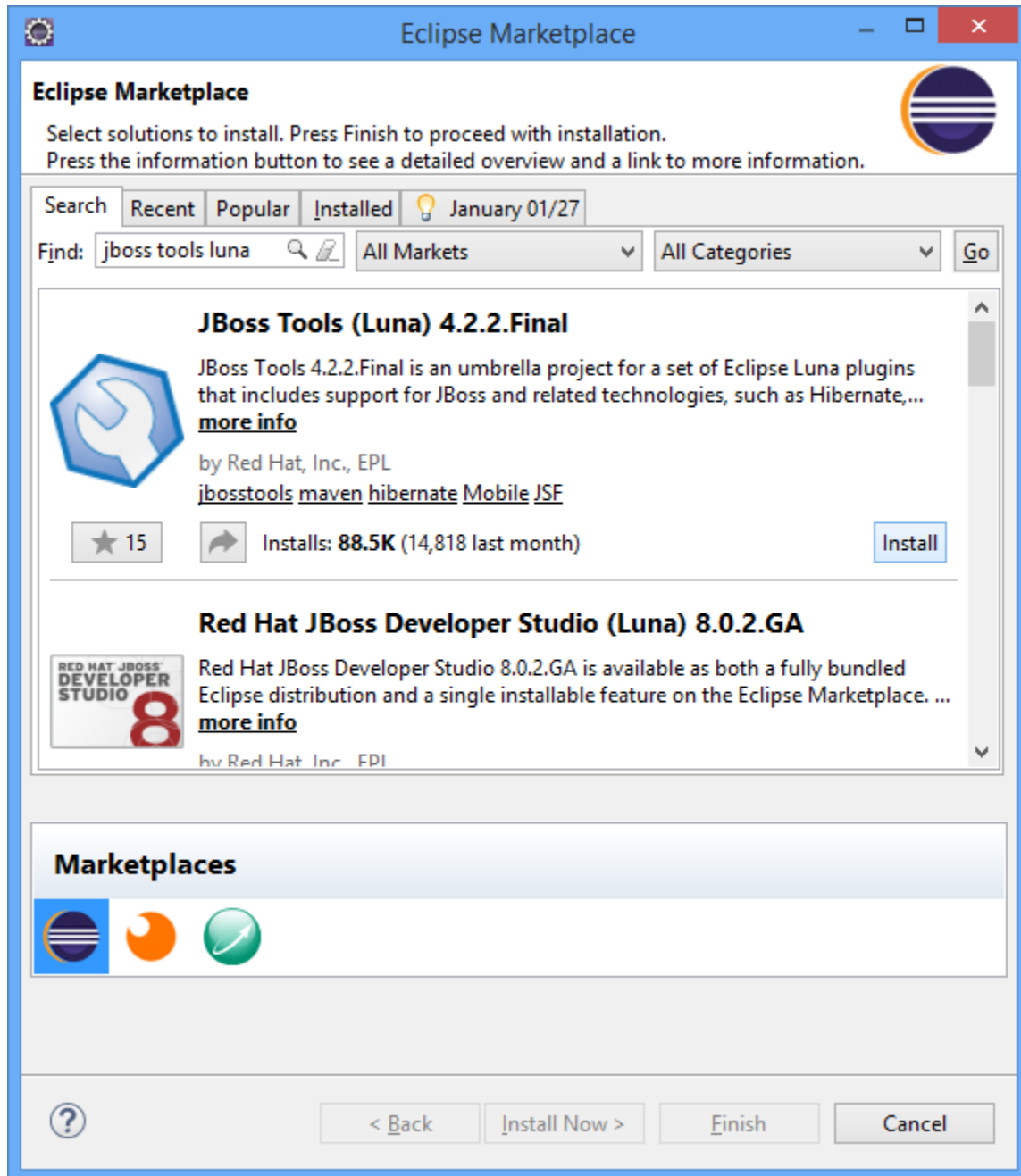
Start Eclipse with:

c:\Programs\eclipse_soi\eclipse.exe

Specify a folder for the workspace when it is asked for. Make sure it does not collide with another Eclipse if there are other Eclipse environments installed on your computer. Make sure that the path of the workspace does not contain any spaces or special characters.

3.8 Install JBoss Tools

JBoss Tools is required for Eclipse so that it can use the WildFly server. When Eclipse is running, click on **Help > Eclipse Marketplace...**, type **jboss tools luna** in the search box, and hit enter:

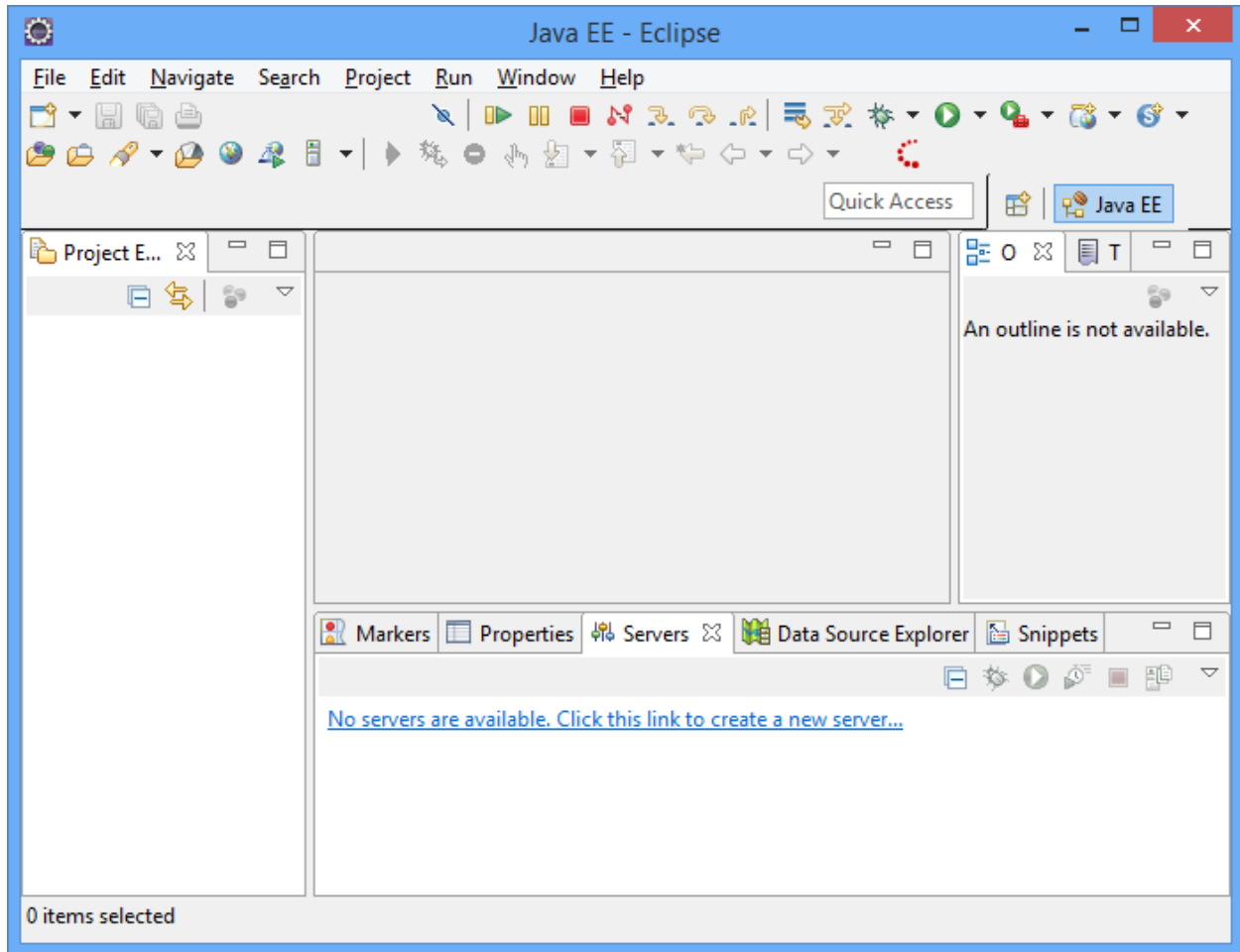


Install the **JBoss Tools (Luna) 4.2.2.Final** by clicking on the **Install** button and following the steps of the wizard.

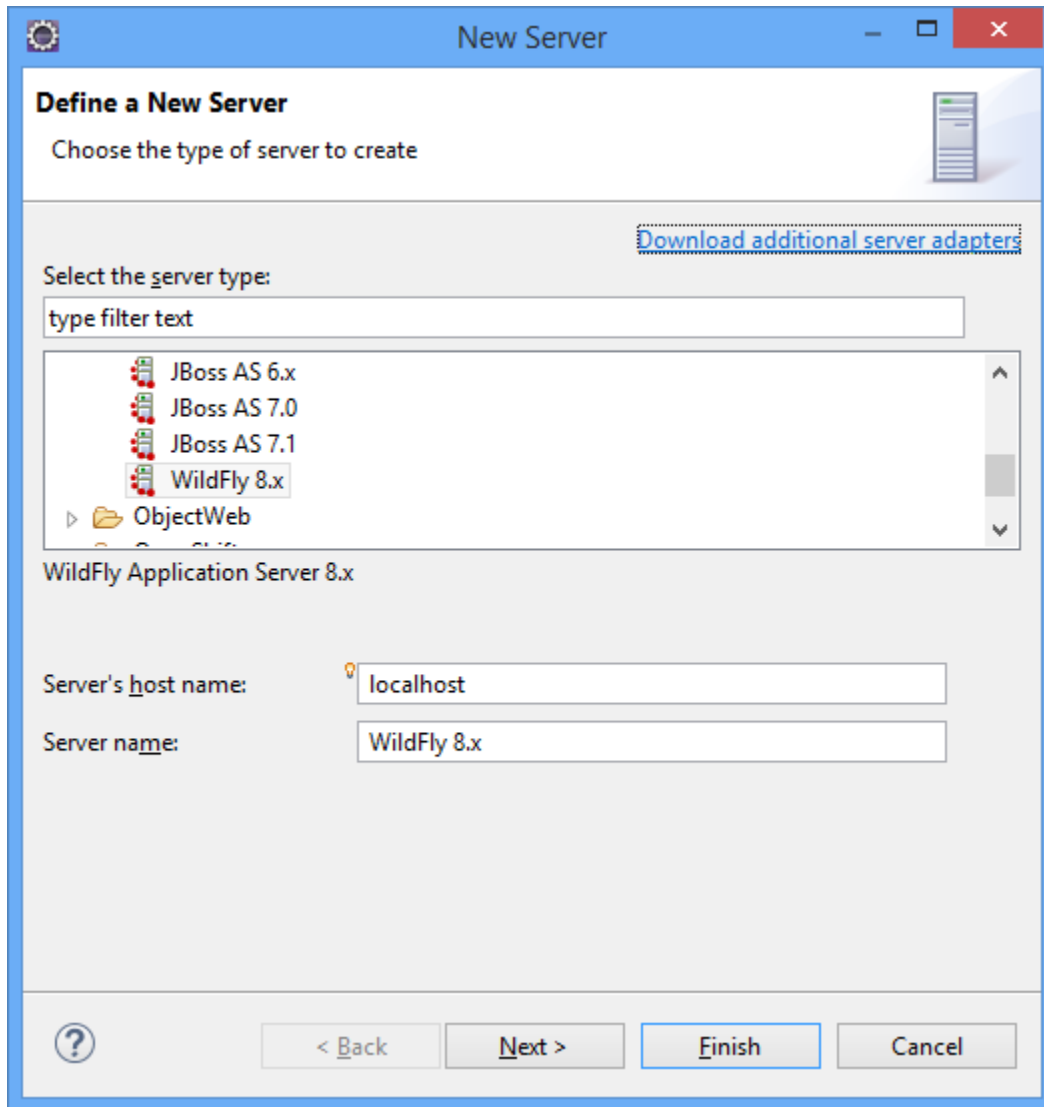
Say yes to restart Eclipse in the end.

3.9 Configure WildFly under Eclipse


Open the **Servers** tab at the bottom of the Eclipse window:



Click on the link to create a new server and select WildFly 8.x:



Click **Next** and accept the defaults:



Create a new Server Adapter
WildFly Application Server 8.x

A Server Adapter manages starting and stopping instances of your server. It manages command line arguments and keeps track of which modules have been deployed.

The server is: ☒ Local
☐ Remote

Controlled by: ☒ Filesystem and shell operations
☐ Management Operations

☐ Server is externally managed. Assume server is started.

The selected profile requires a runtime.
☒ Assign a runtime to this server

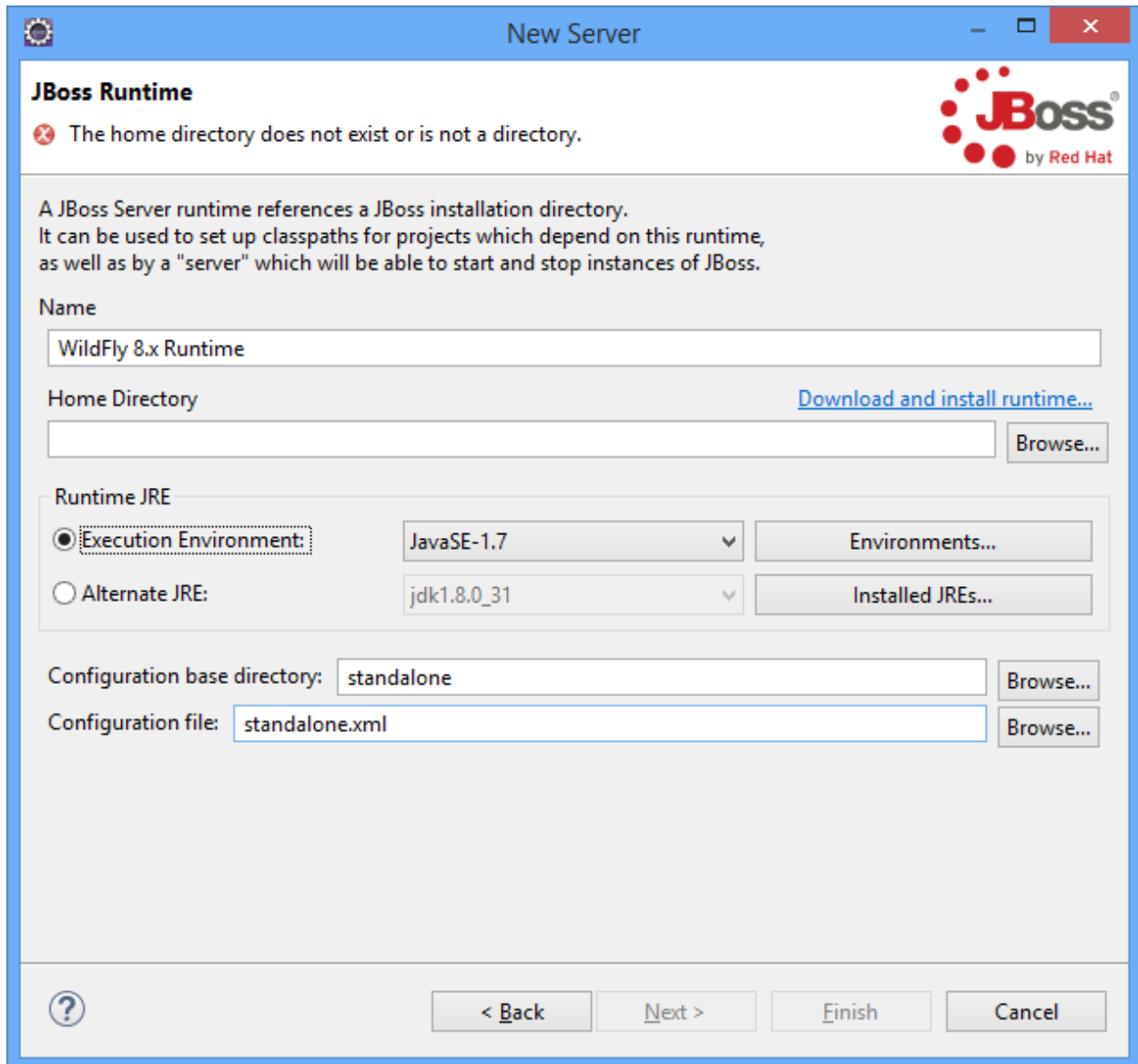
Create new runtime (next page) ▼

Runtime Details

JRE:
Home Directory:
Base Directory:
Configuration File:


? < Back Next > Finish Cancel


Click **Next**:



The image shows a 'New Server' dialog box for JBoss Runtime. It features a title bar with standard window controls. The main area is titled 'JBoss Runtime' and includes a Red Hat logo. A red error icon and message state: 'The home directory does not exist or is not a directory.' Below this, explanatory text describes the runtime's role. The configuration section includes a 'Name' field with 'WildFly 8.x Runtime', a 'Home Directory' field with a 'Browse...' button and a 'Download and install runtime...' link, and a 'Runtime JRE' section with radio buttons for 'Execution Environment:' (selected) and 'Alternate JRE:', each with a dropdown menu and an 'Environments...' or 'Installed JREs...' button. At the bottom, there are fields for 'Configuration base directory:' (set to 'standalone') and 'Configuration file:' (set to 'standalone.xml'), both with 'Browse...' buttons. The footer contains a help icon and four buttons: '< Back', 'Next >', 'Finish', and 'Cancel'.

JBoss Runtime

 **JBoss**
by Red Hat

 The home directory does not exist or is not a directory.

A JBoss Server runtime references a JBoss installation directory.
It can be used to set up classpaths for projects which depend on this runtime,
as well as by a "server" which will be able to start and stop instances of JBoss.

Name
WildFly 8.x Runtime

Home Directory [Download and install runtime...](#)
Browse...


Runtime JRE

☒ Execution Environment: JavaSE-1.7 Environments...

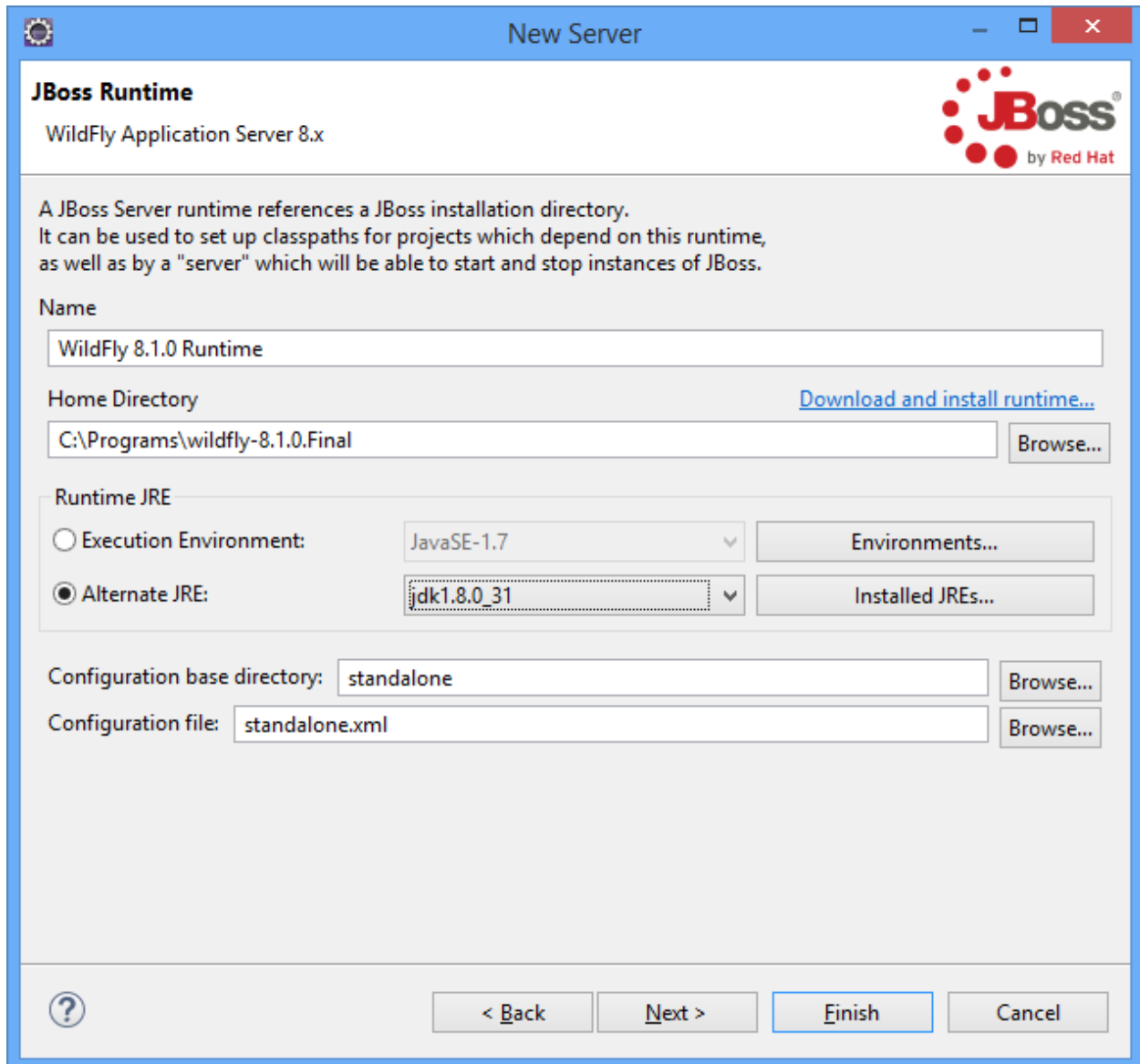
☐ Alternate JRE: jdk1.8.0_31 Installed JREs...

Configuration base directory: standalone Browse...

Configuration file: standalone.xml Browse...

 < Back Next > Finish Cancel

Click on the **Browse...** button at the **Home Directory** textbox and locate the WildFly installation directory, and switch the JRE to the installed JDK version:



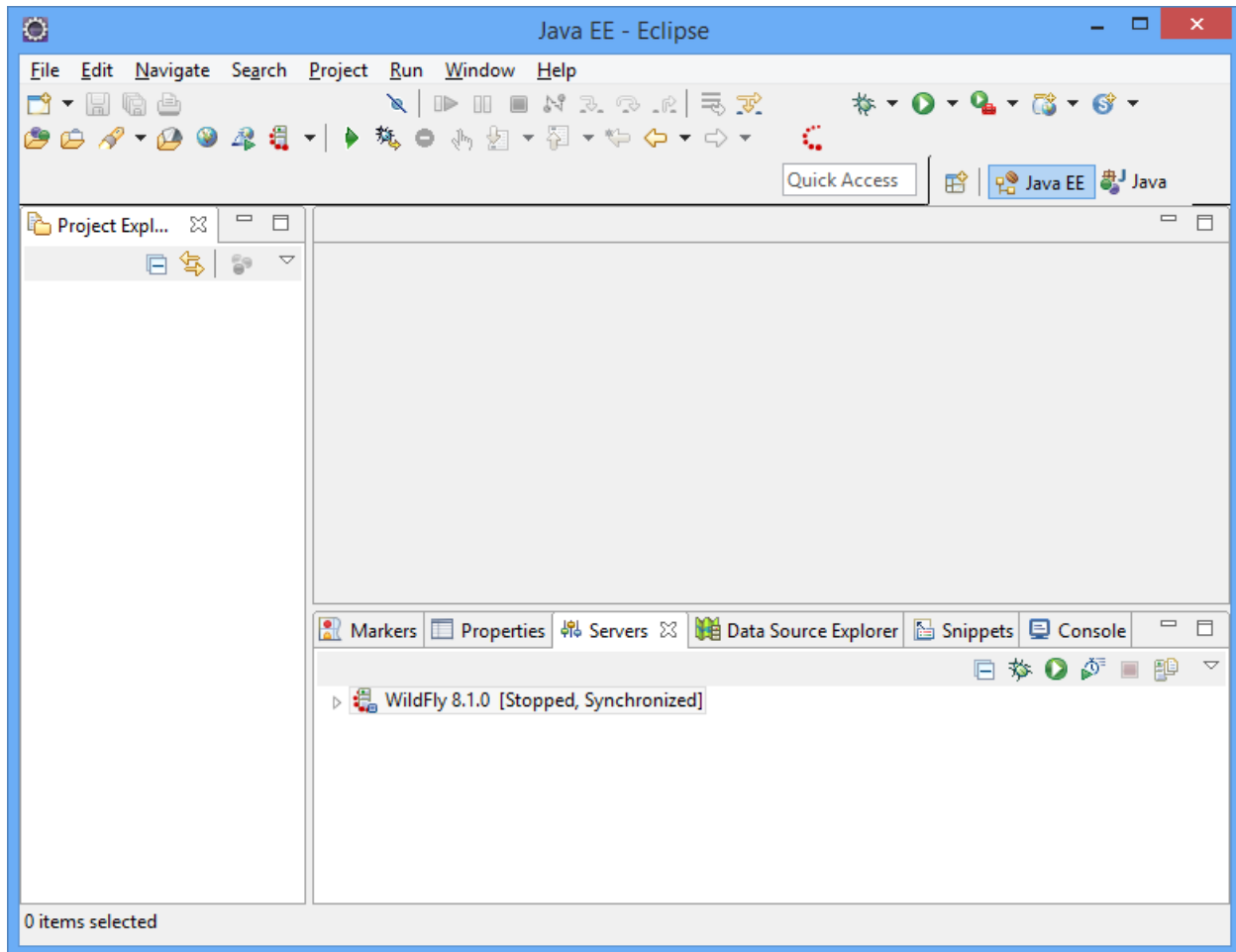
The image shows a 'New Server' dialog box for JBoss Runtime. The title bar says 'New Server'. The main header is 'JBoss Runtime' with the JBoss logo and 'WildFly Application Server 8.x'. Below this is a descriptive text: 'A JBoss Server runtime references a JBoss installation directory. It can be used to set up classpaths for projects which depend on this runtime, as well as by a "server" which will be able to start and stop instances of JBoss.'

The 'Name' field contains 'WildFly 8.1.0 Runtime'. The 'Home Directory' field contains 'C:\Programs\wildfly-8.1.0.Final' and has a 'Browse...' button next to it. A link 'Download and install runtime...' is also present. The 'Runtime JRE' section has two options: 'Execution Environment' (selected with a radio button) set to 'JavaSE-1.7' with an 'Environments...' button, and 'Alternate JRE' (selected with a radio button) set to 'jdk1.8.0_31' with an 'Installed JREs...' button. The 'Configuration base directory' field contains 'standalone' with a 'Browse...' button. The 'Configuration file' field contains 'standalone.xml' with a 'Browse...' button.

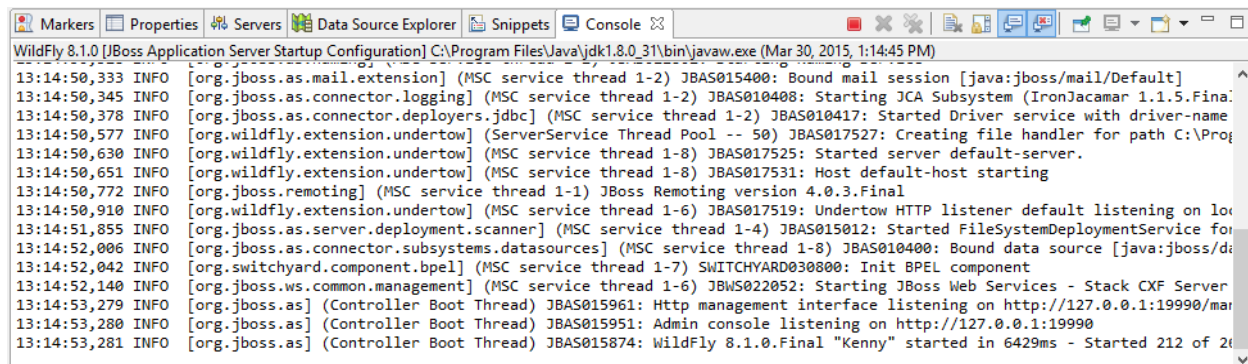
At the bottom, there is a help icon (?), and navigation buttons: '< Back', 'Next >', 'Finish' (highlighted in blue), and 'Cancel'.

Click **Next** and **Finish**.

The new server should be visible in the servers tab:

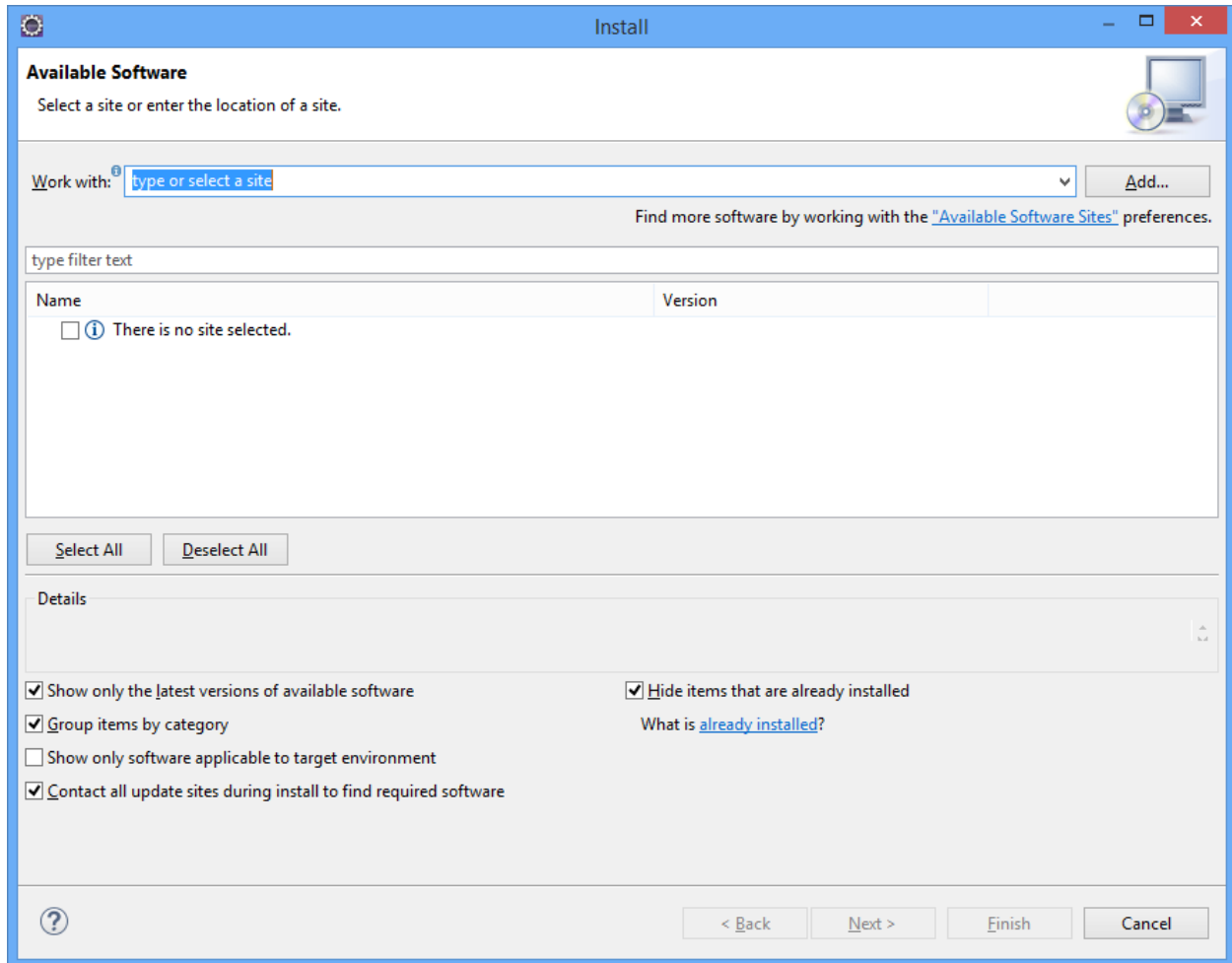


Right click on the server and start it. The server should start without errors, the logs can be seen in the Console window:



3.10 Install SwitchYard Tools

Select **Help > Install New Software...** in Eclipse:

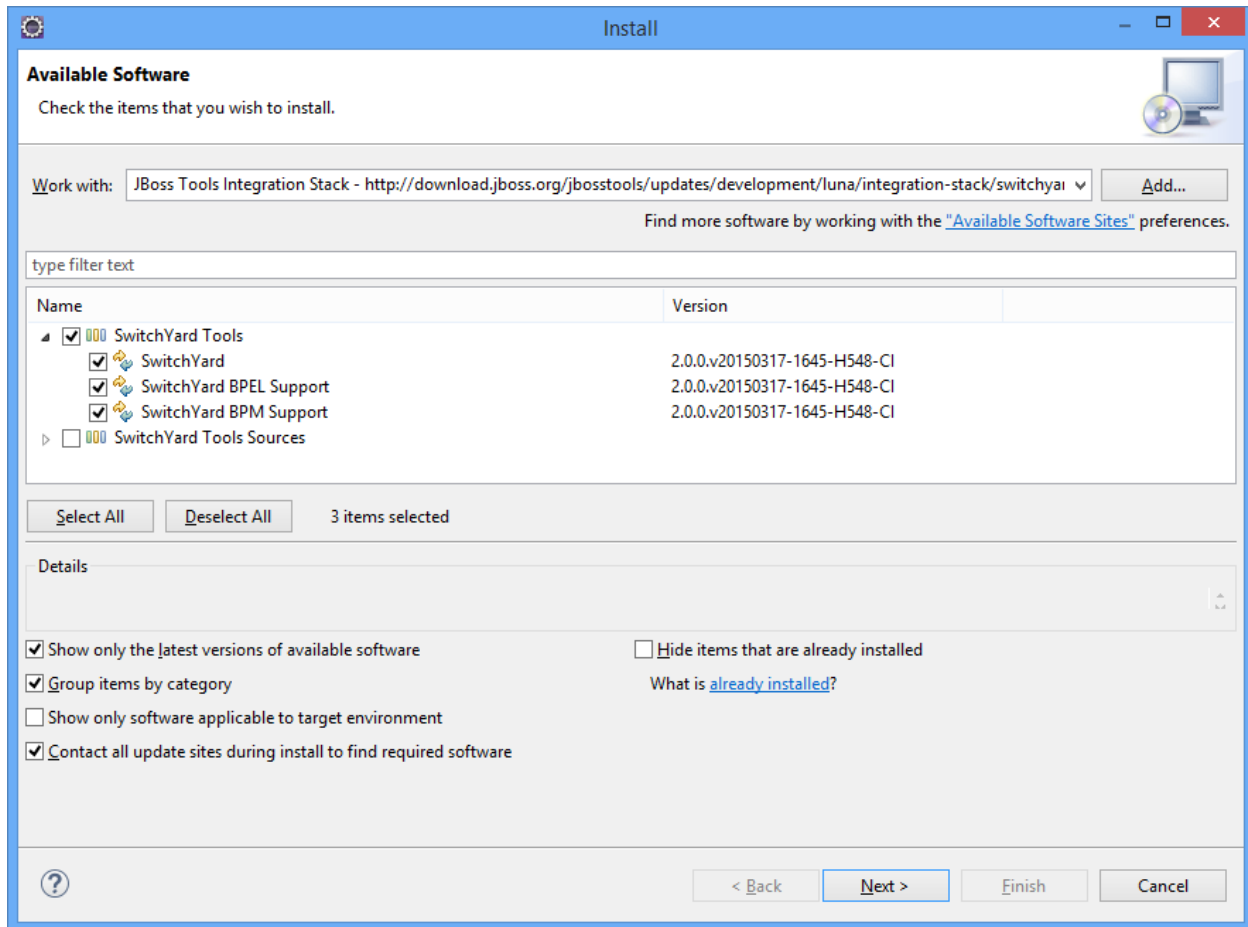


Click on **Add...** and specify the following in the **Add Repository** dialog box:

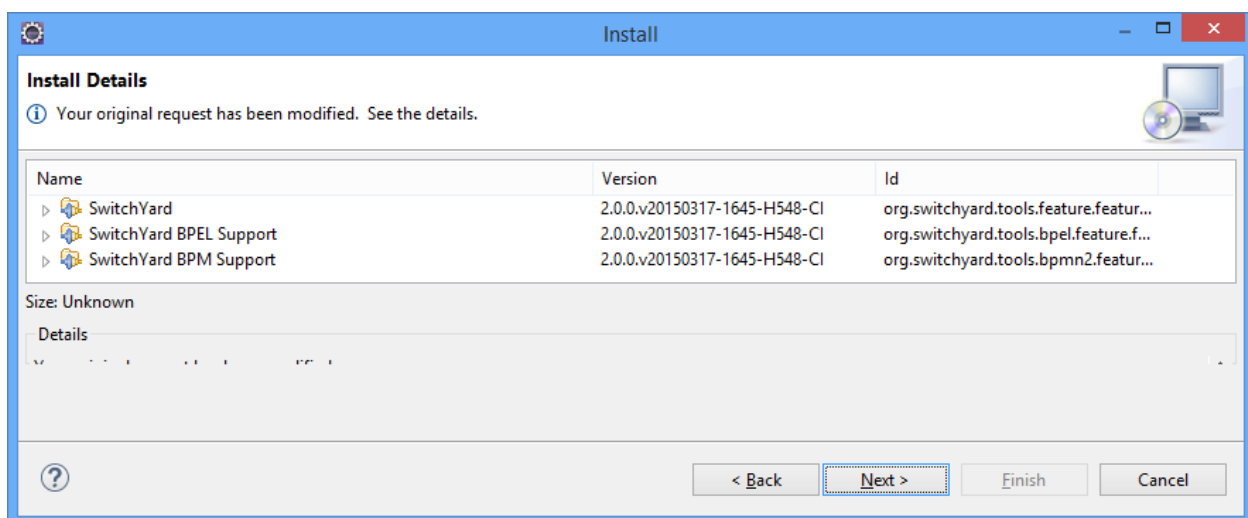
- **Name:** JBoss Tools Integration Stack
- **Location:** <http://download.jboss.org/jbosstools/updates/development/luna/integration-stack/switchyard/2.0.0.CR1/>

Click **OK**.

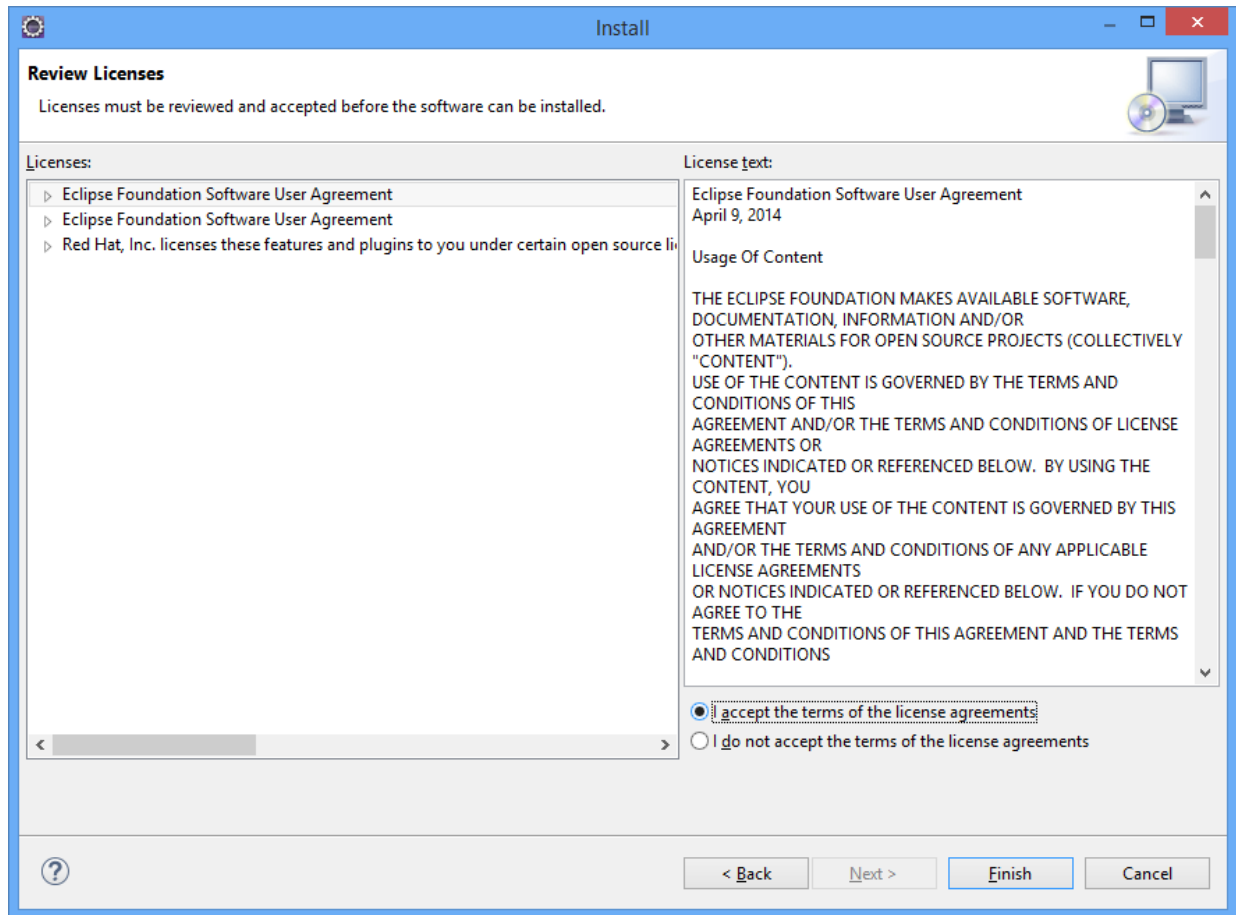
Open the **SwitchYard Tools** group and select the SwitchYard components:



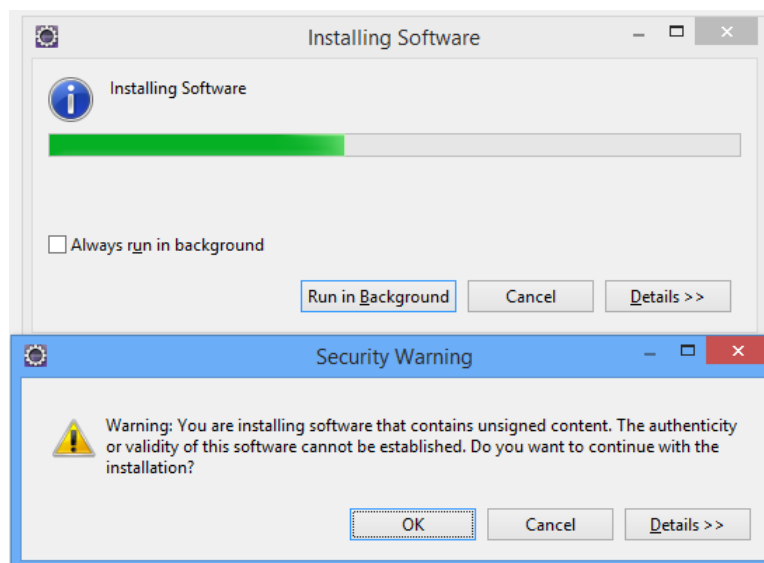
Click **Next**:



Click **Next** again, and accept the license agreement:

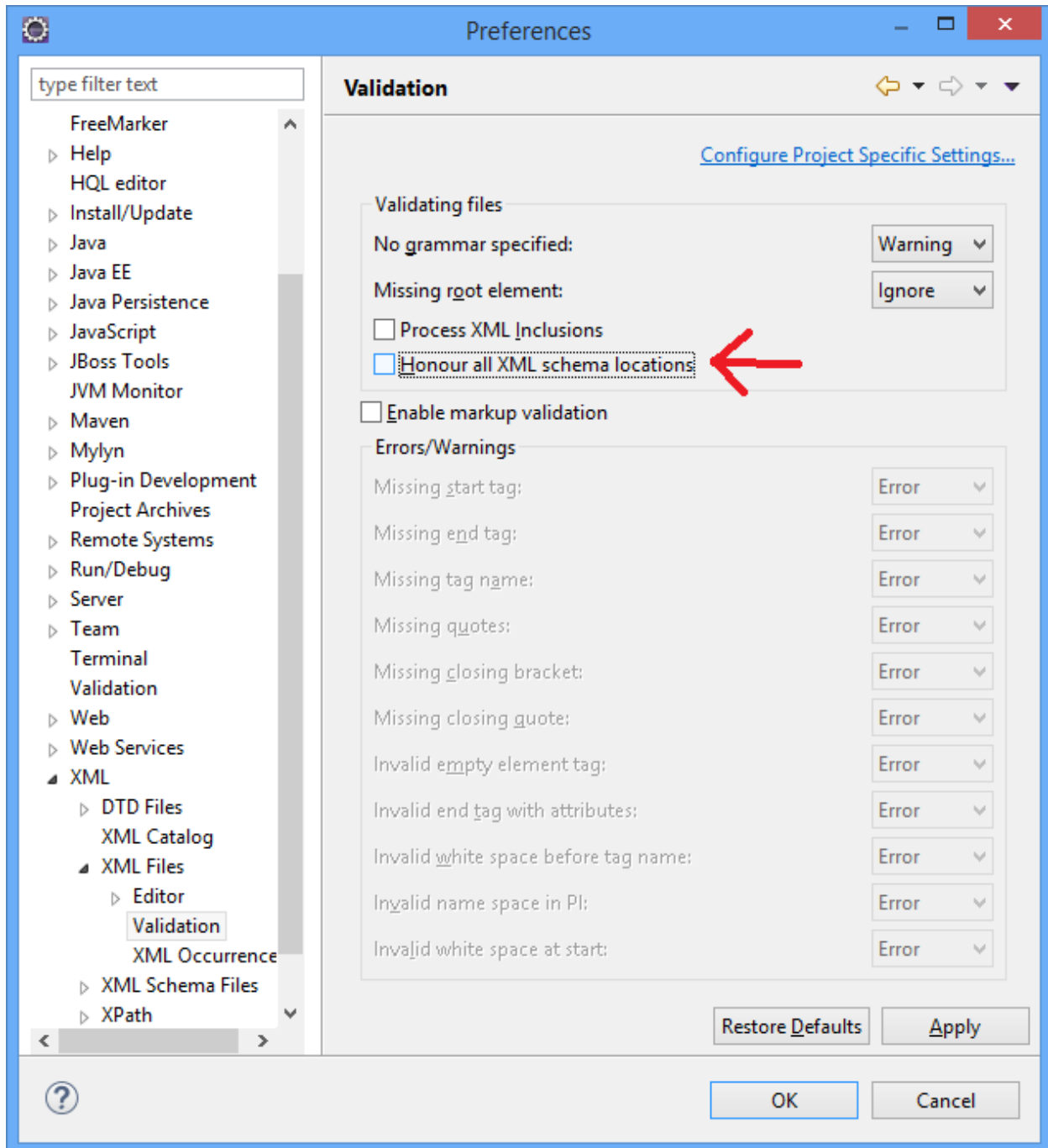


Click **Finish** and wait for the components to be installed. Accept to install unsigned components if asked:



Restart Eclipse in the end.

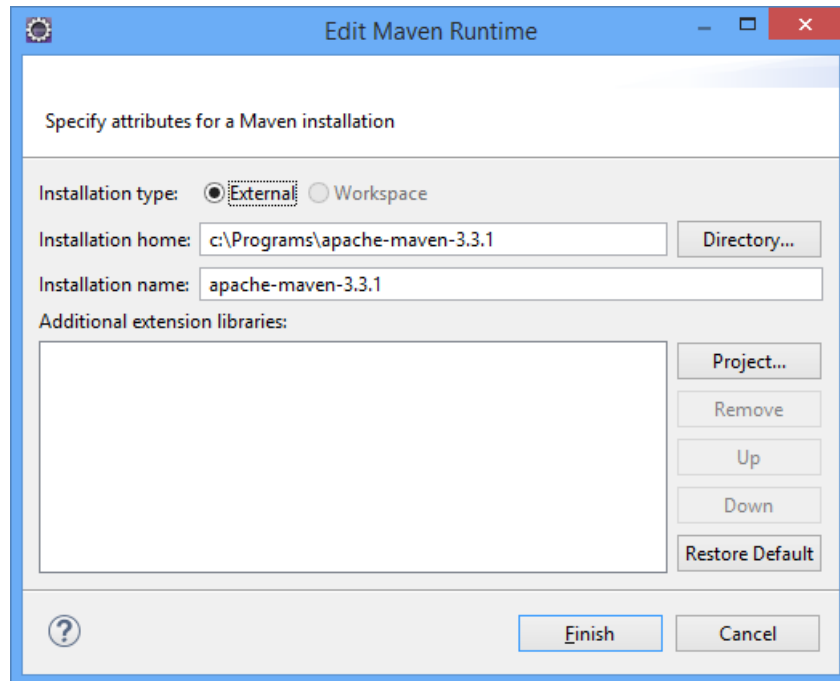
Open the **Window > Preferences** dialog and find **XML > XML Files > Validation**. Make sure to uncheck the **Honour all XML schema locations** checkbox:



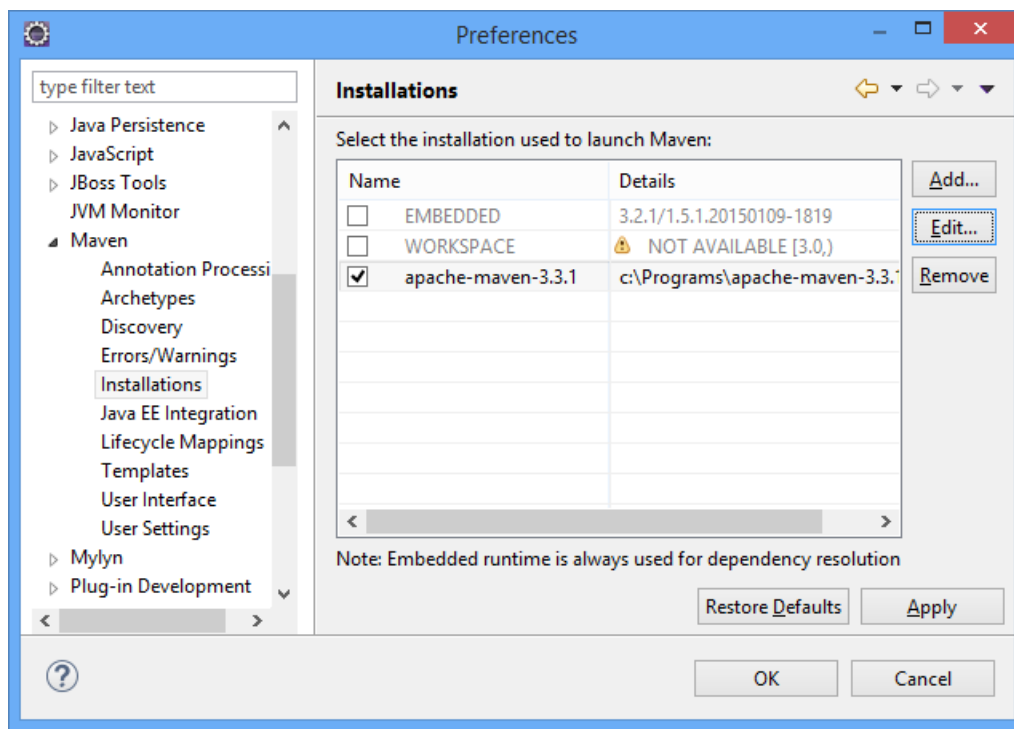
Click **OK**.

3.11 Configure Maven

Open the **Window > Preferences** dialog and find **Maven > Installations**. Click on Add... to add a new Maven installation and locate the Maven directory:



Click **Finish** and select this new installation as default:



Click **OK** to close the **Preferences** dialog.