

Eclipse 2.1.2

Tomcat 5.0 und Struts auf der Win32- Plattform integrieren

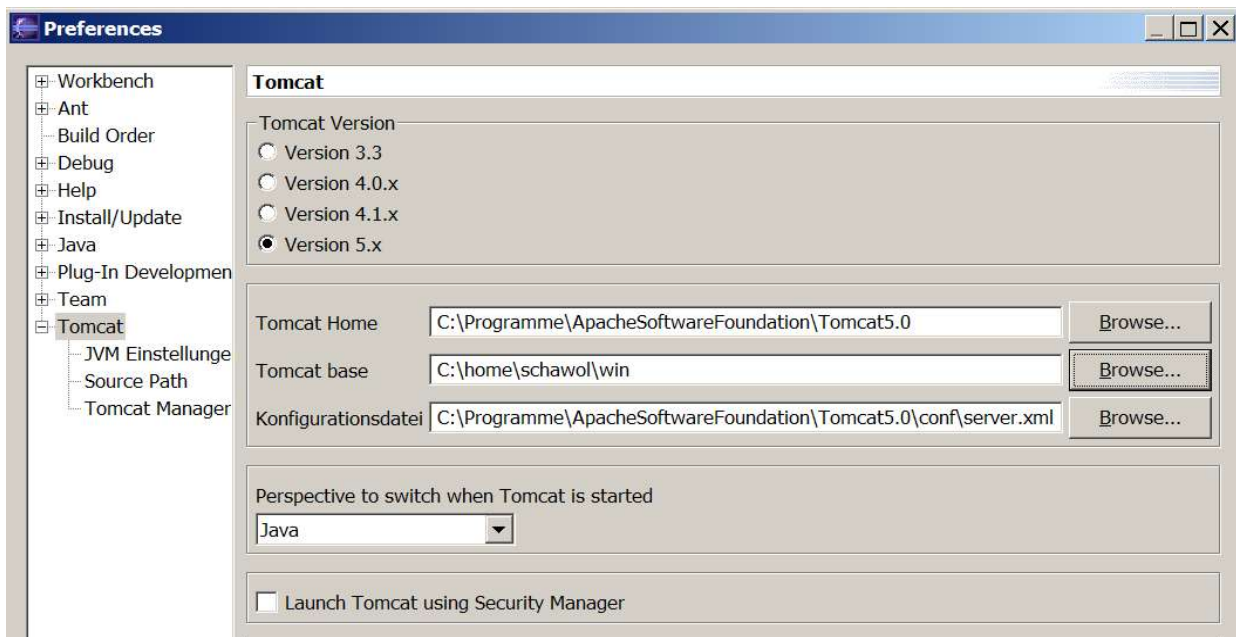
Die Open Source Entwicklungsumgebung Eclipse kann durch Installation von Plugins für viele Anwendungsbereiche erweitert werden. Im Folgenden wird die Servlet Engine Tomcat und das Framework Struts zur Entwicklung von Web-Anwendungen integriert. Die ersten drei Schritte zu einer Entwicklungsumgebung zum Erstellen von Web-Anwendungen sind das Installieren von Eclipse, Tomcat und dem Eclipse Tomcat Plugin. Diese Schritte werden sehr kurz beschrieben, so hierzu bereits genügend Dokumentation zu finden ist. Mehr Details finden Sie zur Verwendung des Tomcat Plugins und zur Integration des Frameworks Struts.

1. Eclipse Entwicklungsumgebung installieren (<http://www.eclipse.org/downloads/index.php>) Das Zipfile eclipse-SDK-[Version]-win32.zip ist die richtige Version der Entwicklungsumgebung. Nach dem Download reicht es aus das Paket zu entpacken und danach das Programm eclipse.exe aufrufen. Dadurch wird die Installation abgeschlossen.
2. Tomcat 5.0 installieren (<http://jakarta.apache.org/site/binindex.cgi>)
3. Tomcat Plugin für Eclipse von sysdeo installieren (<http://www.sysdeo.com/eclipse/tomcatPlugin.html>) Auf dieser Website nach unten scrollen und tomcatPluginV221.zip (Stand: 05.01.2004) herunterladen. Um das Plugin zu installieren genügt es die Datei in das Verzeichnis <Eclipse-Home>/plugins auszupacken.

Falls es unter dem Punkt Window-> Preferenes auf der linken Seite einen Punkt Tomcat gibt, war die Installtion erfolgreich.

Vervollständigen der Installation:

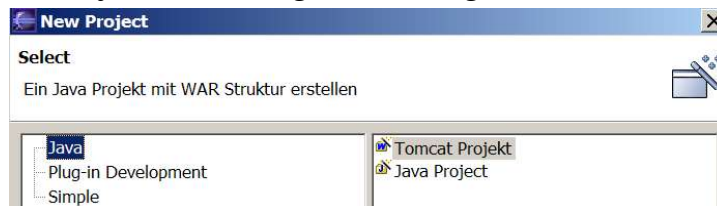
Folgende Einstellungen müssen vorgenommen werden



- Festlegen der verwendeten Version von Tomcat
- Einstellen des Verzeichnisses, in dem Tomcat installiert ist

Bei Bedarf kann angegeben werden in welche Perspektive gewechselt werden soll, wenn Tomcat gestartet wird.

Nach diesen Einstellungen kann das erste Tomcat-Projekt erstellt werden.
Der Punkt File->New->Project führt in folgenden Dialog:



Mit dem Assistenten kann der Projektname, die URL der Anwendung und die „Context Root“ des Projekts angegeben werden. Für einen ersten Test genügt es, die Angabe des Projektname mit dem Button „Finish“ zu bestätigen. Die URL der Anwendung entspricht damit dem Projektname.

Jetzt muss das angelegte Projekt der Tomcat Servlet Engine bekannt gemacht werden. Dazu markiert man das Projekt, wählt aus dem Kontextmenü „Tomcat Project“, dem untergeordneten Contextmenü den Punkt „Context in server.xml einfügen“ aus. Dadurch wird die URL der Anwendung in die Datei server.xml im Verzeichnis <TomcatRoot>/conf/ eingetragen. Zusätzlich dazu wird die docBase und das workDir auf die entsprechenden Unterverzeichnisse im Eclipse Workspace gesetzt. Jetzt ist dem Tomcat die URL der Anwendung bekannt und mit den Verzeichnissen im Eclipse Workspace verknüpft.

Der generierte Eintrag sieht wie folgt aus:

```
<Context path="/[AppURL]" reloadable="true" docBase="[Eclipse-Workspace][Projektname]"  
workDir="[Eclipse-Workspace][Projektname]work" />
```

Test des Plugins durch Ansprechen eines selbst erstellten Servlets.

Anlegen des Servlets com.iternum.test.TestSrv.

Anklicken des Ordners WEB-INF/src und Auswahl von New->Pakage in den entsprechenden Contextmenüs. Danach den Ordner WEB-INF/src öffnen, das Paket com.iternum.test anklicken und durch Auswahl von new->Class in den entsprechenden Contextmenüs eine neue Klasse mit dem Namen TestSrv erzeugen. Das Servlet TestSrv enthält folgenden Quellcode:

```
package com.iternum.test;  
  
import javax.servlet.http.HttpServlet;  
import java.io.PrintWriter;  
  
public class TestSrv extends HttpServlet {  
  
    public void doPost(  
        javax.servlet.http.HttpServletRequest request,  
        javax.servlet.http.HttpServletResponse response)  
        throws javax.servlet.ServletException, java.io.IOException {  
        doWork(request, response);  
    }  
    public void doGet(  

```

```
        javax.servlet.http.HttpServletRequest request,
        javax.servlet.http.HttpServletResponse response)
        throws javax.servlet.ServletException, java.io.IOException {
            doWork(request, response);
        }
    public void doWork(
        javax.servlet.http.HttpServletRequest request,
        javax.servlet.http.HttpServletResponse response)
        throws javax.servlet.ServletException, java.io.IOException {
        PrintWriter out = response.getWriter();
        out.println("<html><head><title>meine erste Seite</title></head><body>");
        out.println("hallo Welt ...");
        out.println("</body></html>");
    }
}
```

Um das Servlet im entsprechenden Webcontent anzusprechen, muss eine Datei mit dem Namen web.xml erstellt werden in der das Servlet angemeldet wird. Die Datei web.xml muss im Verzeichnis <Eclipse-Workspace><Projektname>WEB-INF erzeugt werden und mindestens folgenden Inhalt haben:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE web-app
    PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"
    "http://java.sun.com/j2ee/dtds/web-app_2_2.dtd">
<web-app>
    <display-name>theFirst</display-name>

    <!-- Servletname einer Servletklasse zuordnen -->
    <servlet>
        <servlet-name>TestSrv</servlet-name>
        <servlet-class>com.iternum.test.TestSrv</servlet-class>
    </servlet>

    <!-- URL einem Servletnamen zuordnen -->
    <servlet-mapping>
        <servlet-name>TestSrv</servlet-name>
        <url-pattern>Test</url-pattern>
    </servlet-mapping>

    <!-- Dateien, nach denen beim Start der Web-Anwendung gesucht werden soll.-->
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>
</web-app>
```

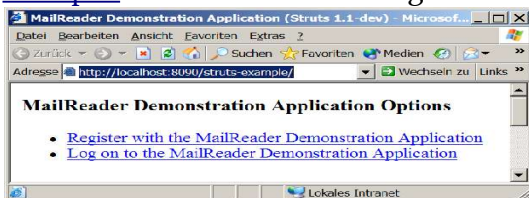
Nach dem Speichern aller veränderten Dateien und dem Neustart von Tomcat sollte die Eingabe der URL <http://localhost:8090/theFirst/Test> folgendes Ergebnis liefern. (Anmerkung: Falls Sie den Standard Port von Tomcat benutzen, so geben Sie statt 8090 bitte 8080 ein!)



Damit ist die Integration von Tomcat in die Entwicklungsumgebung eclipse abgeschlossen. Jetzt kann die Bibliothek struts und das Plugin zur grafischen Darstellung der Datei struts-config.xml eingebunden werden.

1. Download der Bibliothek struts (<http://jakarta.apache.org/site/binindex.cgi>) Der Link 1.1 zip führt zur Datei jakart-struts-1.1.zip (Stand: 05.01.2004).
2. Entpacken der Datei.
3. Inhalt des Unterverzeichnisses webapps in das Verzeichnis <Tomcathome>\webapps kopieren.

Als ersten Test der Installation der Beispiele des Struts Frameworkes unter der Standalone Version der Tomcat Servlet Engine kann folgende URL eingegeben werden: <http://localhost:8090/struts-example/>. Erscheint die nachfolgende Seite, wurden die Beispiele erfolgreich installiert.



Hintergrund

Die Tomcat Servlet Engine packt beim Start alle Dateien mit der Endung “.war“ in ein Unterverzeichniss unter webapps aus. In den Unterverzeichnisses, die mit „struts-“ beginnen befinden sich nach dem Entpacken alle notwendigen Bibliotheken (*.jar) und Type-Libraries, die vom Framework benötigt werden. Die Beispielverzeichnisse können so als Vorlage für den Inhalt des Projektverzeichnisses innerhalb von Eclipse verwendet werden.

Achtung: Will man das erstellte TestServlet weiterhin benutzen, so muss der Eclipse Workspace vor den nächsten Schritten gesichert werden, denn während der Integration von struts wird die Datei web.xml überschrieben!

Installation des Plugins zur grafischen Darstellung der Datei struts-config.xml

Vorraussetzung für den Editor und das Plugin ist die Installation der Plugins

- org.eclipse.draw2d
- net.sf.solareclipse.xml.ui

Die Bibliothek org.eclipse.draw2d kann man unter <http://www.eclipse.org/gef/> herunterladen. Nach dem Download muss die Datei GEF-runtime-I20031015.zip in das Verzeichnis <Eclipse-Home> entpackt werden.

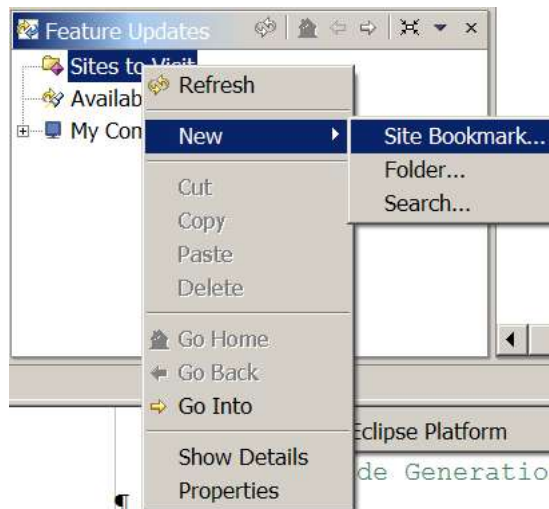
Das Plugin net.sf.solareclipse.xml.ui findet sich auf der Sourceforge Site unter <http://sourceforge.net/projects/solareclipse/>. Dort kann man die Datei net.sf.solareclipse_0.4.0.bin.dist.zip (Stand: 07.01.2004) herunterladen und im Verzeichnis <Eclipse-Home> entpacken.

Nach der Installation der beiden Plugins kann das Struts-Plugin installiert werden.

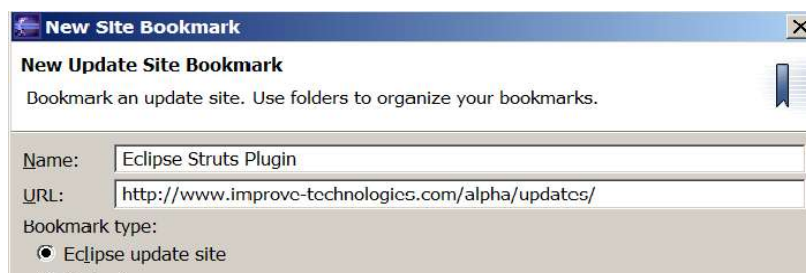
Software Manager so konfigurieren, dass die Seite <http://www.improve-technologies.com/alpha/struts-config-editor/> als Ressource für Plugins erkannt wird.



Im Resource View „Sites to visit“ markieren und in den Contextmenüs folgende Punkte auswählen:

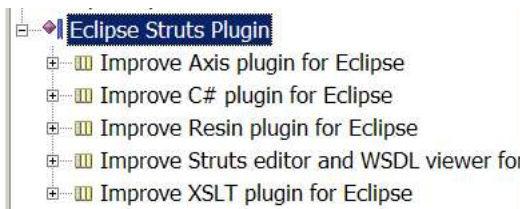


Folgende Einstellungen sind vorzunehmen



Sind die Einstellungen korrekt eingetragen worden, so findet man im linken Bereich des Update

Managers folgende Einträge:



Den Download des Plugins startet man dadurch, dass man den Punkt „Improve Struts editor and WSDL viewer for Eclipse“ aufklappt und die aktuelle Version (Struts editor and WSDL viewer 0.4.1, Stand 07.01.2004) auswählt. Danach auf der rechten Seite „download now“ auswählen und das Copyright bestätigen.

Test des Plugins durch Kopieren der Webapplikation struts-blank in das aktuelle Projektverzeichnis.

Folgende Dateien müssen kopiert werden:

<Tomcat-Home>\webapp\struts-blank\pages nach <Eclipse-Workspace>\projectname\pages

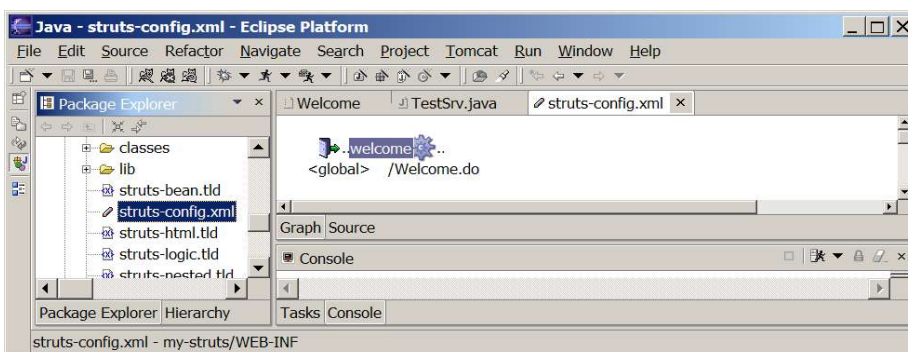
<Tomcat-Home>\webapp\struts-blank\WEB-INF\classes\ressources nach

<Eclipse-Workspace>\projectname\WEB-INF\classes\ressources

<Tomcat-Home>\webapp\struts-blank\lib nach <Eclipse-Workspace>\projectname\lib

Alle Dateien im Verzeichnis <Tomcat-Home>\webapp\struts-blank\WEB-INF nach <Eclipse-Workspace>\projectname\WEB-INF

Nach einem Refresh des aktuellen Projekts kann die Datei struts-config.xml grafisch angezeigt werden.



Wird Tomcat gestartet, so kann die Webapplikation struts-blank auch unter der Adresse

<http://localhost:8090/<Eclipse-Projektname>/> erreicht werden.

Erscheint nach Eintippen der URL folgende Seite, so hat die Integration von Struts mit Eclipse und Tomcat funktioniert.

