

# Tomcat Installation & Konfiguration

Contributed by Christian Wettinger  
Wednesday, 07 July 2004  
Last Updated Friday, 24 October 2008

Durch meine Arbeit bei der Firma Samhammer AG in Weiden bin ich nun auch mit der Programmierung in JAVA in Berührung gekommen. Nachfolgend ist die Windows-Installation erklärt, damit ein Mercedesfahrer auch etwas damit anfangen kann.

- Download "Java Development Kit"

Entwicklungswerkzeuge zum Programmieren von Java-Anwendungen, enthält neben der Java-Laufzeitumgebung (JRE) den Java-Compiler javac sowie javadoc, jar, jarsigner und einen Appletviewer. Download

Die Installation erfolgte nach C:\Programme\Java\

- Download von Apache Tomcat 5.5

Apache Tomcat stellt eine Umgebung zur Ausführung von Java-Code auf Webservern bereit.  
tomcat.apache.org

(In diesem Fall habe ich die Version 5.5 gewählt)

- Installation von Tomcat

Das ZIP-File entpacken. Z.B.:

C:\apache-tomcat-5.5.26

- Setzen der Umgebungsvariable

Unter Windows erfolgt dies unter Einstellungen --> Systemsteuerung --> System --> Erweitert --> Umgebungsvariablen

CATALINA\_HOME = C:\apache-tomcat-5.5.26

JAVA\_HOME = C:\Programme\Java\jdk1.6.0\_06

- Kurztest durchführen

Über die Konsole starten wir zuerst den Tomcat (C:\apache-tomcat-5.5.26\bin\startup.bat).

Danach kann man über den Browser und der Url: <http://localhost:8080> auf den Tomcat zugreifen.

Sollte beim Start des Tomcats über die Kommandozeile nachfolgender Fehler auftauchen, so kann man die benötigte Umgebungsvariable auch direkt am Beginn der startup.bat setzen. Dazu vermerkt man am Anfang des genannten Files "set JAVA\_HOME=C:\Programme\Java\jdk1.6.0\_06".

Fehlermeldung: Neither the JAVA\_HOME nor the JRE\_HOME environment variable is defined

- Tomcat-Administration

Stoppen Sie Tomcat und fügen Sie in  
'C:\apache-tomcat-5.5.26\conf\tomcat-users.xml'  
folgende Zeilen nach '<tomcat-users>' ein:

```
<role rolename="admin"/>
```

```
<role rolename="manager"/>
```

```
<user username="MeinName" password="MeinPasswort" roles="admin,manager"/>
```

Wenn Sie jetzt Tomcat neu starten, können Sie Tomcat über  
<http://localhost:8080/manager/html> administrieren

- Standard Port ändern

Falls man anstelle des Tomcat Standard-Ports 8080 lieber den Port 80 hätte, kann man das in der Datei C:\apache-tomcat-5.5.26\conf\server.xml anpassen:

```
<!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->
```

```
<Connector port="8080" maxHttpHeaderSize="8192"
```

```
    maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
```

```
    enableLookups="false" redirectPort="8443" acceptCount="100"
```

```
    connectionTimeout="20000" disableUploadTimeout="true" />
```

Im Connector-Tag muss der Port-Parameter auf 80 abgeändert werden, dann kann man den Tomcat auch ganz normal über <http://localhost> erreichen. ACHTUNG: Der Server muss aber vorher neu gestartet werden!

- ROOT Context ändern

Jede Anwendung von Tomcat ist innerhalb eines Context organisiert. Ein Context fasst alle zusammengehörigen Java-Klassen, JSP-Seiten usw. einer Anwendung zusammen. Ein Tomcat kann eine Vielzahl von Contexten enthalten, alle sind aber voneinander separiert und können ihre eigenen, individuellen Einstellungen definieren. Das bedeutet gleichfalls, dass es unter Tomcat keinen zentralen Document-Root gibt. Stattdessen muss in der server.xml ein entsprechender Context-Eintrag definiert werden, der das Basis-Verzeichnis, den Context-Root, definiert.

In der Konfigurationsdatei %CATALINA\_HOME%\conf\server.xml die

```
auskommentierte Einstellung <Context path="" docBase="ROOT" debug="0"/>
```

aktivieren. Dieses Element befindet sich ziemlich am Ende von server.xml.

In Version 5 ist das Element fehlerhaft (<.../> der Slash fehlt).

- Invoker Servlet aktivieren

Im Fall des Standard-Servlet-Mappings wird ein Servlet unter der URI `servlet/SERVLETNAME` zugänglich. Um diese

Möglichkeit einzuschalten, müssen wir die globale Konfigurationsdatei `conf/web.xml` modifizieren und an zwei Stellen das so genannte Invoker-Servlet aktivieren.

In der Konfigurationsdatei `C:\apache-tomcat-5.5.26\conf\web.xml` die auskommentierte

Einstellung

```
<!-- The mapping for the invoker servlet -->
```

```
<!--
```

```
<servlet-mapping>
```

```
<servlet-name>invoker</servlet-name>
```

```
<url-pattern>/servlet/*</url-pattern>
```

```
</servlet-mapping>
```

```
-->
```

aktivieren (beachten Sie, dass es mehrere Mappings gibt. Sie müssen jene mit \* aktivieren).

In Tomcat 5 müssen Sie zudem den in `web.xml` weiter oben stehenden Invoker Servlet Teil aktivieren, sonst wird das Servlet nicht gefunden:

```
<!-- The "invoker" servlet, which executes anonymous servlet classes -->
```

```
<!-- that have not been defined in a web.xml file. Traditionally, this -->
```

```
<!-- servlet is mapped to the URL pattern "/servlet/*", but you can map -->
```

```
<!-- it to other patterns as well. The extra path info portion of such a -->
```

```
<!-- request must be the fully qualified class name of a Java class that -->
```

```
<!-- implements Servlet (or extends HttpServlet), or the servlet name -->
```

```
<!-- of an existing servlet definition. This servlet supports the -->
```

```
<!-- following initialization parameters (default values are in square -->
```

```
<!-- brackets): -->
```

```
<!-- -->
```

```
<!-- debug Debugging detail level for messages logged -->
```

```
<!-- by this servlet. [0] -->
```

```
<!--
```

```
<servlet>
```

```
<servlet-name>invoker</servlet-name>

<servlet-class>

  org.apache.catalina.servlets.InvokerServlet

</servlet-class>

<init-param>

  <param-name>debug</param-name>

  <param-value>0</param-value>

</init-param>

<load-on-startup>2</load-on-startup>

</servlet>
```

-->

#### - Servlet Reloading aktivieren

Der Prozeß des "Servlet Reloading" eröffnet die Möglichkeit, das Servlet neu zu kompilieren und zu testen, ohne das der Server neu gestartet werden muß. Dies ist vor allem in der Entwicklungsphase wichtig, aber auch wenn eine neue Version eines Servlets benutzt werden soll, da der Betrieb nicht unterbrochen werden muß.