Table of Contents

Für die Integration von JVx in einen Applikationsserver ist die Konfiguration des Deployment-Deskriptor erforderlich.

Der Deployment-Desktriptor ist im Normalfall eine XML Datei mit der Bezeichnung web.xml. Für z.B. Apache Tomcat muss die Datei wie folgt aufgebaut werden:

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xmlns="http://java.sun.com/xml/ns/javaee"
         xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
         xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
         id="WebApp_ID"
         version="2.5">

  <!--
**********************************************************************
***
    Context configuration
**********************************************************************
***
   -->

  <display-name>Application name</display-name>

  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.jsp</welcome-file>
  </welcome-file-list>

  <!--
**********************************************************************
***
    JVx configuration
**********************************************************************
***
   -->

  <!-- Servlet for accessing the JVx server through http(s) streaming
protocol. -->
  <servlet>
    <servlet-name>ServletServer</servlet-name>
    <servlet-
class>com.sibvisions.rad.server.http.ServletServer</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>ServletServer</servlet-name>
```

```
    <url-pattern>/services/Server</url-pattern>
  </servlet-mapping>

  <!-- Servlet for downloading content (reports, csv export, any binary
data). -->
  <servlet>
    <servlet-name>DownloadServlet</servlet-name>
    <servlet-
class>com.sibvisions.rad.server.http.DownloadServlet</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>DownloadServlet</servlet-name>
    <url-pattern>/services/Download</url-pattern>
  </servlet-mapping>

  <!-- Servlet for uploading content (images, import files, any binary
data). -->
  <servlet>
    <servlet-name>UploadServlet</servlet-name>
    <servlet-
class>com.sibvisions.rad.server.http.UploadServlet</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>UploadServlet</servlet-name>
    <url-pattern>/services/Upload</url-pattern>
  </servlet-mapping>

  <!-- Online Help integration. -->
  <servlet>
    <servlet-name>Help</servlet-name>
    <servlet-
class>com.sibvisions.rad.help.server.HelpServiceImpl</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>Help</servlet-name>
    <url-pattern>/help/services/Help</url-pattern>
  </servlet-mapping>

  <!--
************************************************************************
***
    Special mime-type mappings
************************************************************************
***
   -->

  <!-- Mime type for csv export. -->
  <mime-mapping>
```

```
    <extension>csv</extension>
    <mime-type>application/vnd.ms-excel</mime-type>
  </mime-mapping>

</web-app>
```

Wie bereits im Listing erkennbar, beschränkt sich die Konfiguration auf ein Minimum. Abhängig von den gewünschten Client Optionen müssen die notwendigen Server Objekte verwendet werden.

- Für die Kommunikation zwischen Client und Server über das http(s) Protokoll wird der ServletServer verwendet
- Wenn der Client Daten zum Server Up- oder vom Server Downloaden muss, wird zusätzlich das UploadServlet und DownloadServlet benötigt
- Der HelpService findet Verwendung wenn eine dynamische Online Hilfe angeboten wird

From:
https://doc.sibvisions.com/ - **Documentation**

Permanent link:
https://doc.sibvisions.com/de/jvx/common/setup/deployment_descriptor

Last update: **2018/02/02 12:53**