developerWorks®

Running your IBM i web solution on ASF Tomcat

Yang Cheng Peng
Tian Gang
Zheng Chang Qing

January 21, 2013

Apache Software Foundation (ASF) Tomcat and HTTP Server for i are both widely used web servers. They can be bundled together to take full use of their own advantages to run web solutions on IBM® i. This article introduces how to bring Tomcat to IBM i, associate with HTTP Server for i, and run a web application in the web environment.

ASF Tomcat (in short, referred to as *Tomcat* in this article) is an open source web server and servlet container developed by Apache Software Foundation (ASF). Tomcat runs with great performance and has been a collaboration of the best-of-breed developers from around the world. Comparing to the previous version, Tomcat V7 implements the Servlet 3.0 and Java™ Server Pages (JSP) 2.2 specifications and offers a lot of enhancements for most of the operating systems, including IBM i. A number of years ago, IBM i actually included Tomcat on IBM i. The last version that IBM shipped was version 3.2.4. What we are covering in this article is not this support. One of the major reasons for not including Tomcat in IBM i is that Tomcat is a pure Java application and can be quickly and easily downloaded and started on IBM i with little to no effort as you will see in this article.

The IBM HTTP Server for i is a web server implementation that is based on the open-source server code provided by the Apache Software Foundation and that is optimized for the IBM i environment. HTTP Server for i on IBM i 7.1 is based on Apache 2.2.11 and all HTTP Server related security vulnerabilities (known as common vulnerabilities and exposures [CVE]) are patched with HTTP Server program temporary fixes (PTFs) to ensure PCI compliance. The IBM i team has been including an IBM i version of the Apache HTTP server for a long time. Since the HTTP server is built on C, the binaries need to be created especially for IBM i to use the advantages that the IBM i OS brings. This support has been part of the DG1 product since 2000.

It is a good solution to associate the two technologies, Tomcat and HTTP Server for i, together to take full use of their advantage with IBM i platform edge. This article considers IBM i 7.1 and Tomcat 7.0.28 as examples. This same support would run equally as well on IBM i 6.1. You would just have to substitute the correct IBM i 6.1 entity.

The Tomcat server can be connected to a local or remote HTTP Server on IBM i. Figure 1 and Figure 2 illustrate the different scenarios respectively.

Figure 1. Tomcat and HTTP Server for i on the same IBM i system (local mode)

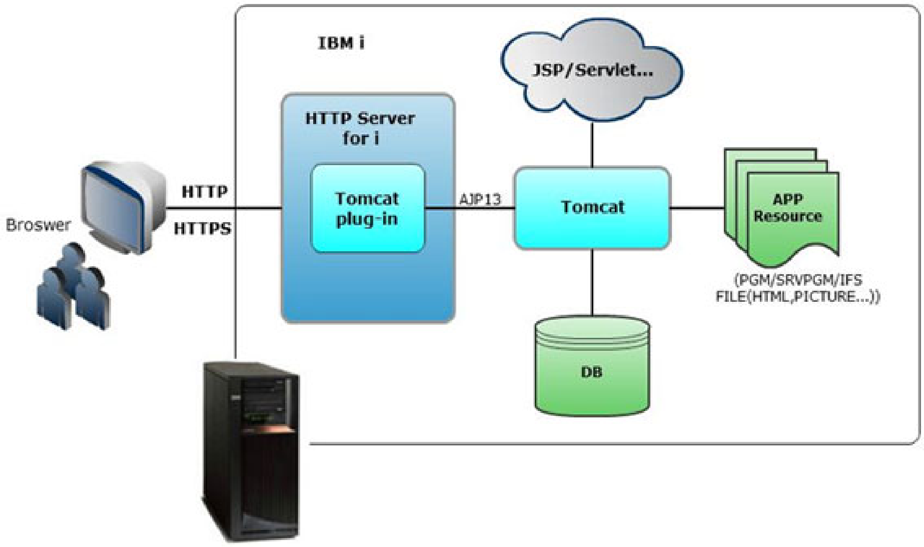
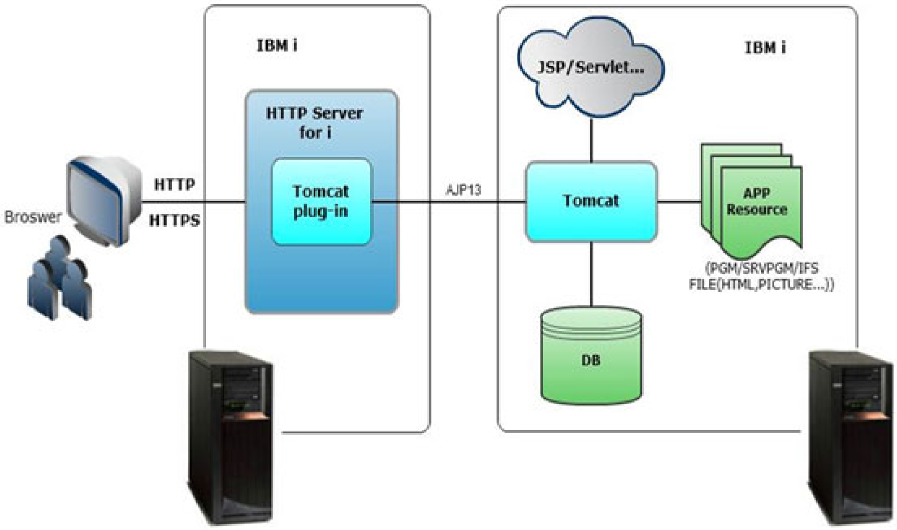


Figure 2. Tomcat and HTTP Server for i on different IBM i systems (remote mode)



Software prerequisites

The previous section provided a basic introduction of Tomcat and HTTP Server for i. In order to get your web applications successfully running on Tomcat, the following licensed programs, PTFs, and software are required.

- License program
  Tomcat V7 can run on Java Development Kit (JDK) 1.6 32-bit or JDK 1.6 64-bit. In this article, JDK 1.6 32-bit is taken as example.

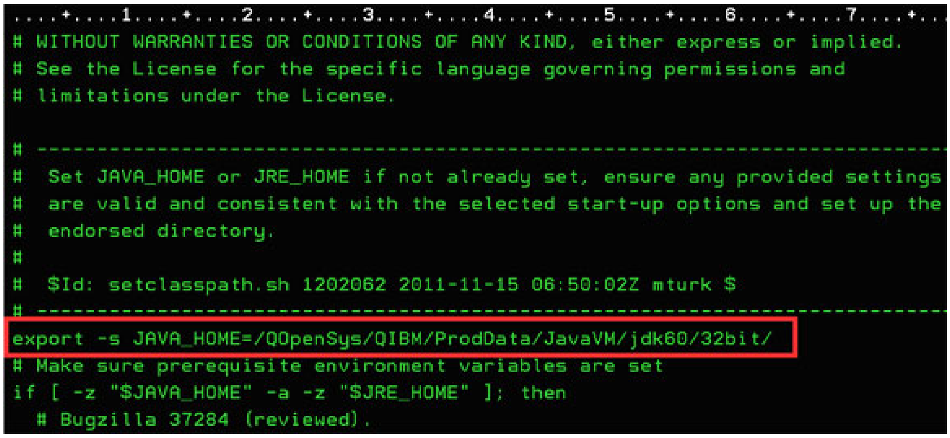
  - 5770-SS1 Option 30 QSH
  - 5770-DG1 *BASE IBM HTTP Server for i
  - 5761-JV1 *BASE IBM Developer Kit for Java
  - 5761-JV1 Option 11 Java SE 6 32 bit
- Required PTFs
  - Latest PTF Group for 5770DG1 (minimum SF99368 - level 14)
  - Latest PTF Group for Java (minimum SF99572 - level 10)
- Non-licensed software
  - Tomcat is not a licensed program on IBM i; the binary image can be download from the Apache website.
  - Distribution of Tomcat V7.0.28

Install Tomcat V7 on IBM i

After all the software are prepared, perform the following steps to install Tomcat V7.0.28 on IBM i.

1. Upload Tomcat V7.0.28 distribution to IBM i and place it in an integrated file system directory, for example, **/home/download.**
2. Extract the Tomcat distribution file in Qshell.
   cd /home/download
   jar -xvf apache-tomcat-7.0.28.zip
3. Set the JAVA_HOME environment variable. In order for Tomcat to start and run, it needs to know where the Java virtual machine (JVM) is located. This information needs to be added to the Tomcat configuration file. Edit the Tomcat configuration file: **/home/download/apache-tomcat-7.0.28/bin/setclasspath.sh.** Add the Java home information to the file. The following is the Java home information. **export -s JAVA_HOME=/QOpenSys/QIBM/ProdData/JavaVM/jdk60/32bit/**

   ### Figure 3. Export JAVA_HOME on IBM i

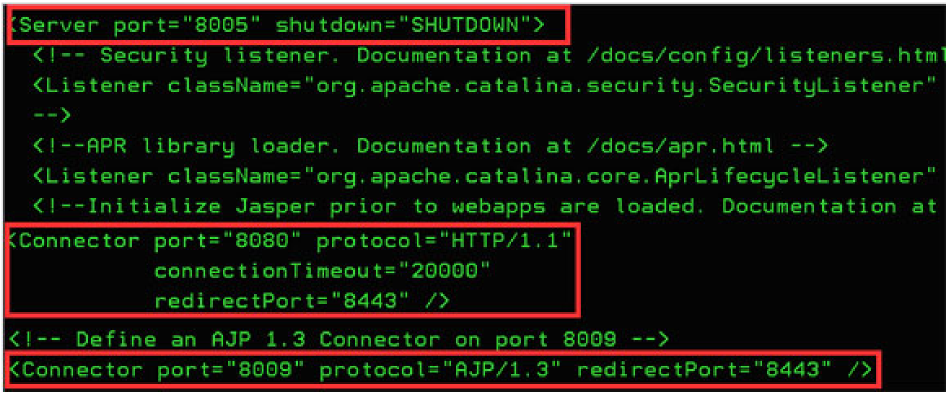
   

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+..
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

# -----------------------------------------------------------------------------
#  Set JAVA_HOME or JRE_HOME if not already set, ensure any provided settings
#  are valid and consistent with the selected start-up options and set up the
#  endorsed directory.
#
#  $Id: setclasspath.sh 1202062 2011-11-15 06:50:02Z mturk $
# -----------------------------------------------------------------------------
export -s JAVA_HOME=/QOpenSys/QIBM/ProdData/JavaVM/jdk60/32bit/
# Make sure prerequisite environment variables are set
if [ -z "$JAVA_HOME" -a -z "$JRE_HOME" ]; then
  # Bugzilla 37284 (reviewed).
```

4. Start Tomcat V7 on IBM i.
   1. Verify the availability of the ports used by Tomcat. Tomcat uses 8005, 8080, and 8009 by default among which 8080 is the default HTTP port. Enter the command NETSTAT OPTION(*CNN) to verify whether the ports are in use. If port 8009 and 8080 are occupied, edit the configuration file, /home/download/apache-tomcat-7.0.28/conf/

server.xml, to change default ports 8005, 8009, and 8080 to other ports which are not in use. If you have multiple Tomcat servers running on this IBM i, you will need to use different ports for these values.

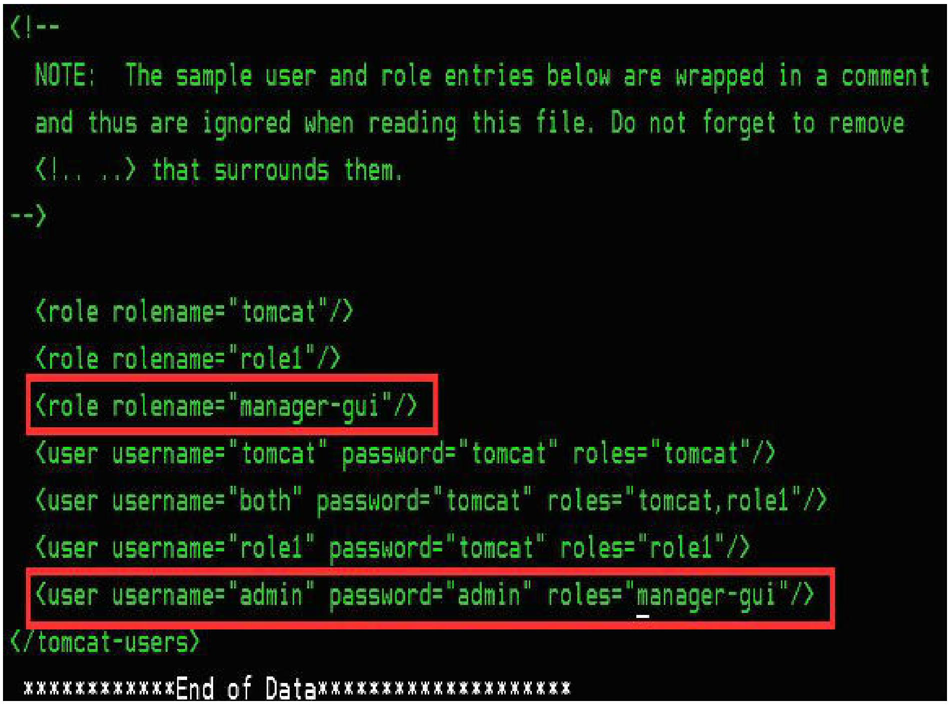
Figure 4. Editing Tomcat default ports



```
<Server port="8005" shutdown="SHUTDOWN">
  <!-- Security listener. Documentation at /docs/config/listeners.html
  <Listener className="org.apache.catalina.security.SecurityListener"
  -->
  <!--APR library loader. Documentation at /docs/apr.html -->
  <Listener className="org.apache.catalina.core.AprLifecycleListener"
  <!--Initialize Jasper prior to webapps are loaded. Documentation at
<Connector port="8080" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443" />
<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
```

2. Update the configuration of application Tomcat Web Application Manager application. Tomcat default web application, Tomcat Web Application Manager, is taken as an example to verify whether Tomcat Server works correctly on IBM i. So we need to update the configuration of the application before starting Tomcat.
    - Edit /home/download/apache-tomcat-7.0.28/conf/tomcat-users.xml to ensure the sample application is accessible.
    - Add <role rolename= "manager-gui”/> after all existing role elements.
    - Add <user username="admin” password=”admin” roles=”manager-gui”/> after all existing user elements.

Figure 5. Editing Tomcat user file

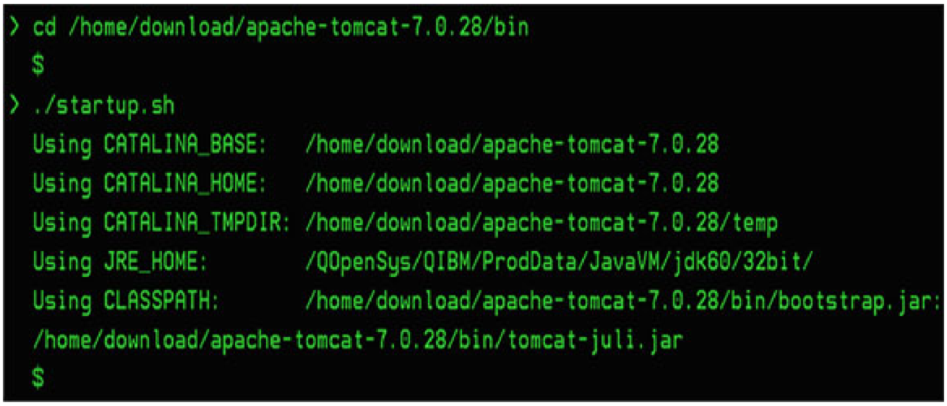


```
<!--
  NOTE:  The sample user and role entries below are wrapped in a comment
  and thus are ignored when reading this file. Do not forget to remove
  <!.. ..> that surrounds them.
-->

  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <role rolename="manager-gui"/>
  <user username="tomcat" password="tomcat" roles="tomcat"/>
  <user username="both" password="tomcat" roles="tomcat,role1"/>
  <user username="role1" password="tomcat" roles="role1"/>
  <user username="admin" password="admin" roles="manager-gui"/>
</tomcat-users>
 ************End of Data********************
```

3. Start Tomcat. Tomcat is installed and configured successfully. It is now ready to start through a script in Qshell. Change to the Tomcat bin directory, in this example, /home/download/apache-tomcat-7.0.28/bin. There are several useful scripts such as startup.sh, shutdown.sh, setClasspath.sh, Catalina.sh, and so on in this directory. Run the startup.sh script in Qshell to start the Tomcat server.

```
/startup.sh
```

Figure 6. Starting Tomcat using shell script

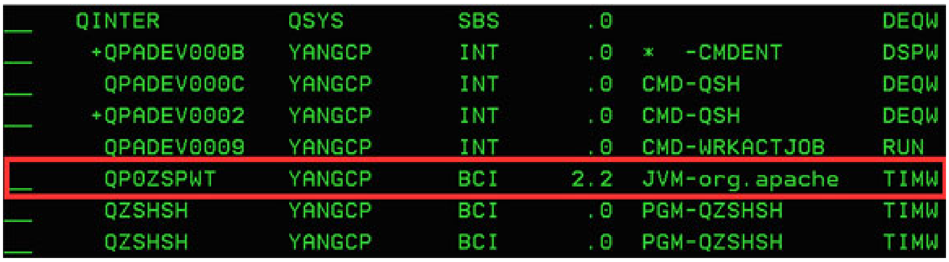


```
> cd /home/download/apache-tomcat-7.0.28/bin
  $
> ./startup.sh
  Using CATALINA_BASE:   /home/download/apache-tomcat-7.0.28
  Using CATALINA_HOME:   /home/download/apache-tomcat-7.0.28
  Using CATALINA_TMPDIR: /home/download/apache-tomcat-7.0.28/temp
  Using JRE_HOME:        /QOpenSys/QIBM/ProdData/JavaVM/jdk60/32bit/
  Using CLASSPATH:       /home/download/apache-tomcat-7.0.28/bin/bootstrap.jar:
  /home/download/apache-tomcat-7.0.28/bin/tomcat-juli.jar
  $
```

4. Verify that Tomcat is started on IBM i. Verify whether the Tomcat server is running by entering the CL command, `WRKACTJOB`. Look for the QP0ZSPWT job in the QINTER subsystem and make sure that its status is `TIMW`. Tomcat Server can be run in a

subsystem defined by users. An article planned for future publication introduces how to run Tomcat in a specialized subsystem and other advanced topics.

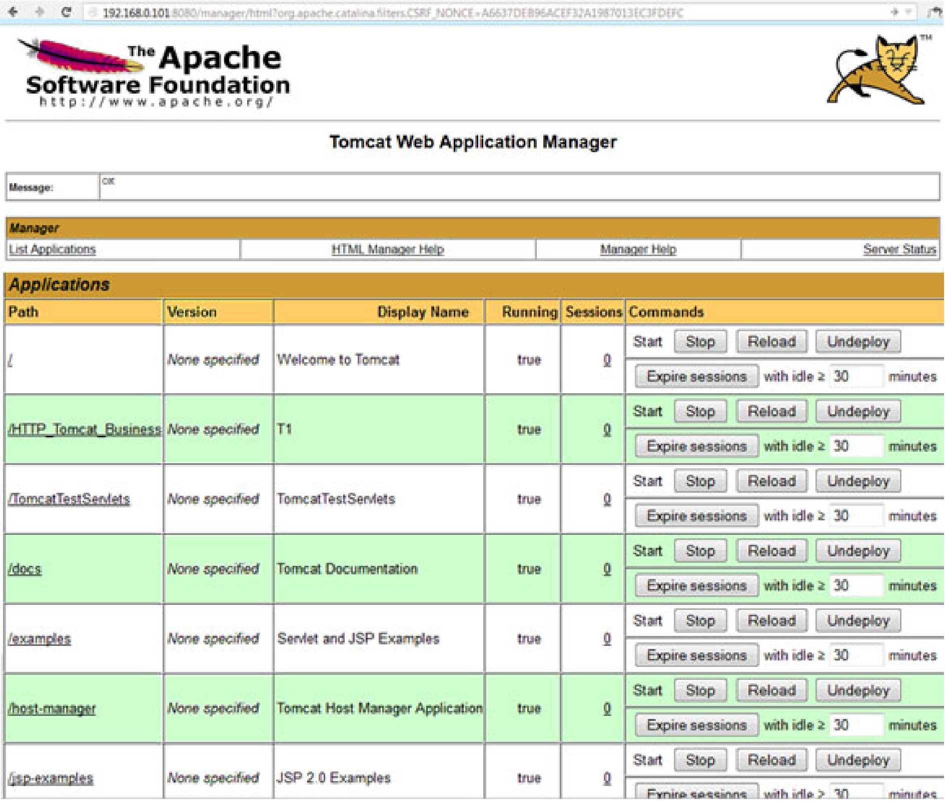
Figure 7. Tomcat job on IBM i



```
__    QINTER              QSYS        SBS        .0                 DEQW
__      +QPADEV000B       YANGCP      INT        .0    *  -CMDENT   DSPW
__       QPADEV000C       YANGCP      INT        .0    CMD-QSH      DEQW
__      +QPADEV0002       YANGCP      INT        .0    CMD-QSH      DEQW
         QPADEV0009       YANGCP      INT        .0    CMD-WRKACTJOB RUN
__       QP0ZSPWT         YANGCP      BCI       2.2    JVM-org.apache TIMW
__       QZSHSH           YANGCP      BCI        .0    PGM-QZSHSH   TIMW
__       QZSHSH           YANGCP      BCI        .0    PGM-QZSHSH   TIMW
```

5. Verify that Tomcat works correctly on IBM i.
    - Open a web browser and enter the URL http://your.server.name:8080/manager.
    - Enter the default user ID and password (admin/admin) when prompted. The manager application is displayed as shown in the following figure :

    Figure 8. Main page of Tomcat Web Application Manager

    192.168.0.101:8080/manager/html?org.apache.catalina.filters.CSRF_NONCE=A6637DEB96ACEF32A1987013EC3FDEFC

    Tomcat Web Application Manager

    Message: OK

    Manager

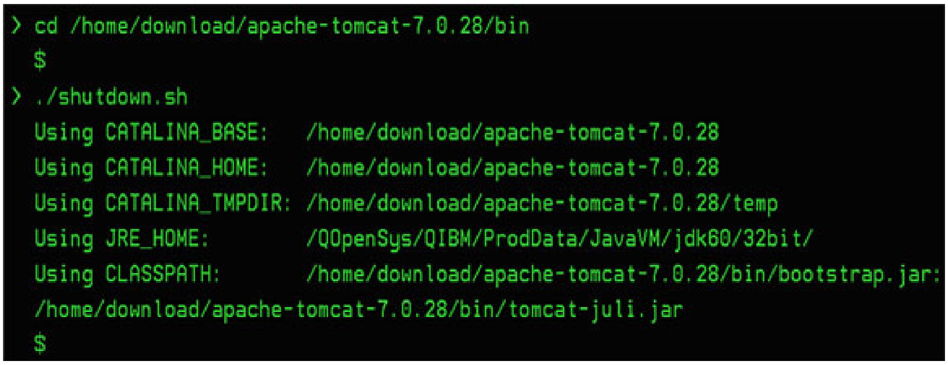
    List Applications | HTML Manager Help | Manager Help | Server Status

    Applications

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/HTTP_Tomcat_Business	None specified	T1	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/TomcatTestServlets	None specified	TomcatTestServlets	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/jsp-examples	None specified	JSP 2.0 Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

    Now Tomcat has been installed on IBM i successfully. If you want to stop Tomcat, run the **/home/download/apache-tomcat-7.0.28/bin/shutdown.sh** script in Qshell.

Figure 9. Stopping Tomcat in QSH



```
> cd /home/download/apache-tomcat-7.0.28/bin
  $
> ./shutdown.sh
  Using CATALINA_BASE:   /home/download/apache-tomcat-7.0.28
  Using CATALINA_HOME:   /home/download/apache-tomcat-7.0.28
  Using CATALINA_TMPDIR: /home/download/apache-tomcat-7.0.28/temp
  Using JRE_HOME:        /QOpenSys/QIBM/ProdData/JavaVM/jdk60/32bit/
  Using CLASSPATH:       /home/download/apache-tomcat-7.0.28/bin/bootstrap.jar:
  /home/download/apache-tomcat-7.0.28/bin/tomcat-juli.jar
  $
```

Next section introduces how to associate HTTP Server for i with Tomcat to make a full production web environment on IBM i.

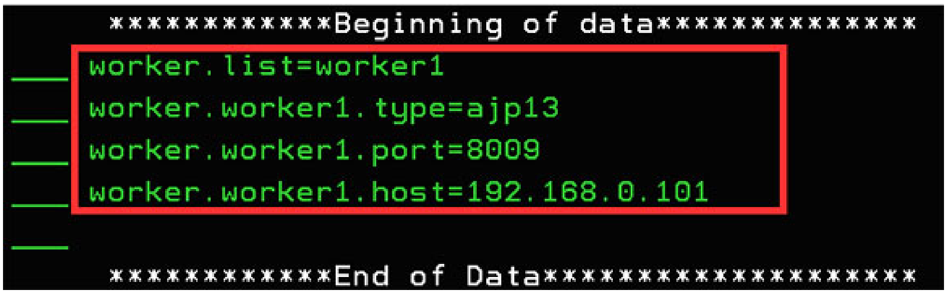
Associate HTTP Server for i with Tomcat

Perform the following steps to associate HTTP Server for i with Tomcat.

1. Create an instance of HTTP Server for i.
   An instance of HTTP Sever for i must exist before associating HTTP Server with Tomcat. IBM Web Administration for i (http://your.server.name:2001/HTTPAdmin) can help create an instance of HTTP Sever for i. Refer to the IBM i Information Center for more details about creating instances of HTTP Server for i. This article uses the instance httpserver (root directory: /www/httpserver, port: 5555) as an example to demonstrate how to configure and associate HTTP Server for i with Tomcat on IBM i.
2. Configure HTTP Server for i.
   In order to associate HTTP Server for i with Tomcat, the Tomcat plug-in module which works as a connector between HTTP Server and Tomcat is needed to route the requests from the HTTP Server for i to Tomcat and get response back from Tomcat to HTTP Server. The default Tomcat plug-in service program included by the 5770DG1 license program is QHTTPSVR/QZTCJK.
   1. Copy server.xml from the Tomcat to HTTP Server for i.
      If Tomcat and HTTP Server for i are located on the same IBM i server, copy server.xml from the /home/download/apache-tomcat-7.0.28/conf directory to the /www/httpserver/conf directory directly. If Tomcat and HTTP Server for i are installed on different IBM i servers, copy server.xml from the server where Tomcat is located to the server where HTTP Server is located.
   2. Generate workers.properties.
      Create the workers.properties file in the /www/httpserver/conf directory. Add the following lines of code to the workers.properties file:

```
            worker.list=worker1
            worker.worker1.type=ajp13
            worker.worker1.port=8009
            worker.worker1.host=f4p04.cn.ibm.com
```

Figure 10. Edit workers.properties using Programming Development Manager (PDM)



```
      ************Beginning of data**************
___ worker.list=worker1
___ worker.worker1.type=ajp13
___ worker.worker1.port=8009
___ worker.worker1.host=192.168.0.101

___
      ************End of Data********************
```

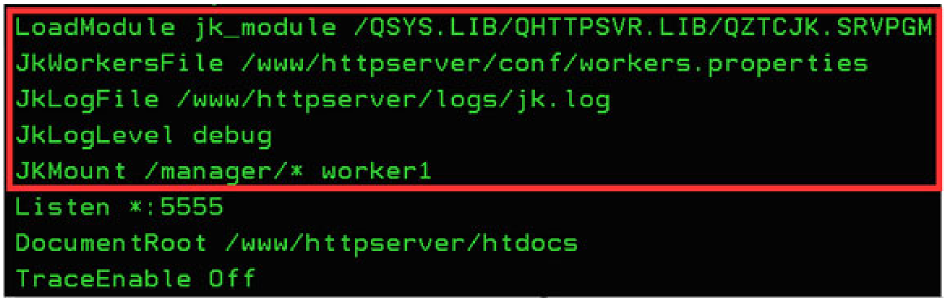
Description of the property file:

- worker.list=worker1
  The name of the Tomcat worker, worker1 is used as the default name.
- worker.worker1.type=ajp13
  The protocol used to exchange data between HTTP Server for i and Tomcat, ajp13 is used by default.
- worker.worker1.port=8009
  The port for the communication between HTTP Server for i and Tomcat, 8009 is used by default.
- worker.worker1.host
  The IP address or hostname of the server where Tomcat runs. If Tomcat and HTTP Server are located on different servers, the value refers to the server where Tomcat is located.

3. Edit the configuration file.
   Edit the /www/httpserver/conf/httpd.conf file. Add Tomcat plug-in related directives to httpd.conf.

```
            LoadModule jk_module /QSYS.LIB/QHTTPSVR.LIB/QZTCJK.SRVPGM
            JkWorkersFile /www/ct/conf/workers.properties
            JkLogFile /www/ct/logs/jk.log
            JkLogLevel debug
            JKMount /manager/* worker1
```

Figure 11. Adding Tomcat plug-in related directives to httpd.conf



```
LoadModule jk_module /QSYS.LIB/QHTTPSVR.LIB/QZTCJK.SRVPGM
JkWorkersFile /www/httpserver/conf/workers.properties
JkLogFile /www/httpserver/logs/jk.log
JkLogLevel debug
JKMount /manager/* worker1
Listen *:5555
DocumentRoot /www/httpserver/htdocs
TraceEnable Off
```

Description of the directives:

  - **LoadModule jk_module /QSYS.LIB/QHTTPSVR.LIB/QZTCJK.SRVPGM:** The directive specifies the Tomcat plug-in module loaded by HTTP server for i for communication with the Tomcat server. It must be the first entry in the configuration file.
  - **JkWorkersFile /www/httpserver/conf/workers.properties:** The directive specifies the name of a worker file for Tomcat servlet containers. It can only be used once in the configuration file and must be put into the global configuration area.
  - **JkLogFile /www/httpserver/logs/jk.log:** The directive specifies the full path name of the log file for the Tomcat plug-in.
  - **JkLogLevel debug:** The directive specifies the log level of the log file defined by JkLogFile.)
  - **JKMount /manager/* worker1:**The directive specifies a mount point from a context to a Tomcat worker. It can be used multiple times in the configuration file and can be specified in the global configuration and VirtualHost area.

3. Verify the association between Tomcat and HTTP Server for i.
   Start HTTP Server for i with the STRTCPSVR SERVER(*HTTP) HTTPSVR(httpserver) CL command. Open a web browser and enter the URL: **http://IP address or hostname:port/manager/**. The page displayed through the HTTP Server hostname and port should look exactly the same as the page displayed through the Tomcat hostname and port (8080). Refer to Figure 8.
   Note: If Tomcat and HTTP Server for i are located on different servers, replace **IP address or hostname** with the IP or host name of the server that HTTP Server for i is running on.

Deploy IBM i web application on Tomcat

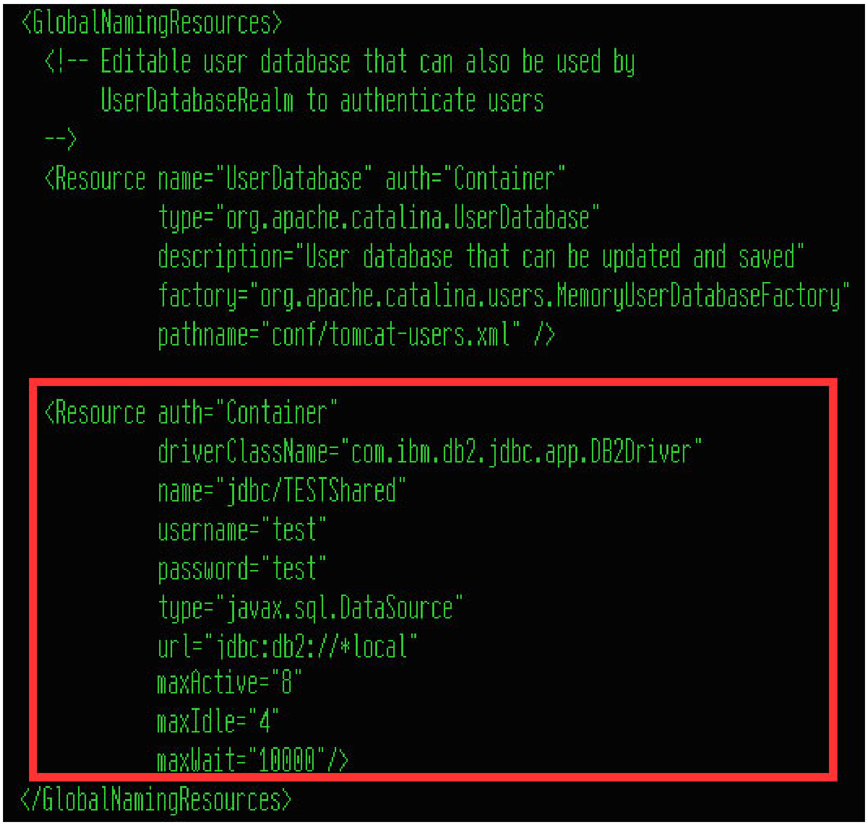
Now HTTP Server for i and Tomcat have been associated together on IBM i and Tomcat sample application, Tomcat Application Manager, can be visited through HTTP server for i. This section introduces how to deploy an IBM i web application on Tomcat. A sample application (TomcatTestServlets) is given to test the time of database queries using IBM Toolbox and native IBM DB2® JDBC Driver (Refer to attachments 1 and 2 to get the .war file and the source code).

1. This sample uses the data source and you need to edit the configuration files to add data sources in the Tomcat configuration files and your applications. If your applications do not require the data source, you can ignore this step.
   1. Edit the /home/download/apache-tomcat-7.0.28/conf/server.xml configuration file. Insert the following lines in the `<GlobalNamingResources>` node to add a data source:

```
        Resource auth="Container"
                  driverClassName="com.ibm.db2.jcc.DB2Driver"
                  name="jdbc/TEST"
                  username="test"
                  password="test"
                  type="javax.sql.DataSource"
                  url="jdbc:db2://ip | hostname:port/DBName"
                    maxActive="8"
                    maxIdle="4"
                    maxWait="10000"/>
```

      The new configuration file server.xml is as shown below:

Figure 12. Adding data source in the server.xml file



```
<GlobalNamingResources>
  <!-- Editable user database that can also be used by
       UserDatabaseRealm to authenticate users
  -->
  <Resource name="UserDatabase" auth="Container"
            type="org.apache.catalina.UserDatabase"
            description="User database that can be updated and saved"
            factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
            pathname="conf/tomcat-users.xml" />

  <Resource auth="Container"
            driverClassName="com.ibm.db2.jdbc.app.DB2Driver"
            name="jdbc/TESTShared"
            username="test"
            password="test"
            type="javax.sql.DataSource"
            url="jdbc:db2://*local"
            maxActive="8"
            maxIdle="4"
            maxWait="10000"/>
</GlobalNamingResources>
```

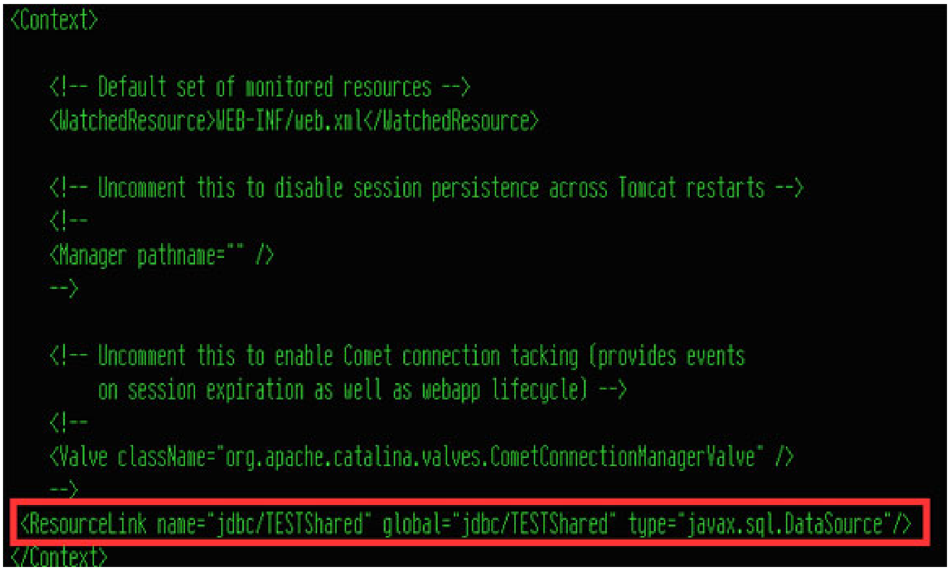
Description of Resource attributes:

- **auth:** Authentication type of data source. The value must be **Application** or **Container**. Default value is **Container**.
- **driverClassName:** Fully qualified Java class name of the JDBC driver to be used.
- **name:** Qualified name of the data source.
- **username:** Database username to be passed to the JDBC driver
- **password:** Database password to be passed to the JDBC driver.
- **type:** Qualified Java class name expected by the web application when it performs a lookup for this resource.
- **url:** Connection URL to be passed to the JDBC driver.
- **initialSize:** Initial connections that will be created in the pool during pool initialization. Default: 0.
- **maxActive:** Maximum connections that can be allocated from this pool at the same time. Default: 8.
- **minIdle:** Minimum connections that will sit idle at the same time. Default: 0
- **maxIdle:** Maximum connections that can sit idle at the same time. Default: 8
- **maxWait:** Maximum duration (in milliseconds) that the pool will wait for a connection to be returned. Default: -1 (infinite)

2. Edit the **/home/download/apache-tomcat-7.0.28/conf/context.xml** configuration file. Insert the following lines in the <context/> node. This item specifies a link to the data source defined in **server.xml:**

3. Edit the **TomcatTestServlets/WebContent/WEB-INF/web.xml** configuration file. Add the following lines to the very end of the file. These lines specify a reference to the data source defined in **server.xml:**
    - **name:** Qualified name of the link to the data source defined in server.xml.
    - **global:** Name of the linked global resource in the global Java Naming and Directory Interface (JNDI) context.
    - **type:** Qualified Java class name expected by the web application when it performs a lookup for this resource

    ### Figure 13. Adding a link to the data source in context.xml

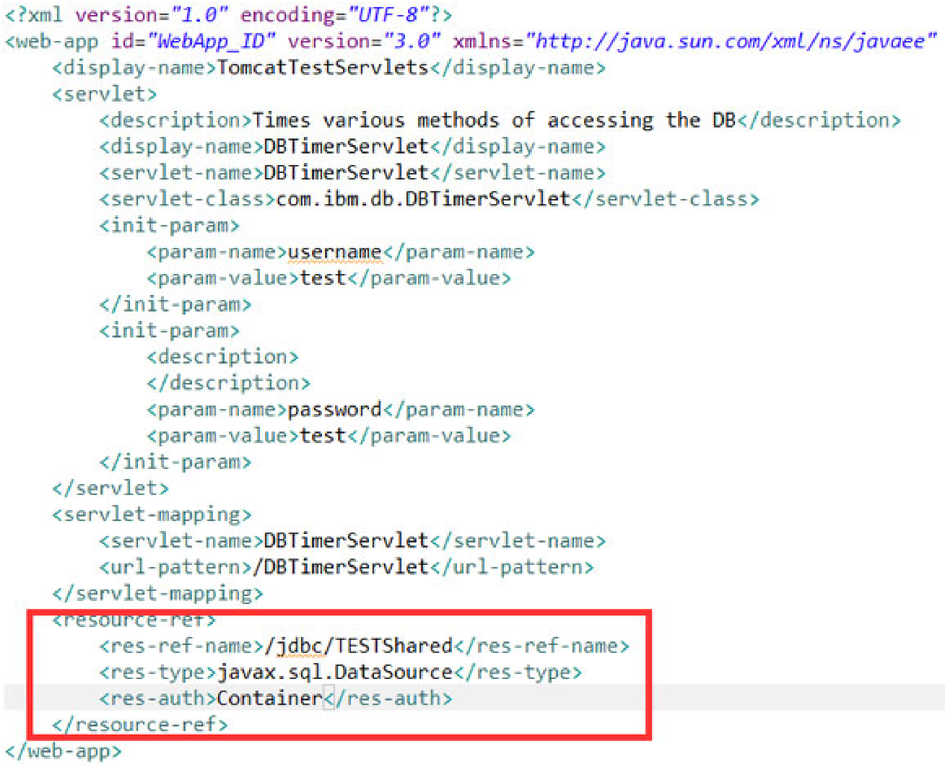
    ```
    <Context>

        <!-- Default set of monitored resources -->
        <WatchedResource>WEB-INF/web.xml</WatchedResource>

        <!-- Uncomment this to disable session persistence across Tomcat restarts -->
        <!--
        <Manager pathname="" />
        -->

        <!-- Uncomment this to enable Comet connection tacking (provides events
             on session expiration as well as webapp lifecycle) -->
        <!--
        <Valve className="org.apache.catalina.valves.CometConnectionManagerValve" />
        -->
     <ResourceLink name="jdbc/TESTShared" global="jdbc/TESTShared" type="javax.sql.DataSource"/>
    </Context>
    ```

    - **res-ref-name:** Qualified name of data source’s link or data source itself. The value must be the same as the resource link’s name defined in context.xml.
    - **res-type:** Qualified Java class name expected by the web application when it performs a lookup for this resource.
    - **res-auth:** Authentication type of data source. The value must be Application or Container. Default: Container.
4. Edit configuration file: TomcatTestServlets/WebContent/WEB-INF/web.xml. Add following lines to the very end of the file. These lines specify a reference to data source defined in server.xml:

    ```
            resource-ref
            res-ref-name /jdbc/TEST /res-ref-name
            res-type    javax.sql.DataSource res-type
            res-auth    Container res-auth
            /resource-ref
    ```

    The new configuration file, web.xml, appears as shown in Figure 14.

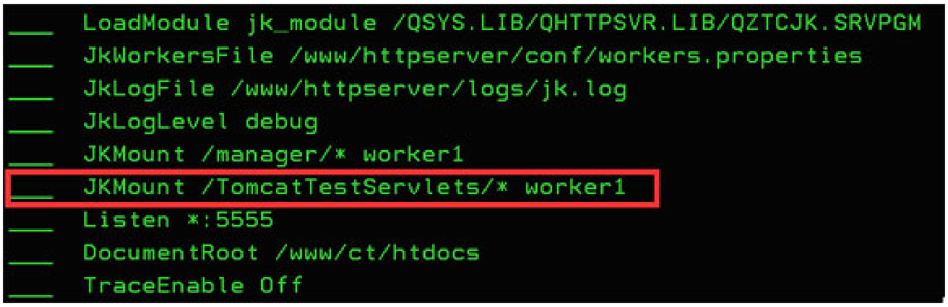
Figure 14. Adding a reference to the data source in web.xml



```
<?xml version="1.0" encoding="UTF-8"?>
<web-app id="WebApp_ID" version="3.0" xmlns="http://java.sun.com/xml/ns/javaee"
    <display-name>TomcatTestServlets</display-name>
    <servlet>
        <description>Times various methods of accessing the DB</description>
        <display-name>DBTimerServlet</display-name>
        <servlet-name>DBTimerServlet</servlet-name>
        <servlet-class>com.ibm.db.DBTimerServlet</servlet-class>
        <init-param>
            <param-name>username</param-name>
            <param-value>test</param-value>
        </init-param>
        <init-param>
            <description>
            </description>
            <param-name>password</param-name>
            <param-value>test</param-value>
        </init-param>
    </servlet>
    <servlet-mapping>
        <servlet-name>DBTimerServlet</servlet-name>
        <url-pattern>/DBTimerServlet</url-pattern>
    </servlet-mapping>
    <resource-ref>
        <res-ref-name>/jdbc/TESTShared</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <res-auth>Container</res-auth>
    </resource-ref>
</web-app>
```

2. Copy **TomcatTestServlets.war** to the **/home/download/apache-tomcat-7.0.28/webapps** directory on IBM i.
3. Edit the **/www/httpserver/conf/httpd.conf** configuration file. Add a new mount point from the sample application web root (TomcatTestServlets) to Tomcat worker (worker1) in **httpd.conf.**
   JKMount /TomcatTestServlets/* worker1
   The new configuration file, httpd.conf, is as shown in Figure 15.

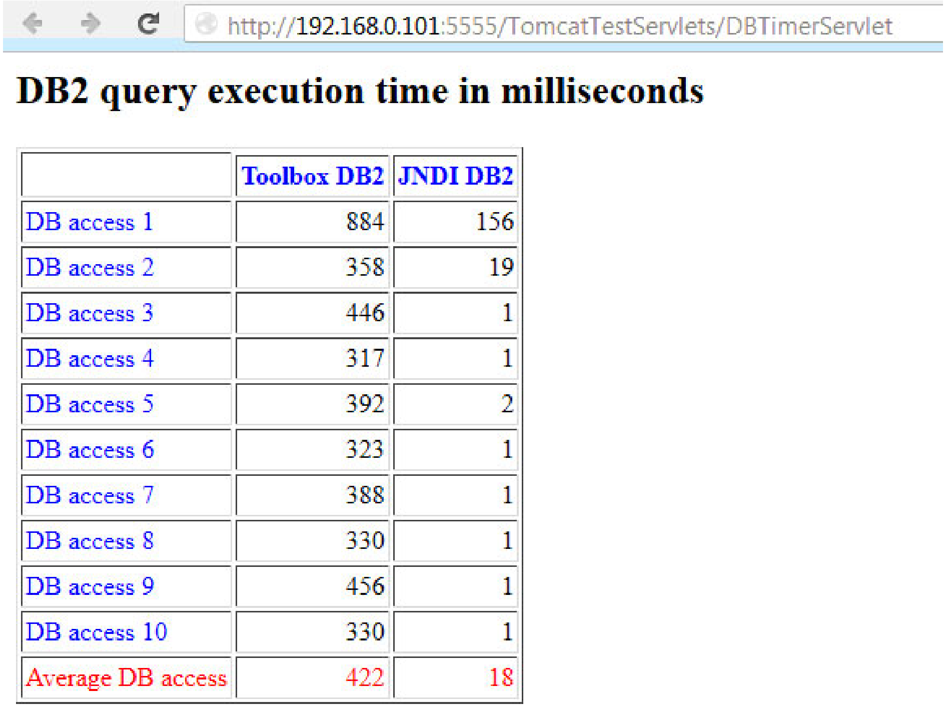
Figure 15. Viewing the application through HTTP Server for i



```
___   LoadModule jk_module /QSYS.LIB/QHTTPSVR.LIB/QZTCJK.SRVPGM
___   JkWorkersFile /www/httpserver/conf/workers.properties
___   JkLogFile /www/httpserver/logs/jk.log
___   JkLogLevel debug
___   JKMount /manager/* worker1
___   JKMount /TomcatTestServlets/* worker1
___   Listen *:5555
___   DocumentRoot /www/ct/htdocs
___   TraceEnable Off
```

4. Restart Tomcat and HTTP Server for i. Enter the URL: **http://192.168.0.101:5555/TomcatTestServlets/DBTimerServlet** to verify the application. The page, as shown in Figure 16 is displayed.

Figure 16. Viewing the application through HTTP Server for i



http://192.168.0.101:5555/TomcatTestServlets/DBTimerServlet

DB2 query execution time in milliseconds

	Toolbox DB2	JNDI DB2
DB access 1	884	156
DB access 2	358	19
DB access 3	446	1
DB access 4	317	1
DB access 5	392	2
DB access 6	323	1
DB access 7	388	1
DB access 8	330	1
DB access 9	456	1
DB access 10	330	1
Average DB access	422	18

Attachments

1. TomcatTestServlets.war:
   (**Note:** To download the code, right-click the link and select **Save as** or **Save link as**.)
2. TomcatTestServlets-sourcecode:

Summary

Many customers would like to use Tomcat on IBM i today because IBM i is a great platform for running any highly multithreading workload, such as Tomcat. However, Tomcat is not included with the IBM license program on IBM i currently. This article recommends a feasible solution explaining how to deploy and customize Tomcat on IBM i, associate Tomcat with HTTP Server for i, and run web solutions in a full-production web environment. In addition, an article planned for future publication will focus on some advanced topics, such as running Tomcat specified subsystem mentioned in the Install Tomcat V7 on IBM i section, the performance tuning of Tomcat and HTTP Server on IBM i, and so on. Enjoy your experience of using Tomcat on IBM i!.

Related topics

- IBM HTTP Server (powered by Apache) An Integrated Solution for IBM iSeries Servers.
- IBM i 7.1 Infocenter.
- Apache Tomcat
- Apache Tomcat - Downloads
- Install Tomcat on iSeries - AS400 Tomcat 6.0 installation steps
- mrc's Tech Blog: Install Tomcat 7 on the iSeries