Institut für Wirtschaftsinformatik und Gesellschaft

Using Tomcat (a Java Web Server) to Create and Run Web Server Programs Written in ooRexx

2021 – International Rexx Symposium
Online ("Covid-19")
November 7th – November 10th 2021

Rony G. Flatscher

Agenda

- Java based web servers
  - Brief overview
  - Servlets
  - Java server pages (JSP), taglibs
- BSF4ooRexx and Tomcat
  - Setup Tomcat
  - ScriptTagLibs.jar (javax, jakarta)
  - Nutshell samples
- Roundup, outlook, URLs

Java Based Web Servers, 1

- Specifications modularized
  - Java based web servers implement these, e.g.
    - Adobe "Cold Fusion", Apache "Tomcat", Caucho "Resin", Eclipse "Glassfish" (formerly Oracle), Eclipse "Jetty", IBM "WebSphere", Oracle "WebLogic" (formerly BEA), Red Hat "WildFly" (formerly JBoss), ...
  - Everything is and needs to be implemented in Java
  - "Servlets"
    - Subclass abstract class "{javax|jakarta}.servlet.http.HttpServlet"
    - Therefore small Java programs that serve client requests

Java Based Web Servers, 2

- "Java Server Pages (JSP)"
  - Java code injected into HTML/XML text
  - Gets rewritten as a Java Servlet and then compiled
    - Repeated if JSP source changes
  - Taglibs – "JSP Tag Libraries"
    - Custom tags in JSPs
    - Java implemented tag handlers
    - Allows extending JSP functionalities
      - E.g. JSP Standard Tag Library (STL) for easying control structures, accessing SQL databases etc.

Apache Tomcat (https://tomcat.apache.org), 1

- Java web server
  - Original reference implementation of specifications (Sun)
  - Open-source, now by Apache Software Foundation (ASF)
  - Can be deployed as a
    - Fully fledged, standalone web server
    - Component in complex web server configurations
      - E.g. in Apache httpd, NINGX
- Flexible installations
  - Configurable at various levels in a "cascading" manner
  - Deploying/removing web applications without restart, ...

Apache Tomcat (https://tomcat.apache.org), 2

- Tomcat modules
  - "Catalina"
    - Servlet container component that implements and hosts the servlet and Java server pages specifications and manages users defined in "realms"
  - "Jasper"
    - JSP component to manage tag libraries, compile and recompile JSPs
  - "Coyote"
    - Connector component that allows access via http, making Tomcat effectively a web server

Apache Tomcat (https://tomcat.apache.org), 3

- Tomcat directory layout (in `CATALINA_HOME`)
  - "`bin`": start, stop Tomcat from commandline
  - "`conf`": Tomcat configuration files like server.xml, config.xml, tomcat-users.xml
  - "`lib`": jar files available to all deployed web applications
  - "`logs`", "`temp`", "`work`": log files, temporary and work directories
  - "`webapps`": root for "web applications" and "web application archives" (war) files
    - Copying a xyz.war file deploys (and explodes) the web application "xyz"
    - Deleting a xyz.war file undeploys (and deletes) the web application "xyz"
    - Predefined web apps (subdirectories): "`docs`" (Tomcat documentation), "`manager`" (Tomcat manager, must be enabled see URL section), "`ROOT`" (default web app)

Apache Tomcat (https://tomcat.apache.org), 4

- Web application archive (war)
  - A zip/jar (Java archive) file that includes all resources for the application
    - "xyz.war": "xyz" name of the web application (url: "http://server/xyz")
    - Deployment will explode the archive into the subdirectory named "xyz"
  - Any files and subdirectories that constitute the web application
    - File "index.html", optional, returned by default if browsing "http://server/xyz"
    - Directory "WEB-INF", optional may contain
      - The file "web.xml" (title, configuration, additional resource definitions),
      - The directory "classes" (web app specific Java classes),
      - The directory "lib" (web app specific jar files), taglib definitions, ...
    - Directory "META-INF", optional may contain
      - The file "config.xml" (web app related configurations)

Apache Tomcat (https://tomcat.apache.org), 5

- "Java EE" vs. "Jakarta EE"
  - Oracle insists on using exclusively the top level name javax
    - Approached the open-source community
    - Eclipse foundation new owner of "EE" specifications uses the top level name jakarta
      - All specifications and Java class libraries need to be changed! :-(
  - Hence, up to and including *Tomcat 9* ("Java EE") use
    - **javax**.ScriptTagLibs.jar
  - *Tomcat 10 or later* ("Jakarta EE") *use*
    - **jakarta**.ScriptTagLibs.jar

ooRexx and Web Servers, 1

- Apache httpd (https://httpd.apache.org/)
  - "Classic" web server, for decades leading the pack
  - Implemented in C, C++
  - *CGI* (common gateway interface)
    - Each client request served in a proper process, rather expensive
    - Communicating request information via environment variables
    - ooRexx can be used out of the box
  - Apache httpd *modules*
    - Each client request served in a thread, quite efficient
    - Allows processing each Apache stage in request and response

ooRexx and Web Servers, 2

- Java based web servers
  - No support for non-Java programming languages like ooRexx
  - To empower WU students to create web server programs a tag library was created by the author (summer of 2020): ScriptTagLibs.jar
    - Tag library to add two tags to JSPs for supporting script languages
      - Tags "script" and "expr"
      - Supporting Apache BSF and Java's newer javax.script framework
    - Allows ooRexx to be used for creating
      - "Rexx Servlets" (scripts always get the Servlet arguments supplied)
      - RSPs ("Rexx server pages")
    - Can be used for *any* script language that implements javax.script (a.k.a. "JSR-223", Java specification request # 223)

BSF4ooRexx and Tomcat
Setup Tomcat

- Make *BSF4ooRexx* and the *ScriptTagLibs.jar* available to all Tomcat web applications
  - Tomcat, any version
    - Copy bsf4ooRexx-v641-20210516-bin.jar (or newer) to CATALINA_HOME/lib
  - Tomcat 9 or earlier (Java EE)
    - Copy **javax**.ScriptTagLibs.jar to CATALINA_HOME/lib
  - Tomcat 10 or later (Jakarta EE)
    - Copy **jakarta**.ScriptTagLibs.jar to CATALINA_HOME/lib

BSF4ooRexx, Nutshell Example, Code "rexxla_03_ScriptTagLib/helloworld-jsr223-01.jsp"

```
<%@ page session="false" pageEncoding="ISO-8859-1" contentType="text/html; charset=ISO-8859-1" %>
<%@ taglib uri="http://rexxla.org/taglibs/jsr223" prefix="s" %>

<!DOCTYPE html>
<html>

<head>
   <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
   <title>Minimal ScriptTagLibs-JSP</title>
</head>

<body>
   <s:script type="rexx">
      use arg request
      say "<h1>Hello, world (ScriptTagLibs JSP)</h1>"
      say "<p>This JSP was executed, because of the following URL:"
      say "<ul><li>URL <em>request~getRequestURL()</em>: <br>"
      say "<code>"request~getRequestURL~toString"</code>"
      say "<li>its URI being <em>request~getRequestURI()</em>: <br>"
      say "<code>"request~getRequestURI"</code></ul></p>"
   </s:script>
</body>
</html>
```

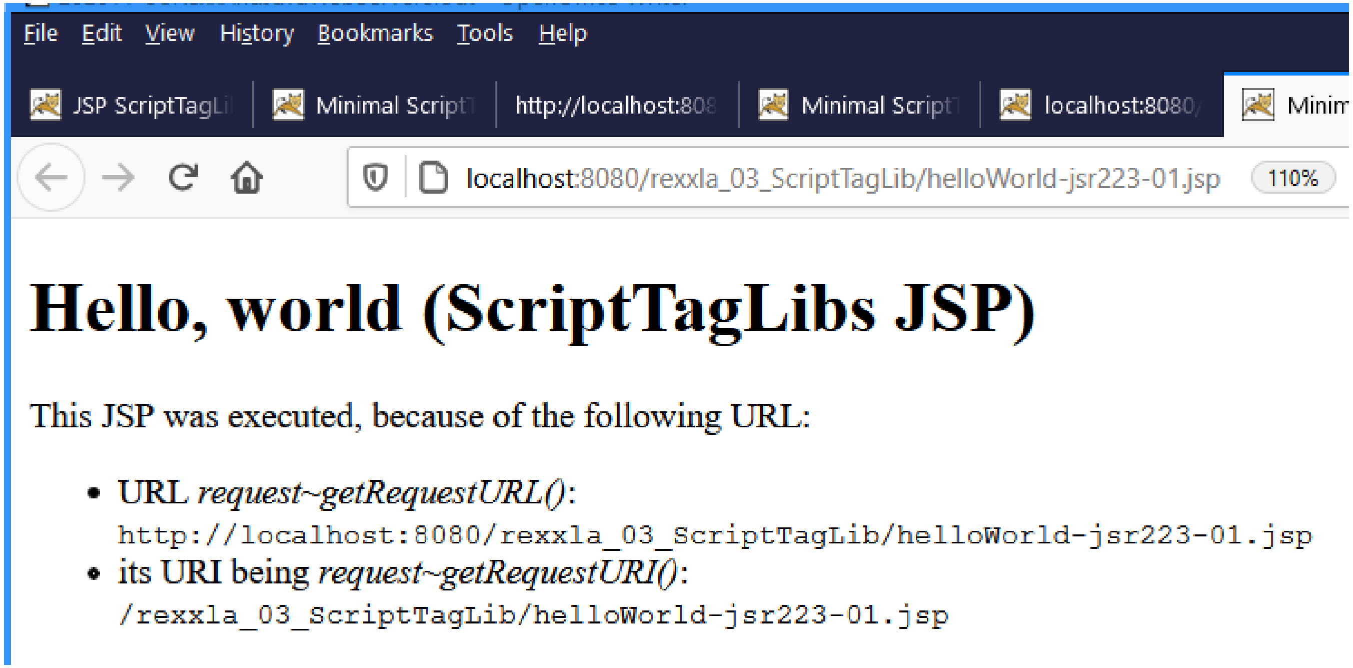
Code snippet from:

https://www.rexxla.org/events/2020/presentations/202011-ooRexxAndJavaWebServers-article-code.zip

Explained in detail in article:

https://www.rexxla.org/events/2020/presentations/202011-ooRexxAndJavaWebServers-article.pdf

BSF4ooRexx, Nutshell Example, Result "rexxla_03_ScriptTagLib/helloworld-jsr223-01.jsp"


File Edit View History Bookmarks Tools Help
JSP ScriptTagLi
Minimal Script
http://localhost:808
Minimal Script
localhost:8080/
Minim
localhost:8080/rexxla_03_ScriptTagLib/helloWorld-jsr223-01.jsp
110%

Hello, world (ScriptTagLibs JSP)

This JSP was executed, because of the following URL:

- URL *request~getRequestURL()*: `http://localhost:8080/rexxla_03_ScriptTagLib/helloWorld-jsr223-01.jsp`
- its URI being *request~getRequestURI()*: `/rexxla_03_ScriptTagLib/helloWorld-jsr223-01.jsp`

Code snippet from:
https://www.rexxla.org/events/2020/presentations/202011-ooRexxAndJavaWebServers-article-code.zip

Explained in detail in article:
https://www.rexxla.org/events/2020/presentations/202011-ooRexxAndJavaWebServers-article.pdf

BSF4ooRexx, Nutshell Examples, Live

- Web applications (see URL section)
  - Can be downloaded from the *ScriptTagLibs* BSF4ooRexx project page (see URL section)
  - Web application "demoRexx.war"
    - Exploded in CATALINA_HOME/webapps/demoRexx
    - Demonstrates ooRexx nutshells and Tomcat
    - Contains a few ooRexx utilities
  - Web application "demoSTL-Core-SQL_ooRexx.war"
    - Exploded in CATALINA_HOME/webapps/demoSTL-Core-SQL_ooRexx
      - Demonstrates SQL STL, using SQL from web app, CGI with Tomcat

BSF4ooRexx, Nutshell Examples, Live

- Live demonstration of (BSF4)ooRexx web applications

  ...

Roundup and Outlook

- Java web servers, Servlets, JSPs
- "ScriptTagLibs"
  - ***javax***.*ScriptTagLibs.jar*, ***jakarta***.*ScriptTagLibs.jar*
- BSF4ooRexx & ScriptTagLibs with Apache Tomcat
  - Use ooRexx instead of Java in JSPs, hence "RSPs" ;-)
  - Possible to create ooRexx "servlet" programs
    - Each script gets the "request", "response" and "out" objects as arguments (in that order)
- WU students can create (BSF4)ooRexx web server applications !
  - And anyone else ...

Links

- Tomcat
  - https://en.wikipedia.org/wiki/Apache_Tomcat
  - https://tomcat.apache.org/
  - https://cwiki.apache.org/confluence/display/TOMCAT/Specifications
- ScriptTagLibs (with directions installing and configuring Tomcat)
  - https://sourceforge.net/projects/bsf4oorexx/files/Sandbox/rgf/taglibs/ga/
    - ScriptTagLibs: **javax**.ScriptTagLibs.jar, **jakarta**.ScriptTagLibs.jar
      - demoRexx.war (ooRexx nutshell samples, a few utilities)
      - demoSTL-Core-SQL_ooRexx.war (standard taglib library "core" to ease usage of SQL, equivalent use of SQL from ooRexx)
        - JSP Standard Tag Library (STL): https://tomcat.apache.org/taglibs/standard/
- ApacheCon 2021 Presentation on the *ScriptTagLibs* library:

  "Apache Tomcat: Enabling Scripting Languages in JSPs": https://epub.wu.ac.at/8303/
- WU student's Tomcat papers: http://wi.wu.ac.at/rgf/diplomarbeiten/
- Article: https://www.rexxla.org/2020/presentations/202011-ooRexxAndJavaWebServers-article.pdf

  https://epub.wu.ac.at/8117/