BeaWeblogic

Getting Cocoon running in BEA Weblogic

In general Cocoon is running in every Application Server. Mostly it is used inside Apache Tomcat, sometimes it is necessary to run Cocoon in different Application Servers.

BEA Weblogic is one of mostly used Application Servers, so it might be useful to get Cocoon running here.

Cocoon 2.1.x and BEA Weblogic 6.1

We've tested Cocoon 2.1.x under Solaris, Windows and Linux. The most problems occur using Solaris Environment. There are only a few steps to do:

* Flowscript is only working with the latest SVN version of Cocoon on Weblogic 6.1 because of Rhino issues. So check out latest Cocoon or take a nightly build.
* Compile the SVN checkout with the build script using **JDK 1.3.1**
* Deploy the application, go to the weblogic console
    * In "Configuration/Other" of the Webapp check "Prefer Web Inf Classes"
    * For Solaris you **HAVE TO** go to "Configuration/Files" and set the "Reload Period" to -1 because of performance issues

Cocoon 2.2.0 DEV and BEA Weblogic 8.1

Here are also different isues regarding Rhino and Xalan - for Xalan it is necessary to switch to ParanoidCocoonServlet and set init-classloader to true. Here also the Rhino problems are only solved in the latest SVN Version, not with 2.1.5.1! To get around the Xalan problems, you also can use xsltc instead of the xalan Transformer, then you can leave the ParanoidCocoonServlet and init-classloader settings. For Cocoon in general you also have to create a weblogic.xml here, because "Prefer Web Inf Classes" is no more accessible through the console; here a small example:

```
<!DOCTYPE weblogic-web-app PUBLIC "-//BEA Systems, Inc.//DTD Web Application 8.1//EN" "http://www.bea.com
/servers/wls810/dtd/weblogic810-web-jar.dtd">
<weblogic-web-app>
        <jsp-descriptor>
                <jsp-param>
                        <param-name>keepgenerated</param-name>
                        <param-value>true</param-value>
                </jsp-param>
        </jsp-descriptor>
        <container-descriptor>
                <prefer-web-inf-classes>true</prefer-web-inf-classes>
        </container-descriptor>
        <charset-params>
                <input-charset>
                        <resource-path>/*</resource-path>
                        <java-charset-name>UTF-8</java-charset-name>
                </input-charset>
        </charset-params>
        <context-root>/</context-root>
</weblogic-web-app>
```

AUTHOR: Sascha-Matthias Kulawik, AUTHOR-CONTACT: sascha@kulawik.de

---

Portal Framework with Cocoon 2.x.x and BEA Weblogic 5.x/6.x/7.x

Additionally there is a Problem with the Thread-Handling of WLS. The Portal Framework needs to be patched to use the correct Class-Loader for newly created Threads.

These changes are `"org.apache.cocoon.webapps.portal.components.PortalManagerImpl.java"`

```
~ SNIP

            CopletThread copletThread = new CopletThread();
            Thread theThread = new Thread(copletThread);

            // Patch Weblogic 5/6/7
            try {
              theThread.setContextClassLoader(this.getClass().getClassLoader());
            } catch (Exception e){}
            // Patch

            loadedCoplet[6] = copletThread;
            copletThread.init(copletID,
                              ContextHelper.getObjectModel(this.componentContext),
                              this.getLogger(),
                              ContextHelper.getResponse(this.componentContext),
                              loadedCoplet,
                              this.manager,
                              this.resolver,
                              this.xpathProcessor);
            theThread.start();
            Thread.yield();

~ SNAP
```