

Wrapping a Natural for Ajax Application as a Servlet

In a production environment, it is inconvenient to start an application with a URL such as the following:

http://myappserver:4711/cisnatural/servlet/StartCISPage?PAGEURL=%2Fcisnatural%2FNatLogon.html&xciParameters.natserver=mywebio&xciParameters.natprog=mwo.sh&xciParameters.natport=4712&xciParameters.natparam=stack%3D%28logon+SYSEXNJX%3BMENU-NJX%3BFIN%29

The URL can be shortened by defining a corresponding session profile in the configuration tool. For example:

http://myappserver:4711/cisnatural/servlet/StartCISPage?PAGEURL=%2Fcisnatural%2FNatLogon.html&xciParameters.natsession=DemoApplication

However, this shortened URL is still not practical for security reasons because end users should not be allowed to access the generic servlet `StartCISPage`.

When you define a dedicated servlet for each application, you can easily define the security constraints for each application in the file *web.xml* (for further information on this file, see *Configuring Security*). With the servlet `com.softwareag.cis.server.StartCISPageWithParams`, you define a wrapper servlet for a given Natural for Ajax application so that the application can later be invoked with a URL such as the following:

http://myappserver:4711/cisnatural/servlet/DemoApplication

The following example shows how you define an application as a servlet in the file *web.xml*.

```
<servlet id="DemoApplication">
  <servlet-name>DemoApplication</servlet-name>
  <display-name>DemoApplication</display-name>
  <servlet-class>com.softwareag.cis.server.StartCISPageWithParams</servlet-class>
  <load-on-startup>1</load-on-startup>
  <init-param id="OVERWRITE">
    <param-name>OVERWRITE</param-name>
    <param-value>>false</param-value>
  </init-param>
  <init-param id="PAGEURL">
    <param-name>PAGEURL</param-name>
    <param-value>/cisnatural/NatLogon.html</param-value>
  </init-param>
  <init-param id="xciParameters.natsession">
    <param-name>xciParameters.natsession</param-name>
    <param-value>Local</param-value>
  </init-param>
  <init-param id="xciParameters.natserver">
    <param-name>xciParameters.natserver</param-name>
    <param-value>localhost</param-value>
  </init-param>
  <init-param id="xciParameters.natport">
    <param-name>xciParameters.natport</param-name>
    <param-value>2900</param-value>
  </init-param>
  <init-param id="xciParameters.natparamext">
    <param-name>xciParameters.natparamext</param-name>
    <param-value>STACK=(LOGON SYSEXNJX;MENU-NJX;FIN)</param-value>
  </init-param>
</servlet>
```

You can omit the parameters `xciParameters.natserver`, `xciParameters.natport` and `xciParameters.natparamext` if the corresponding values are defined in the session definition that is referenced in `xciParameters.natsession`. This the recommended way, because the settings can thus be changed in the configuration tool at any time without the need to adapt the file *web.xml*.

To complete the definition, you define a corresponding servlet mapping in the file *web.xml*:

```
<servlet-mapping>
  <servlet-name>DemoApplication</servlet-name>
  <url-pattern>/servlet/DemoApplication</url-pattern>
</servlet-mapping>
```

As the last step, you exchange the servlet class `com.softwareag.cis.server.StartCISPage` with a different servlet class, namely `com.softwareag.cis.server.StartCISPageInSession`. This servlet class cannot be called directly; it can only be called in an Application Designer session which is already active. Therefore, each attempt to start an arbitrary - not wrapped - application by just building a URL based on `StartCISPage` will result in an error message.

To exchange the servlet class, you change the following in the file *web.xml*

```
<servlet id="StartCISPage">
  <servlet-name>StartCISPage</servlet-name>
  <display-name>StartCISPage</display-name>
  <servlet-class>com.softwareag.cis.server.StartCISPage</servlet-class>
  <load-on-startup>1</load-on-startup>
</servlet>
```

to

```
<servlet id="StartCISPage">
  <servlet-name>StartCISPage</servlet-name>
  <display-name>StartCISPage</display-name>
  <servlet-class>com.softwareag.cis.server.StartCISPageInSession</servlet-class>
  <load-on-startup>1</load-on-startup>
</servlet>
```

For a workplace application (see *Working with Workplaces* in the *Natural for Ajax* documentation), the definition looks slightly different. You do not define the start-up page of the workplace directly in the `PAGEURL` parameter. Instead, you define an intermediate navigation page `/HTMLBasedGUI/com.softwareag.cis.util.navigatetopage.html`. The start-up page of the workplace is specified in the `navPage` parameter. The navigation page makes sure that an Application Designer session is created before it navigates to the start-up page of the workplace.

```
<servlet id="WorkplaceDemo">
  <servlet-name>WorkplaceDemo</servlet-name>
  <display-name>WorkplaceDemo</display-name>
  <servlet-class>com.softwareag.cis.server.StartCISPageWithParams</servlet-class>
  <load-on-startup>1</load-on-startup>
  <init-param id="OVERWRITE">
    <param-name>OVERWRITE</param-name>
    <param-value>>false</param-value>
  </init-param>
  <init-param id="PAGEURL">
    <param-name>PAGEURL</param-name>
    <param-value>/HTMLBasedGUI/com.softwareag.cis.util.navigatetopage.html</param-value>
  </init-param>
  <init-param id="navPage">
```

```
<param-name>navPage</param-name>  
  <param-value>/njxdemos/wpdynworkplace.html</param-value>  
</init-param>  
</servlet>
```

In this case, the corresponding servlet mapping is defined as follows:

```
<servlet-mapping>  
  <servlet-name>WorkplaceDemo</servlet-name>  
  <url-pattern>/servlet/WorkplaceDemo</url-pattern>  
</servlet-mapping>
```

Note:

For further information on the above mentioned servlets, see the Java API documentation which is provided with Application Designer.