

## HstComponentInvokerImpl

org.hippoecm.hst.core.container.HstComponentInvokerImpl					
Coverage	Dependencies	Duplications	LCOM4	Sources	Violations
0	12	16	1	0	
35	* @version \$Id: HstComponentInvokerImpl.java 21975 2010-02-26 17:14:32Z aschrijvers \$				
36	*/				
37	<b>public class</b> HstComponentInvokerImpl <b>implements</b> HstComponentInvoker {				
38	<b>private final static</b> Logger log = LoggerFactory.getLogger(HstComponentInvokerImpl.class);				
39	↓ <b>Modifier Order</b> : 'static' modifier out of order with the JLS suggestions.				
40	<b>protected boolean</b> exceptionThrowable;				
41	↑ <b>Visibility Modifier</b> : Variable 'exceptionThrowable' must be private and have accessor methods.				
42	<b>protected String</b> errorRenderPath;				
43	↑ <b>Visibility Modifier</b> : Variable 'errorRenderPath' must be private and have accessor methods.				
44	<b>public void</b> setExceptionThrowable( <b>boolean</b> exceptionThrowable) {				
45	<b>this</b> .exceptionThrowable = exceptionThrowable;				
46	}				
47	<b>public void</b> setErrorRenderPath(String errorRenderPath) {				
48	<b>this</b> .errorRenderPath = errorRenderPath;				
49	}				
50	<b>public void</b> invokeAction(HstContainerConfig requestContainerConfig, ServletRequest servletRequest, ServletResponse servletResponse) <b>throws</b> ContainerExcept				
51	↑ <b>Cyclomatic Complexity</b> : Cyclomatic Complexity is 12 (max allowed is 10).				
52	HstRequest hstRequest = (HstRequest) servletRequest;				
53	↑ <b>Dodgy - Unchecked/unconfirmed cast</b> : Unchecked/unconfirmed cast from javax.servlet.ServletRequest to org.hippoecm.hst.core.component.HstRequest in org.hippoecm.hst.core.container.HstComponentInvokerImpl.invokeAction(HstContainerConfig, ServletRequest, ServletResponse)				
54	HstResponse hstResponse = (HstResponse) servletResponse;				
55	↑ <b>Dodgy - Unchecked/unconfirmed cast</b> : Unchecked/unconfirmed cast from javax.servlet.ServletResponse to org.hippoecm.hst.core.component.HstResponse in org.hippoecm.hst.core.container.HstComponentInvokerImpl.invokeAction(HstContainerConfig, ServletRequest, ServletResponse)				
56	HstComponentWindow window = ((HstRequestImpl) hstRequest).getComponentWindow();				
57	HstComponent component = window.getComponent();				
58	<b>if</b> (component != null) {				
59	ClassLoader currentClassLoader = switchToContainerClassLoader(requestContainerConfig);				
60	}				
72	window.addComponentException(e);				
73	<b>if</b> (log.isDebugEnabled()) {				
74	log.warn("Component exception caught on window " + window.getName() + " with component " + component.getClass().getName() + ": " + e.toString				
75	↑ <b>Void Duplicate Literals</b> : The String literal " with component " appears 12 times in this file; the first occurrence is on line 76				
76	↑ <b>Void Duplicate Literals</b> : The String literal "Component exception caught on window " appears 12 times in this file; the first occurrence is on line 76				
77	} <b>else if</b> (log.isWarnEnabled()) {				
78	log.warn("Component exception caught on window " + window.getName() + " with component " + component.getClass().getName() + ": {}", e.toS				
79	↑ <b>Void Duplicate Literals</b> : The String literal ":" {}" appears 6 times in this file; the first occurrence is on line 78				
80	} <b>catch</b> (Exception e) {				
81	<b>if</b> ( <b>this</b> .exceptionThrowable) {				
82	<b>throw new</b> HstComponentException(e);				
83	}				
84	}				
85	window.addComponentException( <b>new</b> HstComponentException(e));				
86	<b>if</b> (log.isDebugEnabled()) {				
87	log.warn("Component exception caught: " + e.toString(), e);				
88	↑ <b>Void Duplicate Literals</b> : The String literal "Component exception caught: " appears 4 times in this file; the first occurrence is on line 88				
89	} <b>else if</b> (log.isWarnEnabled()) {				
90	log.warn("Component exception caught: {}", e.toString());				
91	↑ <b>Void Duplicate Literals</b> : The String literal "Component exception caught: {}" appears 4 times in this file; the first occurrence is on line 90				
92	} <b>finally</b> {				
93	<b>if</b> (currentClassLoader != null) {				
94	Thread.currentThread().setContextClassLoader(currentClassLoader);				
95	}				
96	} <b>finally</b> {				
97	window.addComponentException( <b>new</b> HstComponentException("The component is not available."));				
98	}				
99	}				
100	<b>public void</b> invokeBeforeRender(HstContainerConfig requestContainerConfig, ServletRequest servletRequest, ServletResponse servletResponse) <b>throws</b> Containe				
101	↑ <b>Cyclomatic Complexity</b> : Cyclomatic Complexity is 12 (max allowed is 10).				
102	HstRequest hstRequest = (HstRequest) servletRequest;				
103	↑ <b>Dodgy - Unchecked/unconfirmed cast</b> : Unchecked/unconfirmed cast from javax.servlet.ServletRequest to org.hippoecm.hst.core.component.HstRequest in org.hippoecm.hst.core.container.HstComponentInvokerImpl.invokeBeforeRender(HstContainerConfig, ServletRequest, ServletResponse)				
104	HstResponse hstResponse = (HstResponse) servletResponse;				
105	↑ <b>Dodgy - Unchecked/unconfirmed cast</b> : Unchecked/unconfirmed cast from javax.servlet.ServletResponse to org.hippoecm.hst.core.component.HstResponse in org.hippoecm.hst.core.container.HstComponentInvokerImpl.invokeBeforeRender(HstContainerConfig, ServletRequest, ServletResponse)				
106	HstComponentWindow window = ((HstRequestImpl) hstRequest).getComponentWindow();				
107	HstComponent component = window.getComponent();				
108	<b>if</b> (component != null) {				
109	ClassLoader currentClassLoader = switchToContainerClassLoader(requestContainerConfig);				

```

198     }
199
200     public void invokeBeforeServeResource(HstContainerConfig requestContainerConfig, ServletRequest servletRequest, ServletResponse servletResponse) throws C
    ↑ Cyclomatic Complexity : Cyclomatic Complexity is 11 (max allowed is 10).
201     HstRequest hstRequest = (HstRequest) servletRequest;
    ↑ Dodgy - Unchecked/unconfirmed cast : Unchecked/unconfirmed cast from javax.servlet.ServletRequest to org.hippoecm.hst.core.component.HstRequest in
    org.hippoecm.hst.core.container.HstComponentInvokerImpl.invokeBeforeServeResource(HstContainerConfig, ServletRequest, ServletResponse)
202     HstResponse hstResponse = (HstResponse) servletResponse;
    ↑ Dodgy - Unchecked/unconfirmed cast : Unchecked/unconfirmed cast from javax.servlet.ServletResponse to org.hippoecm.hst.core.component.HstResponse in
    org.hippoecm.hst.core.container.HstComponentInvokerImpl.invokeBeforeServeResource(HstContainerConfig, ServletRequest, ServletResponse)
203     HstComponentWindow window = ((HstRequestImpl) hstRequest).getComponentWindow();
204     HstComponent component = window.getComponent();
205
206     if (component != null) {
207         ClassLoader currentClassLoader = switchToContainerClassLoader(requestContainerConfig);
208
209         window.addComponentException(new HstComponentException("The component is not available.));
210     }
211 }
212
213     public void invokeServeResource(HstContainerConfig requestContainerConfig, ServletRequest servletRequest, ServletResponse servletResponse) throws Contain
    ↑ Cyclomatic Complexity : Cyclomatic Complexity is 12 (max allowed is 10).
214     HstRequest hstRequest = (HstRequest) servletRequest;
    ↑ Dodgy - Unchecked/unconfirmed cast : Unchecked/unconfirmed cast from javax.servlet.ServletRequest to org.hippoecm.hst.core.component.HstRequest in
    org.hippoecm.hst.core.container.HstComponentInvokerImpl.invokeServeResource(HstContainerConfig, ServletRequest, ServletResponse)
215     HstResponse hstResponse = (HstResponse) servletResponse;
    ↑ Dodgy - Unchecked/unconfirmed cast : Unchecked/unconfirmed cast from javax.servlet.ServletResponse to org.hippoecm.hst.core.component.HstResponse in
    org.hippoecm.hst.core.container.HstComponentInvokerImpl.invokeServeResource(HstContainerConfig, ServletRequest, ServletResponse)
216     HstComponentWindow window = ((HstRequestImpl) hstRequest).getComponentWindow();
217     HstComponent component = window.getComponent();
218
219     String dispatchUrl = ((HstResourceResponseImpl) hstResponse).getServeResourcePath();
    ↑ Dodgy - Unchecked/unconfirmed cast : Unchecked/unconfirmed cast from org.hippoecm.hst.core.component.HstResponse to org.hippoecm.hst.core.component.HstReso
    org.hippoecm.hst.core.container.HstComponentInvokerImpl.invokeServeResource(HstContainerConfig, ServletRequest, ServletResponse)
220
221     if (dispatchUrl == null) {
222         dispatchUrl = window.getServeResourcePath();
223     }
224
225     renderErrorInformation(requestContainerConfig, servletRequest, servletResponse, window);
226 }
227
228     protected void invokeDispatcher(HstContainerConfig requestContainerConfig, ServletRequest servletRequest, ServletResponse servletResponse, String dispatch
    ↑ Signature Declare Throws Exception : A method/constructor shouldn't explicitly throw java.lang.Exception
229     RequestDispatcher disp = null;
230
231     if (dispatchUrl != null) {
232         if (log.isDebugEnabled()) {
233             log.debug("Invoking dispatcher of url: {}", dispatchUrl);
234         }
235     }
236
237     private ClassLoader switchToContainerClassLoader(HstContainerConfig requestContainerConfig) {
238         ClassLoader containerClassLoader = requestContainerConfig.getContextClassLoader();
239         ClassLoader currentClassLoader = Thread.currentThread().getContextClassLoader();
240
241         if (containerClassLoader != currentClassLoader) {
242             ↑ Compare Objects With Equals : Use equals() to compare object references.
243             Thread.currentThread().setContextClassLoader(containerClassLoader);
244             return currentClassLoader;
245         } else {
246             return null;
247         }
248     }
249
250     servletRequest.setAttribute(ContainerConstants.HST_REQUEST, hstRequest);
251     servletRequest.setAttribute(ContainerConstants.HST_RESPONSE, hstResponse);
252 }
253
254     private void removeHstObjectAttributesForServlet(ServletRequest servletRequest, HstRequest hstRequest, HstResponse hstResponse) {
255     ↑ Unused formal parameter : Avoid unused method parameters such as 'hstRequest'.
256     ↑ Unused formal parameter : Avoid unused method parameters such as 'hstResponse'.
257
258     // Removes hst request/response into attribute map after dispatching
259     servletRequest.removeAttribute(ContainerConstants.HST_REQUEST);
260     servletRequest.removeAttribute(ContainerConstants.HST_RESPONSE);
261 }
262
263
264

```