jax-ws   JAX_WS-1145

Memory leak on ThreadLocalContainerResolver

Details

Type:	● Bug	Status:	OPEN
Priority:	↑ Major	Resolution:	Unresolved
Affects Version/s:	2.2.7, 2.2.8	Fix Version/s:	None
Component/s:	runtime		
Labels:	None		
Environment:	Tomacat 7.0.50, Weblogic 12.1.2		

Description

On com.sun.xml.ws.client.dispatch.DispatchImpl.invoke(T), an invocation to containers.get on thread local variable is done but there is no invocation to the remove method to detach the thread local variable from the current thread.
I think that the remove should be called inside the com.sun.xml.ws.api.server.ThreadLocalContainerResolver.exitContainer(Container) method instead of using the set.
I attach a screenshot where with MAT you can see the leak (we have already undeployed and removed the application from WLS12c)

Activity

All	**Comments**	Work Log	History	Activity

👤 grifo85 added a comment - 14/Mar/14 3:17 PM

I'm not able to attach the image, sorry. If you need more information, tell me how I can send you the details.
thanks a lot.

👤 hartmannt added a comment - 04/Sep/14 11:40 AM

I have the same issue with apache-tomcat-7.0.55 and JAXWS 2.2.8 (Java is 1.7.0_51b13 on Win7 64bit)

```
Aug 28, 2014 6:03:28 PM org.apache.catalina.loader.WebappClassLoader checkThreadLocalMapForLeaks
SEVERE: The web application [/xxx] created a ThreadLocal with key of type [com.sun.xml.ws.api.server.ThreadLocalContainerReso
(value [com.sun.xml.ws.api.server.ThreadLocalContainerResolver$1@77d943c1])
and a value of type [com.sun.xml.ws.api.server.Container.NoneContainer]
(value [com.sun.xml.ws.api.server.Container$NoneContainer@3a3607e0]) but failed to remove it when the web
application was stopped. Threads are going to be renewed over time to try and avoid a probable memory leak.
```

People

Assignee:

👤 Unassigned

Reporter:

👤 grifo85

Votes:

0   Vote for this issue

Watchers:

3   Start watching this issue

Dates

Created:

14/Mar/14 3:14 PM

Updated:

04/Sep/14 11:40 AM

Agile

View on Board