

Hey all - hoping someone can point me in the right direction here. I've defined a connection pool in the ESB runtime for a MariaDB database (hosted on Amazon RDS, if it matters). The datasource XML that I've defined and deployed in the runtime looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0"
  default-activation="lazy">
  <bean id="CDHDataSource" class="com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource">
    <property name="url" value="jdbc:mysql:///?noDatetimeStringSync=true"/>
    <property name="user" value="xxxxxxx"/>
    <property name="password" value="xxxxxxx"/>
  </bean>
  <bean id="cdh" class="org.apache.commons.dbcp.datasources.SharedPoolDataSource" destroy-method="close">
    <property name="connectionPoolDataSource" ref="CDHDataSource"/>
    <property name="maxActive" value="250"/>
    <property name="maxIdle" value="50"/>
    <property name="maxWait" value="10000"/>
    <property name="validationQuery" value="select 1"/>
  </bean>
  <service ref="cdh" interface="javax.sql.DataSource">
    <service-properties>
      <entry key="osgi.jndi.service.name" value="jdbc/cdh"/>
    </service-properties>
  </service>
</blueprint>
```

Note that i've stripped out the related database credentials information for obvious reasons...

The connection pool works and services that I deploy to the runtime use it fine. My issue is that overnight, when there is no activity (especially in non-production environments), the connection pool sessions become inactive. When that occurs, calls to the service using the connection pool begin to fail with errors such as "Could not send query: Connection timed out" or "Could not send query: Last package not finished". So, I understand that the issue is that the database is timing out the connections due to inactivity. I'm trying to update my connection pool properties to account for this, and that's where i'm having issues.

Based on the documentation I've read regarding the [SharedPoolDataSource](#) class, I believe I should be able to add some properties to the bean that will address my issue. I've tried to update my connection pool definition as follows. Specifically, I believe "testOnBorrow" should address my problem. I added a few other properties based on how I think I want the connection pool to behave.

```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0"
  default-activation="lazy">
  <bean id="CDHDataSource" class="com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource">
    <property name="url" value="jdbc:mysql:///?noDatetimeStringSync=true"/>
    <property name="user" value="xxxxxxxxxx"/>
    <property name="password" value="xxxxxxxxxx"/>
  </bean>
  <bean id="cdh" class="org.apache.commons.dbcp.datasources.SharedPoolDataSource" destroy-method="close">
    <property name="connectionPoolDataSource" ref="CDHDataSource"/>
    <property name="maxActive" value="250"/>
    <property name="maxIdle" value="50"/>
    <property name="maxWait" value="10000"/>
    <property name="validationQuery" value="select 1"/>
  <property name="testOnBorrow" value="true"/>
  <property name="testWhileIdle" value="true"/>
    <property name="timeBetweenEvictionRunsMillis" value="600000"/>
    <property name="numTestsPerEvictionRun" value="50"/>
  </bean>
  <service ref="cdh" interface="javax.sql.DataSource">
```

```
<service-properties>
  <entry key="osgi.jndi.service.name" value="jdbc/cdh"/>
</service-properties>
</service>
```

When I deploy this update to the connection pool, it deploys without error. However, as soon as I call my service that uses this connection pool, I'm seeing the following errors in the ESB runtime log. If I remove the testOnBorrow property, I get a similar error for each of the other properties that I've added. However, these are all properties that I believe should exist for the SharedPoolDataSource class, based on the documentation. Obviously, I must be misunderstanding something about this, but it's not obvious to me. Any thoughts or help would be much appreciated... I'm using Talend ESB 6.1.1 currently.

```
20:37:31,649 | ERROR | pool-36-thread-9 | lueprint.container.ServiceRecipe 309 | 13 - org.apache.aries.blueprint.core - 1.4.5 | Error retrieving service from ServiceRecipe
org.osgi.service.blueprint.container.ComponentDefinitionException: Unable to find property descriptor testOnBorrow on class org.apache.commons.dbcp.datasources.SharedPoolDataSource
  at org.apache.aries.blueprint.container.BeanRecipe.getPropertyDescriptor(BeanRecipe.java:976)
  at org.apache.aries.blueprint.container.BeanRecipe.setProperty(BeanRecipe.java:958)
  at org.apache.aries.blueprint.container.BeanRecipe.setProperties(BeanRecipe.java:929)
  at org.apache.aries.blueprint.container.BeanRecipe.setProperties(BeanRecipe.java:910)
  at org.apache.aries.blueprint.container.BeanRecipe.internalCreate2(BeanRecipe.java:844)
  at org.apache.aries.blueprint.container.BeanRecipe.internalCreate(BeanRecipe.java:811)
  at org.apache.aries.blueprint.di.AbstractRecipe$1.call(AbstractRecipe.java:79)
  at java.util.concurrent.FutureTask.run(FutureTask.java:266)
  at org.apache.aries.blueprint.di.AbstractRecipe.create(AbstractRecipe.java:88)
  at org.apache.aries.blueprint.di.RefRecipe.internalCreate(RefRecipe.java:62)
  at org.apache.aries.blueprint.di.AbstractRecipe.create(AbstractRecipe.java:106)
  at org.apache.aries.blueprint.container.ServiceRecipe.createService(ServiceRecipe.java:285)
  at org.apache.aries.blueprint.container.ServiceRecipe.internalGetService(ServiceRecipe.java:252)
  at org.apache.aries.blueprint.container.ServiceRecipe.getService(ServiceRecipe.java:365)
  at org.apache.aries.blueprint.container.ServiceRecipe$TriggerServiceFactory.getService(ServiceRecipe.java:541)
  at org.eclipse.osgi.internal.serviceregistry.ServiceFactoryUse$1.run(ServiceFactoryUse.java:212)
  at java.security.AccessController.doPrivileged(Native Method)
  at org.eclipse.osgi.internal.serviceregistry.ServiceFactoryUse.factoryGetService(ServiceFactoryUse.java:210)
  at org.eclipse.osgi.internal.serviceregistry.ServiceFactoryUse.getService(ServiceFactoryUse.java:111)
  at org.eclipse.osgi.internal.serviceregistry.ServiceConsumer$2.getService(ServiceConsumer.java:45)
  at org.eclipse.osgi.internal.serviceregistry.ServiceRegistrationImpl.getService(ServiceRegistrationImpl.java:496)
  at org.eclipse.osgi.internal.serviceregistry.ServiceRegistry.getService(ServiceRegistry.java:461)
  at org.eclipse.osgi.internal.framework.BundleContextImpl.getService(BundleContextImpl.java:619)
  at org.apache.aries.blueprint.container.AbstractServiceReferenceRecipe.getServiceSecurely(AbstractServiceReferenceRecipe.java:235)
  at org.apache.aries.blueprint.container.ReferenceRecipe.getService(ReferenceRecipe.java:246)
  at org.apache.aries.blueprint.container.ReferenceRecipe.access$000(ReferenceRecipe.java:56)
  at org.apache.aries.blueprint.container.ReferenceRecipe$ServiceDispatcher.call(ReferenceRecipe.java:306)
  at Proxyc084125b_f043_4c58_8966_b5dacf7c56bf.getConnection(Unknown Source)
  at routines.system.TalendDataSource.getConnection(TalendDataSource.java:16)
  at fw_esb.publish_event_2_3.publish_event.tAssertCatcher_1Process(publish_event.java:3306)
  at fw_esb.publish_event_2_3.publish_event.tRESTRequest_1_In_error(publish_event.java:1186)
  at fw_esb.publish_event_2_3.publish_event.tRESTRequest_1_Loop_error(publish_event.java:1170)
  at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
  at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
  at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
  at java.lang.reflect.Method.invoke(Method.java:498)
  at fw_esb.publish_event_2_3.publish_event$TalendException.printStackTrace(publish_event.java:391)
  at fw_esb.publish_event_2_3.publish_event.tRESTRequest_1_LoopProcess(publish_event.java:12637)
  at fw_esb.publish_event_2_3.publish_event.runJobInTOS(publish_event.java:13210)
  at org.talend.esb.job.controller.internal.SimpleJobTask$JobRunner.run(SimpleJobTask.java:81)
  at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511)
```

at java.util.concurrent.FutureTask.run(FutureTask.java:266)  
at org.talend.esb.job.controller.internal.SimpleJobTask.run(SimpleJobTask.java:63)  
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)  
at java.util.concurrent.ThreadPoolExecutor\$Worker.run(ThreadPoolExecutor.java:617)  
at java.lang.Thread.run(Thread.java:745)