RED HAT®
CONSULTING

Oracle Cartridge EAP Datasource Configuration Guide

PREPARED FOR - FRIT

Table of Contents

1. Synopsis

The purpose of this document is to present the changes required to the Red Hat JBoss Enterprise Application Platform 6 OpenShift cartridge to add a datasource that can connect to the newly Provisioned Oracle DB Tenant provided via the Oracle Cartridge.

2. Explanation of Approach

The approach will be to first install the Oracle OJDBC6 driver as a module, add the driver to the standalone.xml, and then add a datasource definition that calls upon the environment variable set by the Oracle cartridge on gear installation similar to the mysql and postgresql cartridges.

3. Install Driver as JBoss Module

It is assumed that you already have the correct ojdbc6.jar driver file retrieved from either your database team, or the Oracle website.

1. Place the JDBC driver jar (ojdbc6.jar) in to '/usr/share/java'

```
mv ojdbc6.jar /usr/share/java/
```

Make sure all users can read the jar file

```
chmod 644 /usr/share/java/ojdbc6.jar
```

3. Create a directory under $JBOSS_HOME/modules for the module to reside:

```
mkdir -p $JBOSS_HOME/modules/com/oracle/jdbc/main
```

4. Make a logical symbolic link to the driver file

```
ls -s /usr/share/java/ojdbc6.jar $JBOSS_HOME/modules/com/oracle/jdbc/main/ojdbc6.jar
```

5. Create a module configuration file $JBOSS_HOME/modules/com/oracle/jdbc/main/module.xml containing the following:

```
<module xmlns="urn:jboss:module:1.1" name="com.oracle.jdbc">
  <resources>
    <resource-root path="ojdbc6.jar"/>
  </resources>
  <dependencies>
    <module name="javax.api"/>
    <module name="javax.transaction.api"/>
  </dependencies>
</module>
```

Make sure all users can read the odule

```
chmod 644 -R $JBOSS_HOME/modules/com/oracle/jdbc/main
```

Note that the jdbc driver jar must contain a META-INF/services/java.sql.Driver text file that specifies the jdbc Driver, otherwise the Java service provider mechanism used to load the driver will not work. From the main/common vendors only Informix does not have this out of the box at time of writing.

4. Modify standalone.xml

You will need to modify the $CARTRIDGE_DIR/version/shared/standalone/configuration/standalone.xml file to add a Oracle driver and Datasource definition.

Edit $CARTRIDGE_DIR/version/shared/standalone/configuration/config/standalone.xml file and add the following to make the newly created module available as a driver. This will need to be added within the enclosing '<drivers></drivers>' tags.

```
<driver name="oracle" module="com.oracle.jdbc">
    <xa-datasource-class>oracle.jdbc.xa.client.OracleXADataSource</xa-datasource-
class>
</driver>
```

Edit $CARTRIDGE_DIR/version/shared/standalone/configuration/config/standalone.xml file and add the following datasource definition within the '<datasources></datasources>' tags, that makes use of environment variables set by the Oracle Cartridge to point to the newly provisioned Oracle Tenant DB.

```
<datasource jndi-name="java:jboss/datasources/OracleDS" pool-name="OracleDS"
enabled="${oracle.enabled}" use-java-context="true">
    <connection-
url>jdbc:oracle:thin:@//${env.OPENSHIFT_ORACLE_DB_REMOTE_HOST}:${env.OPENSHIFT_ORACLE_
DB_REMOTE_PORT}/${env.OPENSHIFT_ORACLE_DB_TENANT_ID}</connection-url>
    <driver>oracle</driver>
    <pool>
        <min-pool-size>10</min-pool-size>
        <max-pool-size>20</max-pool-size>
        <prefill>true</prefill>
    </pool>
    <security>
        <user-name>${env.OPENSHIFT_ORACLE_DB_USERNAME}</user-name>
        <password>${env.OPENSHIFT_ORACLE_DB_PASSWORD}</password>
    </security>
    <validation>
        <validate-on-match>true</validate-on-match>
        <valid-connection-checker class-
name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleValidConnectionChecker"></va
lid-connection-checker>
        <stale-connection-checker class-
name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleStaleConnectionChecker"></st
ale-connection-checker>
        <exception-sorter class-
name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter"></exception
-sorter>
    </validation>
</datasource>
```

Note that the datasource is not enabled by default. If it were enabled, EAP6 would fail to start if the Oracle add-on cartridge had not been added to the application. Therefore, it is set to false; after the Oracle add-on cartridge is added to the application the developer will need to enable the datasource by setting the flag to "true"

The resulting datasources subsystem section should look similar to the below.

```
...
<subsystem xmlns="urn:jboss:domain:datasources:1.1">
    <datasources>
        <datasource jndi-name="java:jboss/datasources/OracleDS" pool-name="OracleDS"
enabled="${oracle.enabled}" use-java-context="true">
            <connection-
url>jdbc:oracle:thin:@//${env.OPENSHIFT_ORACLE_DB_REMOTE_HOST}:${env.OPENSHIFT_ORACLE_
DB_REMOTE_PORT}/${env.OPENSHIFT_ORACLE_DB_TENANT_ID}</connection-url>
            <driver>oracle</driver>
            <pool>
                <min-pool-size>10</min-pool-size>
```

```
                <max-pool-size>20</max-pool-size>
                <prefill>true</prefill>
            </pool>
            <security>
                <user-name>${env.OPENSHIFT_ORACLE_DB_USERNAME}</user-name>
                <password>${env.OPENSHIFT_ORACLE_DB_PASSWORD}</password>
            </security>
            <validation>
                <validate-on-match>true</validate-on-match>
                <valid-connection-checker class-
name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleValidConnectionChecker"></va
lid-connection-checker>
                <stale-connection-checker class-
name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleStaleConnectionChecker"></st
ale-connection-checker>
                <exception-sorter class-
name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter"></exception
-sorter>
            </validation>
        </datasource>
        <datasource jndi-name="java:jboss/datasources/ExampleDS" enabled="true" use-
java-context="true" pool-name="H2DS">
            <connection-url>jdbc:h2:${jboss.server.data.dir}/test;DB_CLOSE_DELAY=-
1</connection-url>
            <driver>h2</driver>
            <security>
                <user-name>sa</user-name>
                <password>sa</password>
            </security>
        </datasource>
        <datasource jndi-name="java:jboss/datasources/MySQLDS"
enabled="${mysql.enabled}" use-java-context="true" pool-name="MySQLDS" use-ccm="true">
            <connection-
url>jdbc:mysql://${env.OPENSHIFT_MYSQL_DB_HOST}:${env.OPENSHIFT_MYSQL_DB_PORT}/${env.O
PENSHIFT_APP_NAME}</connection-url>
            <driver>mysql</driver>
            <security>
                <user-name>${env.OPENSHIFT_MYSQL_DB_USERNAME}</user-name>
                <password>${env.OPENSHIFT_MYSQL_DB_PASSWORD}</password>
            </security>
            <validation>
                <check-valid-connection-sql>SELECT 1</check-valid-connection-sql>
                <background-validation>true</background-validation>
                <background-validation-millis>60000</background-validation-millis>
            </validation>
            <pool>
                <flush-strategy>IdleConnections</flush-strategy>
                <allow-multiple-users/>
            </pool>
        </datasource>
        <drivers>
```

```
            <driver name="oracle" module="com.oracle.jdbc">
                <xa-datasource-class>oracle.jdbc.xa.client.OracleXADataSource</xa-
datasource-class>
            </driver>
            <driver name="h2" module="com.h2database.h2">
                <xa-datasource-class>org.h2.jdbcx.JdbcDataSource</xa-datasource-class>
            </driver>
            <driver name="mysql" module="com.mysql.jdbc">
                <xa-datasource-
class>com.mysql.jdbc.jdbc2.optional.MysqlXADataSource</xa-datasource-class>
            </driver>
        </drivers>
    </datasources>
</subsystem>
...
```

5. Modify standalone.conf

You will need to modify the $CARTRIDGE_DIR/version/shared/bin/standalone.conf file to add logic to set '${mysql.enabled}'

Edit $CARTRIDGE_DIR/version/shared/standalone/configuration/config/standalone.xml file and add the code below to set ORACLE_ENABLED after the section setting POSTGRESQL_ENABLED

```
ORACLE_ENABLED="false"
if [ -n "$OPENSHIFT_ORACLE_DB_REMOTE_HOST" ]
then
    ORACLE_ENABLED="true"
fi
```

The result should look like the below:

```
POSTGRESQL_ENABLED="false"
if [ -n "$OPENSHIFT_POSTGRESQL_DB_URL" ]
then
    POSTGRESQL_ENABLED="true"
fi

ORACLE_ENABLED="false"
if [ -n "$OPENSHIFT_ORACLE_DB_REMOTE_HOST" ]
then
    ORACLE_ENABLED="true"
fi

max_threads=$(ulimit -u)
```

Now replace the below sed section:

```
cat <<EOF > /tmp/sed_${systime}.sh
sed -i -e "s/\\\${mysql.enabled}/$MYSQL_ENABLED/g" \
       -e "s/\\\${postgresql.enabled}/$POSTGRESQL_ENABLED/g" \
       -e "s/\\\${messaging.thread.pool.max.size}/$messaging_thread_pool_max_size/g" \
       -e
"s/\\\${messaging.scheduled.thread.pool.max.size}/$messaging_scheduled_thread_pool_max
_size/g" \
       -e "s/\\\${env.OPENSHIFT_INTERNAL_IP}/${OPENSHIFT_JBOSSEAP_IP}/g" \
       ${sed_replace_env} \
       ${CART_DIR}/standalone/configuration/standalone.xml > /dev/null 2>&1
EOF
```

With:

```
cat <<EOF > /tmp/sed_${systime}.sh
sed -i -e "s/\\\${oracle.enabled}/$ORACLE_ENABLED/g" \
       -e "s/\\\${mysql.enabled}/$MYSQL_ENABLED/g" \
       -e "s/\\\${postgresql.enabled}/$POSTGRESQL_ENABLED/g" \
       -e "s/\\\${messaging.thread.pool.max.size}/$messaging_thread_pool_max_size/g" \
       -e
"s/\\\${messaging.scheduled.thread.pool.max.size}/$messaging_scheduled_thread_pool_max
_size/g" \
       -e "s/\\\${env.OPENSHIFT_INTERNAL_IP}/${OPENSHIFT_JBOSSEAP_IP}/g" \
       ${sed_replace_env} \
       ${CART_DIR}/standalone/configuration/standalone.xml > /dev/null 2>&1
EOF
```

6. Reference Information

- OpenShift Oracle Cartridge
- How to configure datasource settings in EAP 6