

HP Codar

Software Version 1.50

Configuring HP Codar to Work with Oracle RAC

Contents

Scope and purpose of this document..... 2

Oracle RAC 2

HP Codar database configuration for non-Oracle RAC 2

HP Codar database configuration for Oracle RAC 3

Upgrading an HP Codar database configuration for Oracle RAC..... 5

For More Information 6

© Copyright 2015 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Restricted rights legend: Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Microsoft and Windows are U.S. registered trademarks of Microsoft Corporation. Oracle and Java are registered trademarks of Oracle and/or its affiliates.

Document release date: July 2015

Software release date: July 2015



Scope and purpose of this document

Enterprise applications such as HP Codar are usually deployed in high-availability database environments. This document describes Oracle Real Application Clusters (Oracle RAC) configuration for HP Codar. This document does not cover how to install HP Codar nor how to install and configure Oracle RAC.

Oracle RAC

Oracle RAC provides clustering and high availability for Oracle database environments. In an Oracle RAC environment, two or more database instances concurrently access a single database. A typical Oracle RAC environment includes the instances running on different physical machines with a Single Client Access Name (SCAN) that allows clients to access the database using a single hostname/IP address, instead of specifying the hostnames/IP addresses for each of the instances in the connection URL.

HP Codar database configuration for non-Oracle RAC

The HP Codar installer prompts users to enter database information (such as the server hostname/IP address, port, Oracle service name, username, and password). This information is stored in a configuration file and used by the Management Console to connect to the database.

In a **standalone environment**, the configuration file is located at:

- **Windows:** %CODAR_HOME%\jboss-as\standalone\configuration\standalone.xml
- **Linux:** \$CODAR_HOME/jboss-as/standalone/configuration/standalone.xml

All the database configuration information is specified in the `<datasource>` element. For example:

```
<!-- csa datasource setup -->
<datasource jndi-name="java:jboss/datasources/csaDS" pool-name="OracleDS">
  <connection-url>jdbc:oracle:thin:@//127.0.0.1:1521/XE</connection-url>
  <driver>oracleDriver</driver>
  <pool>
    <min-pool-size>10</min-pool-size>
    <max-pool-size>200</max-pool-size>
    <prefill>true</prefill>
    <use-strict-min>false</use-strict-min>
    <flush-strategy>FailingConnectionOnly</flush-strategy>
  </pool>
  <security>
    <security-domain>csa-encryption-sec</security-domain>
  </security>
</datasource>
```

NOTE: This example is for non-Oracle RAC environments.

In a **clustered environment**, the configuration file is located at:

- **Windows:** %CODAR_HOME%\jboss-as\standalone\configuration\standalone-full-ha.xml
- **Linux:** \$CODAR_HOME/jboss-as/standalone/configuration/standalone-full-ha.xml

All the database configuration information is specified in the `<datasource>` element. For example:

```
<!-- csa datasource setup -->
<datasource jta="true" jndi-name="java:jboss/datasources/csaDS" pool-name="OracleDS"
enabled="true" use-java-context="true" use-ccm="true">
  <connection-url>jdbc:oracle:thin:@//127.0.0.1:1521/XE</connection-url>
  <driver>oracleDriver</driver>
  <pool>
    <min-pool-size>10</min-pool-size>
    <max-pool-size>200</max-pool-size>
    <prefill>true</prefill>
    <use-strict-min>false</use-strict-min>
    <flush-strategy>FailingConnectionOnly</flush-strategy>
  </pool>
  <security>
    <security-domain>csa-encryption-sec</security-domain>
  </security>
</datasource>
```

NOTE: This example is for non-Oracle RAC environments.

HP Codar database configuration for Oracle RAC

To configure HP Codar to work with Oracle RAC:

1. Install HP Codar. In the screen that asks for database information, enter the SCAN or hostname/IP address of one of the nodes when prompted for the database hostname in the HP Codar installer.

If this information is not correct, the installer is not able to connect to the database and installation cannot proceed until this issue is resolved.

2. After installing HP Codar, navigate to the configuration file. This file is located at:

Standalone Environment

- Windows: %CODAR_HOME%\jboss-as\standalone\configuration\standalone.xml
- Linux: \$CODAR_HOME/jboss-as/standalone/configuration/standalone.xml

Clustered Environment

- Windows: %CODAR_HOME%\jboss-as\standalone\configuration\standalone-full-ha.xml
- Linux: \$CODAR_HOME/jboss-as/standalone/configuration/standalone-full-ha.xml

3. Before modifying the `standalone.xml` or `standalone-full-ha.xml` file, back up the file by making a copy of it.

4. Find the section that configures the HP Codar datasource. An example is shown below:

Standalone Environment

```
<!-- csa datasource setup -->
<datasource jndi-name="java:jboss/datasources/csaDS" pool-name="OracleDS">
  <connection-url>jdbc:oracle:thin:@//127.0.0.1:1521/XE</connection-url>
```

Clustered Environment

```
<!-- csa datasource setup -->
<datasource jta="true" jndi-name="java:jboss/datasources/csaDS" pool-
name="OracleDS" enabled="true" use-java-context="true" use-ccm="true">
  <connection-url>jdbc:oracle:thin:@//127.0.0.1:1521/XE</connection-url>
```

5. Replace the `<connection-url>` element with the following (the highlighted text must be replaced with appropriate values):

```
<connection-url>
jdbc:oracle:thin:@(DESCRIPTION=(LOAD_BALANCE=on) (ADDRESS=(PROTOCOL=TCP)
(HOST=<SCAN_hostname>) (PORT=<port>)) (CONNECT_DATA=(SERVICE_NAME=<servicename>)
(FAILOVER_MODE=(TYPE=SELECT) (METHOD=BASIC) (RETRIES=10) (DELAY=5))))
</connection-url>
```

6. Add the following `<validation>` element to the `<datasource>` element:

```
<validation>
  <check-valid-connection-sql>select 1 from dual</check-valid-connection-sql>
  <validate-on-match>>false</validate-on-match>
  <background-validation>>true</background-validation>
  <use-fast-fail>>false</use-fast-fail>
  <exception-sorter class-name=
    "org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter"/>
</validation>
```

When you are done with these changes, the datasource configuration will look like the following:

NOTE: Elements highlighted below are the elements that you have changed and the bold text should be replaced with appropriate values.

Standalone Environment

```
<!-- csa datasource setup -->
<datasource jndi-name="java:jboss/datasources/csaDS" pool-name="OracleDS">
  <connection-url>
    jdbc:oracle:thin:@(DESCRIPTION=(LOAD_BALANCE=on) (ADDRESS=(PROTOCOL=TCP)
    (HOST=<SCAN_hostname>) (PORT=<port>)) (CONNECT_DATA=(SERVICE_NAME=<servicename>)
    (FAILOVER_MODE=(TYPE=SELECT) (METHOD=BASIC) (RETRIES=10) (DELAY=5))))
  </connection-url>
  <driver>oracleDriver</driver>
  <pool>
    <min-pool-size>10</min-pool-size>
    <max-pool-size>200</max-pool-size>
    <prefill>>true</prefill>
    <use-strict-min>>false</use-strict-min>
    <flush-strategy>FailingConnectionOnly</flush-strategy>
  </pool>
  <security>
    <security-domain>csa-encryption-sec</security-domain>
  </security>
  <validation>
    <check-valid-connection-sql>select 1 from dual</check-valid-connection-sql>
    <validate-on-match>>false</validate-on-match>
    <background-validation>>true</background-validation>
    <use-fast-fail>>false</use-fast-fail>
    <exception-sorter class-name=
      "org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter"/>
  </validation>
</datasource>
```

Clustered Environment

```

<!-- csa datasource setup -->
<datasource jndi-name="java:jboss/datasources/csaDS" pool-name="OracleDS"
enabled="true" jta="true" use-java-context="true" use-ccm="true">
  <connection-url>
    jdbc:oracle:thin:@(DESCRIPTION=(LOAD_BALANCE=on) (ADDRESS=(PROTOCOL=TCP)
    (HOST=<SCAN_hostname>) (PORT=<port>)) (CONNECT_DATA=(SERVICE_NAME=<servicename>)
    (FAILOVER_MODE=(TYPE=SELECT) (METHOD=BASIC) (RETRIES=10) (DELAY=5))))
  </connection-url>
  <driver>oracleDriver</driver>
  <pool>
    <min-pool-size>10</min-pool-size>
    <max-pool-size>200</max-pool-size>
    <prefill>true</prefill>
    <use-strict-min>false</use-strict-min>
    <flush-strategy>FailingConnectionOnly</flush-strategy>
  </pool>
  <security>
    <security-domain>csa-encryption-sec</security-domain>
  </security>
  <validation>
    <check-valid-connection-sql>select 1 from dual</check-valid-connection-sql>
    <validate-on-match>false</validate-on-match>
    <background-validation>true</background-validation>
    <use-fast-fail>false</use-fast-fail>
    <exception-sorter class-name=
    "org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter"/>
  </validation>
</datasource>

```

7. Restart HP Codar.

HP Codar is now connected to the Oracle RAC instance and supports failover. If one of the database instances goes down, HP Codar continues to function normally by connecting to another database instance that is up and running.

Upgrading an HP Codar database configuration for Oracle RAC

After upgrading HP Codar from version 1.00 to version 1.50, you must manually reconfigure HP Codar to work with Oracle RAC by completing the tasks described in the [HP Codar database configuration for Oracle RAC](#) section.

In a standalone environment:

The changes you need to make to the `standalone.xml` file in HP Codar version 1.50 are the same changes you made in the `standalone.xml` file for HP Codar version 1.00. The 1.00 version of the `standalone.xml` file was backed up during the upgrade and you may copy the changes from this file (copy only the changes; do not copy the entire file).

The `standalone.xml` file from HP Codar version 1.00 is backed up to the following location:

- **Windows:** %CODAR_HOME%_CSA_4_50_0_installation\Backup\standalone\standalone.xml
- **Linux:** \$CODAR_HOME/_CSA_4_50_0_installation/Backup/standalone/standalone.xml

In a clustered environment:

The changes you need to make to the `standalone-full-ha.xml` file in HP Codar version 1.00 are the same changes you made in the `standalone-full-ha.xml` file for HP Codar version 1.00. The 1.00 version of the `standalone-full-ha.xml` file was backed up during the upgrade and you may copy the changes from this file (copy only the changes; do not copy the entire file).

The `standalone-full-ha.xml` file from HP Codar version 1.00 is backed up to the following location:

- **Windows:** `%CODAR_HOME%_CSA_4_50_0_installation\Backup\standalone\standalone-full-ha.xml`
- **Linux:** `$CODAR_HOME/_CSA_4_50_0_installation/Backup/standalone/standalone-full-ha.xml`

For More Information

HP software product documentation can be found at <https://softwaresupport.hp.com/>

You need to sign-in or register to use this site. Use the **Search** function at the top of the page to find documentation, whitepapers, and other information sources.

To help us improve our documents, please send feedback to clouddocs@hp.com.