

Oracle® Enterprise Repository

Clustering Guide

10g Release 3 (10.3)

July 2009

Oracle Enterprise Repository Clustering Guide, 10g Release 3 (10.3)

Copyright © 2008, 2009, Oracle. All rights reserved.

Primary Author: Vimmiika Dinesh

Contributing Author: Atturu Chandra Prasad Reddy, Jeff Schieli, Sharon Fay, Phil Reed, Scott Spieker

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Clustering Guide

Table of Contents

- **Clustering and Installation Requirements**
- **Options**
- **Server-side Cache Communication**
- **Supported Application Servers**
- **Installation**
- **Oracle Enterprise Repository Database**
- **Create the Clustered Environment**
- **Deploy and Validate Oracle Enterprise Repository on One Cluster Member**
- **Move the Application Properties to the Database**
- **Configure a cluster.properties File on Each Cluster Member**
- **Validate the Installation**
- **Clustering JVM Parameter for WebLogic Server**
- **Clustered JMS Servers for Advanced Registration Flows**
- **Clustering System Settings for Distributed Environments**

Clustering and Installation Requirements

- Session affinity
- Server-side HTTP cache communication

Options

- **Failover**
 - Requires session management and persistent sessions
- **Load Balancing**

Server-side Cache Communication

Background

Oracle Enterprise Repository uses a server-side cache on each application server. Cached data is used only if it is available, otherwise, the database delivers data content to the cache and to the application.

When Oracle Enterprise Repository runs in a cluster, the cluster members must communicate with each other using HTTP. An edit that occurs on one cluster member invalidates the cached element on that cluster member, and communicates the edit to other cluster members. This is accomplished by a system property called `cachesyncurl`, which accepts a URL to the application as a valid value.

On start up, the system writes the `cachesyncurl` to the database and fetches a list of other server's URLs from the database. A message is sent to all discovered URLs that announces the presence of a new member of the cluster. Each server then refreshes the server list from the database. On a clean server shutdown, the value is removed from the list and a cache-refresh notification is broadcast to the server list.

When edits invalidate an element in the local cache, a message is sent to all the other servers noting which cached elements must be invalidated. Upon receipt of the message, the designated element is removed from the cache. On subsequent data request, the cache contains no data, so it first caches and then the database delivers data to the application.

Supported Application Servers

The application servers listed here are currently supported for use with clustering for Oracle Enterprise Repository:

- **Oracle WebLogic Server**
- **IBM WebSphere Application Server**

For information on the supported versions of these application servers, see the *Supported Configurations* documentation, available on Oracle Enterprise Repository index page at http://edocs.bea.com/platform/suppconfigs/configs_al10gr3/oyer10gr3/oyer10gr3.html.

Installation

Process

1. Install and configure Oracle Enterprise Repository.
2. Create the clustered environment that will host Oracle Enterprise Repository.
3. Install and deploy the Oracle Enterprise Repository application on one member of the clustered application servers.
4. Validate the deployment of the application on one member of the cluster.
5. Move the application properties to the database.
6. Shut down the cluster.
7. Install and deploy the application on all of the other cluster members.
8. Configure a `cluster.properties` file on each cluster member.
9. Start the cluster and all members.
10. Validate the cluster.

Oracle Enterprise Repository Database

For information regarding the installation of Oracle Enterprise Repository, refer to the [Oracle Enterprise Repository Installation Guide](#).

Create the Clustered Environment

For information about clustering on WebLogic or WebSphere, please refer to the application server documentation, and to organizational standards.

- **For WebLogic:**
 - Refer to *Using WebLogic Server Clusters*, available from Oracle.
- **For WebSphere Application Server:**
 - Refer to [WebSphere Software Information Center](#). Locate the documentation for the specific appserver version and navigate to:
- **All topics by feature**
 - **Servers**
 - **Clusters**
 - **Balanced workloads with clusters**

Deploy and Validate Oracle Enterprise Repository on One Cluster Member

For information on deployment and validation of Oracle Enterprise Repository on an application server, refer to the [Oracle Enterprise Repository Installation Guide](#). (For example, all of the sample names should be changed.)

Note: Before using the **Move settings to database** option, you have to enable `cmee.eventframework.clustering.enabled`, if JMS clustering is to be achieved.

Move the Application Properties to the Database

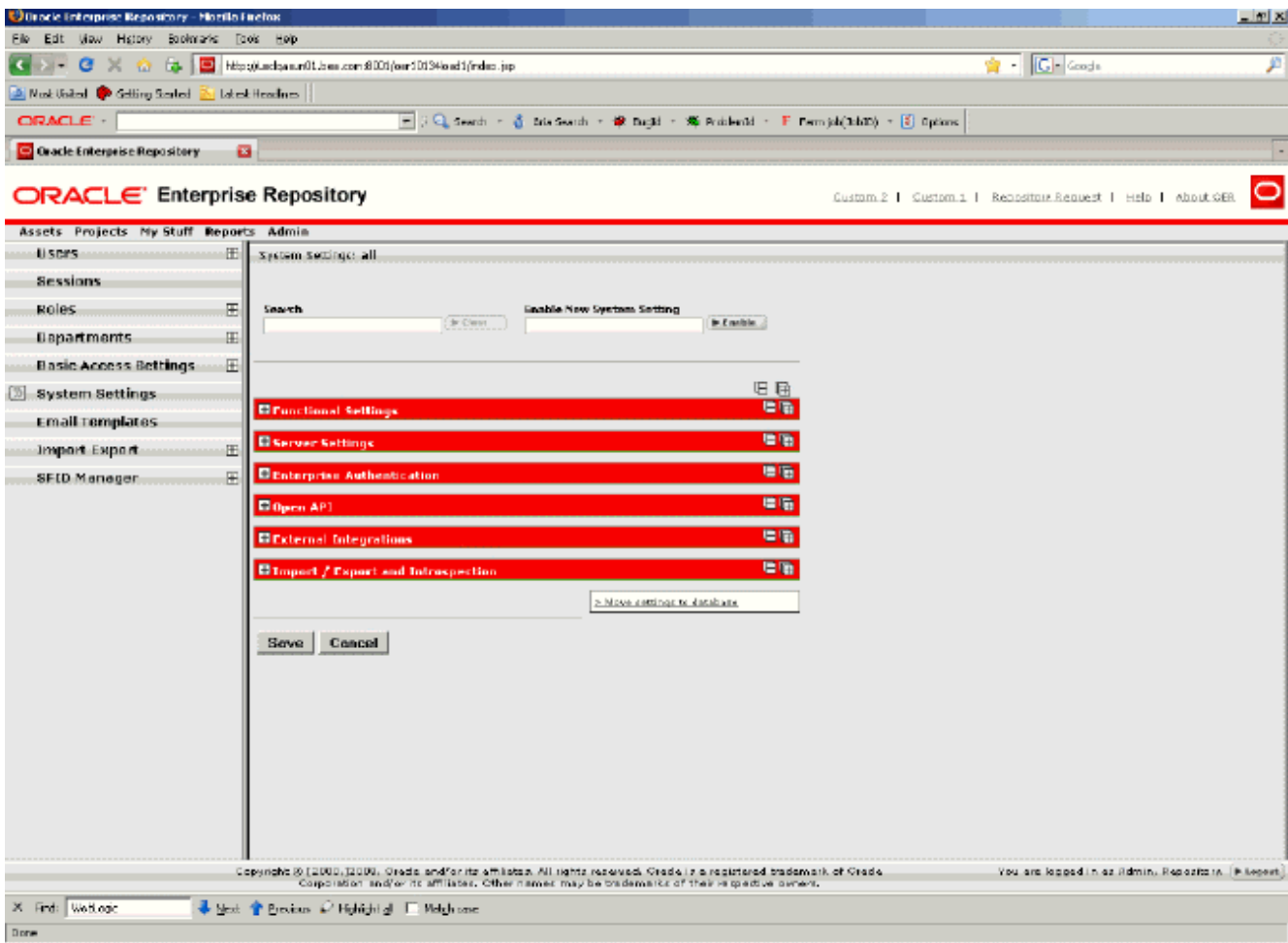
Background

Property files always take precedence when reading properties into the Oracle Enterprise Repository application. The application will look for properties and their corresponding values, first within the database, and then within the property files. Any properties read from the database are overwritten by the corresponding properties in the files. However, if there are no files, the properties within the database will only be referenced; properties that exist solely within the database will never be overwritten.

This procedure begins with deploying one application.

1. In the **Admin** screen, click **System Settings** in the left pane.

The **System Settings** section opens in the main pane.



2. Scroll to the bottom and click the **Move settings to database** button.

A confirmation message appears.

3. Remove the properties files from the classpath.

4. Restart the appserver.

5. Locate the configuration files folder (usually located within the `./WEB-INF/classes/` folder or `aler_home`) within the application server.

6. Remove the property files listed below from the configuration folder:

- o `enterprise.properties`
- o `ldap.properties`
- o `containerauth.properties`
- o `eventing.properties`
- o `jive.properties`
- o `jive.tag.properties`
- o `jive.user.properties`
- o `juddi.properties`
- o `openapiserverlog.properties`

These properties are written to the **entSettings** table within the database.

7. Modify the `cmee.properties` file. Remove all property values except those containing URL values. Update the URL references to point to the proxy server path being used to load balance access to the cluster members.

8. Now, redeploy the application on all the servers.

- **Note:** Any properties enabled after this procedure are written to the database, not to the properties files.

Configure a `cluster.properties` File on Each Cluster Member

1. Stop each cluster member.
2. On each cluster member create a file called `cluster.properties`, which resides in the same place as all other `.properties` files.
 - For exploded directory deployments this location is the `WEB-INF/classes` directory beneath the webapp.
 - For ear file deployments, this location is the `aler_home` directory.

The contents of `cluster.properties` is based on the property `cmee.server.paths.servlet` in the `cmee.properties` file. However, the hostname in the path should refer to the hostname of the cluster member, not the proxy hostname of the entire cluster.

```
cluster.properties
```

```
#cluster.properties
```

```
cachesyncurl=http://<SERVLET-PATH>/<APP_PATH>
```

Example:

```
#cluster.properties
```

```
cachesyncurl=http://server.example.com:7221/cluster01
```

Other properties that are optional

```
# alias is used as an alternate/convenient name to refer  
# to the server  
# example: server1  
# default: same value as =cachesyncurl=  
alias=EclipseServer
```

```
# registrationIntervalSeconds is the number of seconds between  
# attempts to update the server's registration record in the database  
# default: 120  
registrationIntervalSeconds=120
```

```
# registrationTimeoutSeconds is the number of seconds before a server  
# is considered to be inactive/not running  
# make sure this value is higher than the registrationIntervalSeconds  
# default: 240  
registrationTimeoutSeconds=240
```

```
# maxFailures is the number of consecutive attempts that will be made  
# to deliver a message to another server after which it will be determined  
# to be unreachable  
# default: 20  
maxFailures=20
```

```
# maxQueueLength is the number of messages that will be queued up to
# send to another server after which server will be determined to be
# unreachable
# default: 4000
maxQueueLength=5000

# email.to is the address of the email recipient for clustering status
# messages
email.to=jsmith@company.com

# email.from is the address of the sender for clustering status messages
email.from=jsmith@company.com

# email.subject is the subject line of the message for clustering status
# messages
email.subject=Oracle Enterprise Repository Clustering communication failure

# email.message is the body of the message for clustering status messages
email.message=This is an automated message from the Oracle Enterprise Repository informing you of a cluster
member communication failure.
```

Example of a cluster.properties file

```
cachesyncurl=http://server.example.com:7221/cluster01
alias=node1
registrationIntervalSeconds=120
registrationTimeoutSeconds=240
maxFailures=20
maxQueueLength=5000
email.to=jsmith@example.com
email.from=jsmith@example.com
email.subject=Oracle Enterprise Repository Clustering communication failure
email.message=This is an automated message from the Oracle Enterprise Repository informing you of a cluster
member communication failure
```

Note:

1. The time delay should not be more than 120 seconds between the application server and the database server. Network Time Protocol is recommended to keep these servers in sync. The clustering process calculates the difference of time between messaging between the nodes of the cluster.
2. Before restarting the server, you need to add `eventing.properties`, if JMS Clustering is enabled, and this should contain `cmee.eventframework.jms.producers.client.id` property with unique value on each of the cluster member. For example, `cmee.eventframework.jms.producers.client.id=OER_JmsProducer1`
3. Restart each cluster member.

Note: Once a cluster member is inactivated due to exceeding `maxFailures`, then the only way to activate is by restarting the server.

Validate the Installation

Messages are sent to the standard out log of each cluster member.

- "running in single server mode"
 - Indicates that Oracle Enterprise Repository clustering is not configured and the application is running in single server mode.
- "running in multi server mode with a sync-url of..."
 - Indicates the Oracle Enterprise Repository clustering is functioning and the application is running in clustered mode:

Variables

- **cachesyncurl**
 - The value of the `cachesyncurl` in the `cluster.properties` file, which references the same URL as the individual node's instance with the path of `/cachesync` appended. Most cluster configurations will have a proxy server load-balancing each node within the clustered server.

Example:

Node1: `cachesyncurl=http://server1.example.com:7221/cluster01`

Node2: `cachesyncurl=http://server2.example.com:7221/cluster01`

It is also possible to validate the clustering installation by viewing the clustering diagnostic page from the Oracle Enterprise Repository Diagnostics screen. Click **Cluster Info** on the Diagnostics screen to view the Cluster Diagnostic page. This page lists information about all servers registered in the cluster, as well as information about inter-server communications.

Clustering JVM Parameter for WebLogic Server

If cluster nodes are deployed via a centralized administration console, it may be necessary to apply a JVM Parameter to allow the appropriate Oracle Enterprise Repository clustering operation in the absence of the `cluster.properties` file.

This JVM parameter should be applied statically for each member of the cluster or within the managed server startup command file. This JVM parameter can be set within the `JAVA_OPTIONS` environment variable for WebLogic Application servers or within `CATALINA_OPTS` or `JAVA_OPTS` for Tomcat servers. The JVM Parameter is as follows:

```
-Dcmee.cachesyncurl=http://<member host name>:<port>/<APP_PATH>
```

Clustered JMS Servers for Advanced Registration Flows

Note: This feature is only available when using the "Advanced Registration Flows" subsystem for automating the asset registration process. Also, "JMS Clustering" applies only to the embedded ActiveMQ JMS servers in Oracle Enterprise Repository and not to external JMS servers. You need to have JDBC persistence, if you are using ActiveMQ.

In a clustered Oracle Enterprise Repository environment using the Advanced Registration Flows subsystem, each member Oracle Enterprise Repository server in the cluster will have one embedded ActiveMQ JMS server for increased reliability and scalability. For example, for a two-node cluster, there would be two Oracle Enterprise Repository servers, such as `server01` and `server02`, with each having one embedded JMS server. JMS server clustering is enabled using the Oracle Enterprise Repository "Eventing" System Settings, as described in [External Integrations: Eventing](#). Once clustering is enabled for the embedded JMS servers, you then need to specify the connection URL information for the embedded JMS servers on `server01` and `server02`.

For more information, see the **Configuring JMS Servers for Oracle Enterprise Repository** section of the [Configuring and Managing Advanced Registration Flows Guide](#).