

**Oracle® Enterprise Data Quality**

Integrated Version Control

Version 9.0.4

May 2012

**ORACLE®**

**Copyright © 2006, 2012, Oracle and/or its affiliates. All rights reserved.**

Oracle ® Enterprise Data Quality, version 9.0.4

Copyright © 2006, 2012, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. 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, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

# Table of Contents

---

<b>Table of Contents</b> .....	<b>3</b>
<b>1 Introduction</b> .....	<b>1</b>
<b>2 Integration Architecture</b> .....	<b>1</b>
<b>3 Setting up a Workspace</b> .....	<b>2</b>
<b>4 Configuring OEDQ</b> .....	<b>4</b>
4.1 Configuring a New OEDQ installation.....	4
4.2 Configuring a New OEDQ Installation without SSO.....	5
4.3 Retaining Existing Configuration Information.....	5
<b>5 Version Control Interface</b> .....	<b>6</b>
<b>6 Deployment Example</b> .....	<b>7</b>
6.1 Usage.....	8
<b>7 Troubleshooting</b> .....	<b>9</b>

# 1 Introduction

OEDQ now supports integration with Subversion, which is a version control system. This document explains how the integration works and how it is expected to be used.

**NOTE:** OEDQ currently only supports integration with Subversion 1.6. Attempting to integrate with a more recent version will cause an error.

The following sections describe the integration architecture and the steps required to configure OEDQ to use workspaces instead of its internal database:

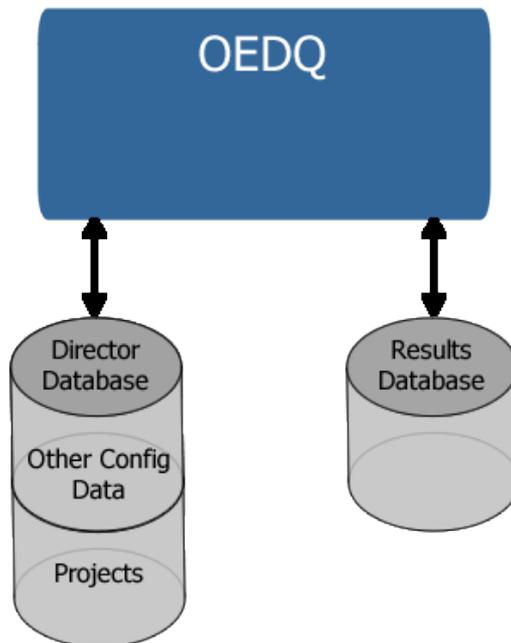
- Integration Architecture
- Setting up a Workspace in SVN
- Configuring OEDQ to use the workspace
- The Version Control interface
- An example of a deployment
- Troubleshooting

# 2 Integration Architecture

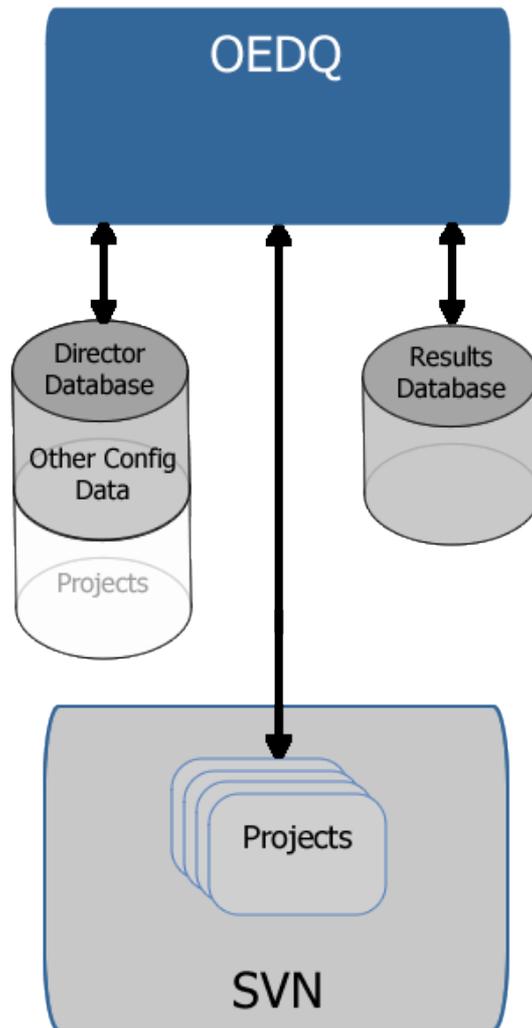
The OEDQ server can be configured to be aware of a Subversion server as a store of configuration information.

**NOTE:** In this instance, configuration information means information that is managed using the Director UI; for example, projects and system-level data.

In a standard OEDQ instance, configuration information is stored in the **director** database:



The diagram below shows an OEDQ instance integrated with Subversion:



**NOTE:** The **director** database is still required, as it holds data derived from the file-mastered configuration that has been normalized to allow querying by the applications.

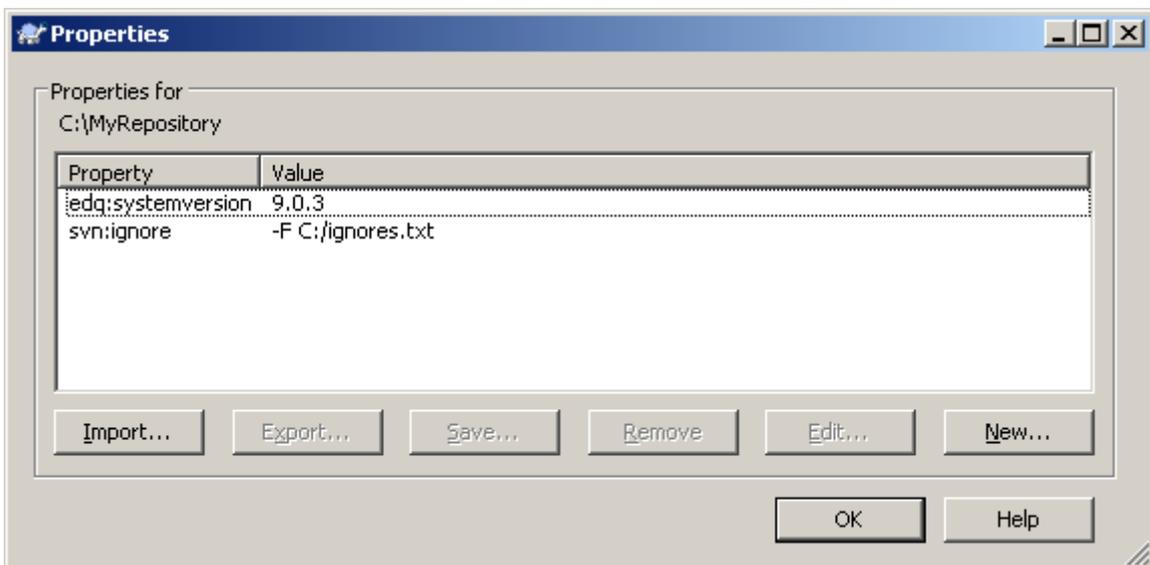
With OEDQ configuration files mastered and stored in an SVN repository, a Subversion client can be used to commit or otherwise access them. To enable users to interact with this data, a Subversion client that can be accessed from the Director UI has been integrated with the OEDQ server.

If Single Sign-On (SSO) has been enabled, OEDQ will automatically use each user's OEDQ credentials when they attempt to access the Subversion server. Therefore the Commit logs in Subversion will correctly represent the users who have committed changes to Subversion.

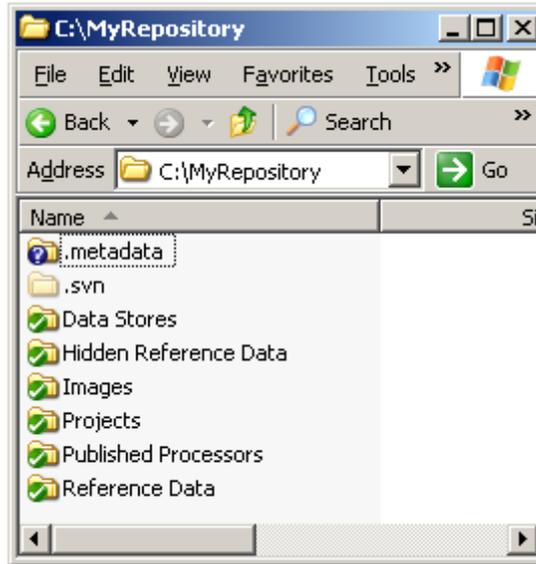
### 3 Setting up a Workspace

The first stage of configuration is to create a folder where the checked out data will be stored, i.e. a workspace:

1. Create a folder on the disk where required (e.g. C:\MyRepository) and commit it to SVN.
2. In another location, create a file called **ignores.txt** containing the line `.metadata.`
3. Inside the newly created directory, run: `svn propset svn:ignore -F <directoryname>/ignores.txt`
4. Also run (change depending on EDQ version): `svn propset edq:systemversion 9.0.3 .` (note the space between the 3 and the final .)
5. Commit the changes above into SVN. Your repository should now look like this:



6. Next create the following directories in the newly created directory:
  - Data Stores
  - Hidden Reference Data
  - Images
  - Projects
  - Published Processors
  - Reference Data



7. Commit all the directories above. The repository should then be setup correctly for EDQ.

The above steps only need to be performed once per application. From this point, all changes can be made using OEDQ.

## 4 Configuring OEDQ

Oracle recommends integrating Subversion with a fresh installation of OEDQ.

Warning: When an OEDQ instance is integrated with Subversion, ALL pre-existing and other configuration information is lost. To retain this information, it will be necessary to package and export it first. See [section 4.3 "Retaining Existing Configuration Information"](#) for further details.

Note: Oracle strongly recommends that a single workspace be assigned to each instance of Director. It is difficult to move between workspaces in a single Director instance.

### 4.1 Configuring a New OEDQ installation

1. Shut down the Application Server.
2. Set up SSO. The easiest way to do this is to add the config/security/login.properties file with the appropriate domain details.

**NOTE:** Enabling SSO for OEDQ and Subversion is the optimal solution. It allows Director users to perform SVN commits from the Director UI. Other Subversion operations (such as Add, Update, Delete and Revert) can be performed without SSO, but will only be

completed once a Commit is run. If it is not possible to enable SSO, see [section 4.2 "Configuring a New OEDQ Installation without SSO"](#).

3. Check-out the workspace from SVN. It is not necessary to checkout the whole tree; just the workspace directory itself is required.
4. Add the following line to the `config/director.properties` file, replacing the directory path with that of the absolute path to the root Workspace director:

```
sccs.workspace = C:\\dev\\svnproject
```

**NOTE:** Spaces also must be escaped.

5. Restart the Application Server, and start Director.
6. Check the top of the Main0.log for the following line:

```
INFO: 08-Feb-2012 10:05:21: SCCS workspace is C:\\MyRepository
```

7. If no errors are shown below this line, OEDQ should now be configured to use Subversion. If there are errors, see the Troubleshooting section for possible solutions.

## 4.2 Configuring a New OEDQ Installation without SSO

If SSO is not enabled, it is still possible to use the Subversion integration. Either:

- Use a single Subversion user account for all commits. This user may be specified in `director.properties` using the following syntax:

```
sccs.vcs.username = <your.username>
```

```
sccs.vcs.password = <yourpassword>
```

- Perform all commits externally to the Director UI, using Subversion controls. Note that this may then lose the link between the person that made the change in Director, and the person that commits the change.

## 4.3 Retaining Existing Configuration Information

As previously stated, it is not possible to automatically migrate configuration information during integration with Subversion.

Therefore, any pre-existing projects and other configuration items in an OEDQ installation must be packaged before integration begins, and reimported to the new installation afterwards, as outlined in the following procedure:

1. Package all configuration items in the current OEDQ instance into DXI files.
2. Install a new instance of OEDQ, with the Subversion integration enabled.
3. Import the DXI files into the new instance, and commit the files to the Subversion workspace.
4. Check that the configuration items are all valid and working correctly.

**NOTE:** All passwords for Data Stores must be re-entered after a configuration import.

5. Decommission the previous instance.

## 5 Version Control Interface

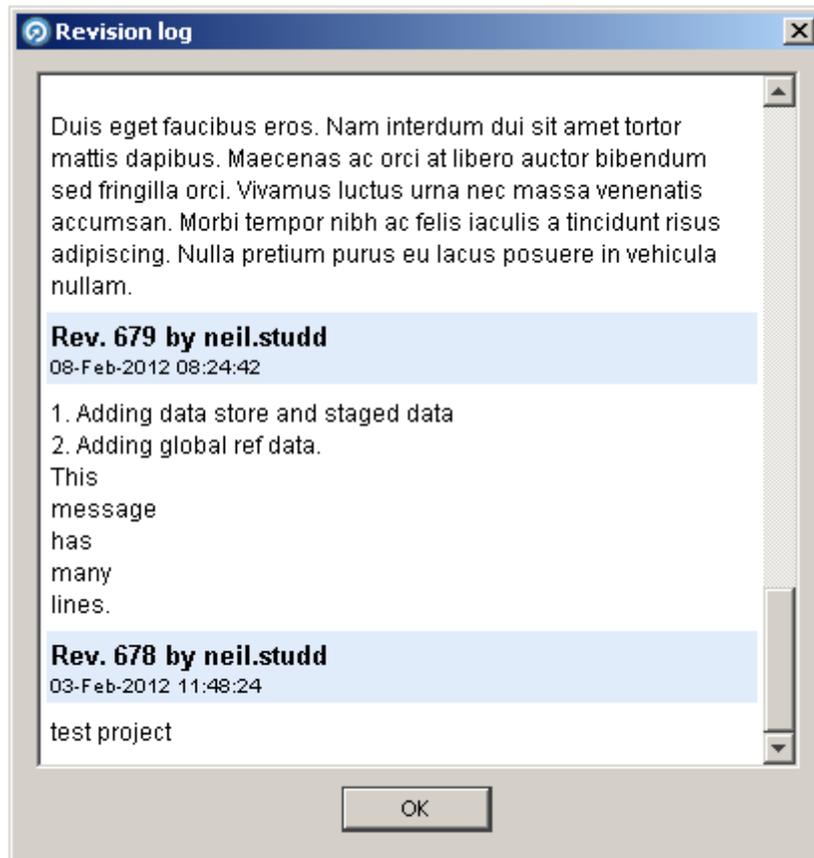
Once OEDQ is integrated with Subversion enabled, the following touch points become visible within the Director application:

- SVN status icon overlays in Project Browser - There are two icons used to indicate the three possible SVN statuses of nodes in the Project Browser:
  - No icon - The node (and its sub-nodes) are all up to date.
  -  - This node (and its sub-nodes) have modifications.
  -  - This node (and its sub-nodes) is new/currently not under Version Control.

For example, the following image shows both icons in use. The **Reference Data** node is modified (green icon) as one of its sub-nodes has changed. A new piece of Reference Data - **Business Words** - has been added, and is marked with the blue icon:



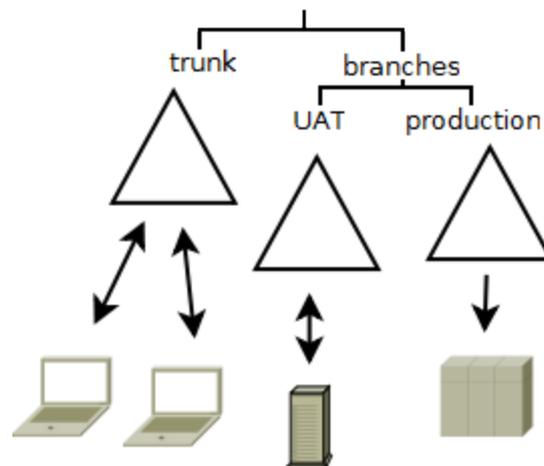
- Version Control tab - The **Properties** dialog (displayed by right-clicking on an item in the Project Browser and selecting Properties) now contains a Version Control tab that describes the state of the item: when it was last updated, its SVN revision, whether it's up-to-date, etc.
- New context menu for Version Control - The Project Browser right-click menu now contains a **Version Control** option. When selected, this displays a sub-menu with SVN options to update, commit, revert or view the log for the item. These options are recursive; e.g. if you perform **View Log** on a single process then you will see the log for this process only; but if you perform **View Log** on the **Processes** node, you'll see changes for all processes. An example of the **Revision Log** dialog is shown below:



## 6 Deployment Example

The image below show an example deployment of a single Subversion server. The server is enabled with SSO, as have the OEDQ instances that access it. The Subversion server holds three copies of configuration data. These are:

- Trunk – the traditional location that all development work is performed on. New features of the configuration are developed and saved here.
- Branches/UAT – this branch represents the copy of the configuration under UAT testing.
- Branches/production – this branch represents the production copy of the configuration.



There are four OEDQ installations using the Subversion server for storing their configuration. The machines are:

- Two development laptops where design work and maintenance of existing projects is carried out.
- A UAT server for User Acceptance Testing changes.
- A Production server for production runs.

## 6.1 Usage

In this example, laptop users develop configuration data for individual projects on their machines and then commit changes back to the Subversion repository on **trunk**. Where the developers are co-operating on developing a project they will periodically update their local installation to pick up changes from the other developers.

When the project is ready to be released to UAT, the Release Manager copies the necessary projects from **trunk** to **UAT** on the Subversion server.

As an example, the following Subversion command may be used to do this:

```
svn cp -m"Release Project X to UAT"
http://svn/repos/config/trunk/ProjectX
http://svn/repos/config/branches/UAT
```

The Test Manager then updates the UAT server's projects to load the new configuration into the OEDQ server. Over a period of time testing continues. As issues are found they are fixed in the UAT environment and committed back to the Subversion repository.

Once the UAT environment has achieved an acceptable test level it is promoted to release. This achieved in much the same way as the release from Development to UAT. The necessary projects are copied across in the Version Control repository and then the Production server is updated to use this configuration.

## 7 Troubleshooting

Error	Cause/Solution
Configuration database is not compatible with workspace	The database has been used with a different SCCS workspace. This error usually arises occurs when any operations have been performed in EDQ before version control is enabled. There are two solutions: drop/recreate the Director database, or reinstall EDQ.
Unable to locate a login configuration	This error is displayed when Single Sign-On is not enabled ( <a href="#">4.2 Configuring a New OEDQ Installation without SSO</a> ) or if the currently configured SSO user is not a trusted delegate.
Unable to open an ra_local session to URL	This may arise when trying to commit files to an invalid repository. The EDQ integration is not compatible with file-based repositories (repositories beginning with <code>file:///</code> or <code>C:\example</code> ). A fully-declared <code>http://</code> path to the repository must be made.