

Upgrading EDQ to 12.1.3 or 12.2.1

ORACLE TUTORIAL | MAY 2015



Table of Contents

Introduction	2
Table of Upgrade Paths by Release and Platform	3
Upgrade Path 1:	4
From EDQ 11g Using an Oracle / WebLogic Platform Installed Using the OUI to EDQ 12.1.3 Using an Oracle / WebLogic Platform	4
Upgrade Path 2	17
From any Release of EDQ Installed Manually (i.e. Without Using the Oracle Universal Installer) on an Oracle / WebLogic Platform to EDQ 12.1.3 Using an Oracle / WebLogic Platform	17
Upgrade Path 3:	30
From any Release of EDQ Using an Oracle / Tomcat Platform to EDQ 12.1.3 Using an Oracle / WebLogic Platform	30
Upgrade Path 4:	44
From any Release of EDQ on a PostgreSQL / Tomcat Platform to EDQ 12.1.3 Using a PostgreSQL / Tomcat Platform	44
Upgrade Path 5:	60
From any Release of EDQ on an Oracle / Tomcat Platform to EDQ 12.1.3 Using an Oracle / Tomcat Platform	60
Upgrade Path 6:	76
From EDQ 11g Using an Oracle / WebLogic Platform Installed Using the OUI to EDQ 12.2.1 Using an Oracle / WebLogic Platform	76
Upgrade Path 7:	91
From EDQ 12.1.3 Using an Oracle / WebLogic Platform Installed Using the OUI to EDQ 12.2.1 Using an Oracle / WebLogic Platform	91
Upgrade Path 8	101
From any Release of EDQ Installed Manually (i.e. Without Using the Oracle Universal Installer) on an Oracle / WebLogic Platform to EDQ 12.2.1 Using an Oracle / WebLogic Platform	101
Upgrade Path 9:	115



From any Release of EDQ Using a Postgres / Tomcat Platform to EDQ 12.2.1 Using an Oracle /
WebLogic Platform 115

Upgrade Path 10: 130

From any Release of EDQ Using an Oracle / Tomcat Platform to EDQ 12.2.1 Using an Oracle /
WebLogic Platform 130

Upgrade Path 11: 145

From any Release of EDQ on a Postgres / Tomcat Platform to EDQ 12.2.1 Using an Oracle / Tomcat
Platform 145

Upgrade Path 12: 166

From any Release of EDQ on an Oracle / Tomcat Platform to EDQ 12.2.1 Using an Oracle / Tomcat
Platform 166



Introduction

This document explains how to upgrade older releases of Oracle Enterprise Data Quality to release 12.1.3 and 12.2.1.

A Little About EDQ's Platform

In order to function, an Enterprise Data Quality server requires:

- A database to function as its repository.
- An application server.
- A Java development kit (JDK).

At release 12.1.3 the certified databases are the Oracle database and Postgresql.

At release 12.2.1 only the Oracle database is certified for production use. Postgresql is certified only for development, test and evaluation use.

At release 12.1.3 and at release 12.2.1, both Oracle WebLogic and Tomcat are certified as application servers.

Please see the Fusion Middleware Certification matrixes for details of the versions of these databases and application servers that are certified, and please ensure that you only use certified versions.

What Platform Does Oracle Recommend?

Customers are urged to upgrade to a platform consisting of the Oracle Database and Oracle WebLogic wherever possible. This is because future upgrades are likely to be much more straightforward with this combination of database repository and application server.

Upgrading Production Instance of EDQ 9.0 and Below That Were Installed Using the the Windows Installer

For EDQ 9.0 and for earlier releases, a Windows Installer was provided that installed Enterprise Data Quality on Windows operating systems using the Tomcat application server and the PostgreSQL database as its platform. These installations were supported. You should follow the appropriate upgrade path in this document to upgrade all production instances of EDQ 9.0 and below that were installed using the Windows installer.

Quick-Start Installations of EDQ 11 and 12

For EDQ 11 and later releases, Oracle renamed the Windows installer the 'Quick-Start Installer (Windows).' For EDQ 11 and for later releases, the Quick-Start Installer must not be used to install production environments. It may be used to install EDQ for development, test and evaluation purposes only. Instances of EDQ 11 and later that are installed with **the Quick-Start installer are not supported, and are not patchable.**

Upgrading Development, Test and Evaluation Instances of EDQ That Were Installed Using Either the Windows Installer or the Quick-Start Installer

The primary intent of this document is to guide you through upgrades to supported installations of Enterprise Data Quality. Whilst you could use the upgrade paths in this document to upgrade development, test and evaluation installations of EDQ that were installed using either the Windows Installer or the Quick-Start Installer, the easiest way to upgrade such instances is simply to run the Quick-Start installer for the release you want to upgrade to. This should automatically recognize and upgrade your existing instance.

Table of Upgrade Paths by Release and Platform

	To	Release 12.1.3 on Oracle / WebLogic	Release 12.1.3 on Postgres / Tomcat	Release 12.1.3 on Oracle / Tomcat	Release 12.2.1 on Oracle / WebLogic	Release 12.2.1 on Postgres / Tomcat	Release 12.2.1 on Oracle / Tomcat
From							
Release 11 on Oracle / WebLogic Installed with OUI		Recommended: See upgrade path 1	Inadvisable – no upgrade path provided	Inadvisable – no upgrade path provided	Recommended: See upgrade path 6	Not possible – no upgrade path provided	Inadvisable – no upgrade path provided
Release 12.1.3 on Oracle / WebLogic installed with OUI		Not Applicable	Inadvisable – no upgrade path provided	Inadvisable – no upgrade path provided	Recommended: See upgrade path 7	Not possible – no upgrade path provided	Inadvisable – no upgrade path provided
Any Release installed on Oracle / WebLogic manually (i.e without the OUI)		Recommended: See upgrade path 2	Inadvisable – no upgrade path provided	Inadvisable – no upgrade path provided	Recommended: See upgrade path 8	Not possible – no upgrade path provided	Inadvisable – no upgrade path provided
Any release installed on Postgres / Tomcat		Not possible – no upgrade path provided	Possible: See upgrade path 4	Not possible – no upgrade path provided	Recommended: See upgrade path 9	Not possible – no upgrade path provided	Possible: See upgrade path 11
Any release installed on Oracle / Tomcat		Recommended: See upgrade path 3	Inadvisable – no upgrade path provided	Possible: See upgrade path 5	Recommended: See upgrade path 10	Not possible – no upgrade path provided	Possible: See upgrade path 12

Upgrade Path 1:

From EDQ 11g Using an Oracle / WebLogic Platform Installed Using the OUI to EDQ 12.1.3 Using an Oracle / WebLogic Platform

Overview

Upgrading from an installation of EDQ 11g that was installed using the Oracle Universal Installer (OUI) and that uses the Oracle database and the WebLogic application server as its platform to EDQ 12.1.3 using the Oracle Database and Oracle WebLogic as its platform is essentially a two-part process:

- Part 1 – Perform a Partial Installation of EDQ 12.1.3
- Part 2 – Upgrade Your Old EDQ Environment.

Part 1: Perform a Partial Installation of EDQ 12.1.3

1. Ensure that your environment satisfies all EDQ 12.1.3 System Requirements, as outlined in section 1.4 of the Installing and Configuring Oracle Enterprise Data Quality guide, and upgrade your environment if necessary.
2. Download EDQ 12.1.3, as described in section 1.5 of the Installing and Configuring Oracle Enterprise Data Quality guide.
3. Download the Fusion Middleware Infrastructure release 12.1.3.
4. If your environment's current Java Development Kit does not support EDQ 12.1.3, install a Java Development Kit to support EDQ 12.1.3, as described in section 2.2 of the Installing and Configuring Oracle Enterprise Data Quality guide.
5. If your environment's current Oracle Fusion Middleware Infrastructure does not support EDQ 12.1.3, install Oracle Fusion Middleware Infrastructure to support EDQ 12.1.3.

Note that the Oracle Fusion Middleware Infrastructure that you used to support your old release of EDQ will not support your new release of EDQ. Therefore, unless, you have already installed the new release of the Oracle Fusion Middleware Infrastructure 12.1.3 for some other purpose, you will have to install it now. Note also that the new release of the Fusion Middleware Infrastructure will have to be installed in a different Fusion Middleware Home to your old release.

6. If your environment's current Oracle Database does not support EDQ 12.1.3, upgrade the Oracle Database to a release that will support EDQ 12.1.3.
7. Install Enterprise Data Quality 12.1.3 assets, as described in section 3 (Installing Enterprise Data Quality) of the Installing and Configuring Oracle Enterprise Data Quality guide, ensuring that you install EDQ in your new Fusion Middleware Infrastructure's Fusion Middleware Home. For the avoidance of doubt, this step entails installing the Oracle Fusion Middleware EDQ distribution. This is usually downloaded as a zip file that contains a single file with a name similar to **fmw_12.1.3.x.x_edq.jar**. Running this jar file invokes the Oracle Universal Installer.

This step places elements of Enterprise Data Quality in your environment's new Fusion Middleware Home location. Some of these elements will be used later in the installation process. Note, however, that you will not have a working EDQ environment at this stage.

Part 2: Upgrade your Old EDQ Environment

1. Back up your old EDQ environment.
 - a. Before you begin, Oracle recommends that you make a complete backup of your old EDQ environment, including your EDQ domain. For instructions, see "Backup and Recovery Strategies for Upgrade" in the Planning an Upgrade of Oracle Fusion Middleware documentation.
2. Shutdown your old EDQ environment's Managed WebLogic Server.
3. Ensure that your old EDQ environment's WebLogic Server Administration Server remains running.
4. If your old EDQ environment used a File-Based Keystore, migrate it to an OPSS Security Store.

At release 12c, all Oracle Fusion Middleware products store their security credentials in an Oracle Platform Security Services (OPSS) schema within an Oracle database. The Oracle term for this is security store. If you were using a file-based keystore in the configuration directories, you must now use an OPSS DB-based security store.

To convert to an OPSS DB-based security store, you create the OPSS, Audit Services, and Audit Services Viewer schemas. The Audit Services and Audit Services Viewer schemas are required by the Oracle Platform Security Services schema. After creating these schemas, you then create a WebLogic Server data source for the OPSS schema, and then re-associate the EDQ keystore to the OPSS schema.

- a. Create the OPSS Schema
 - i. Download the 11g Repository Creation Utility (RCU) from Oracle. For instructions, see http://docs.oracle.com/cd/E23104_01/download_readme_ps6/download_readme_ps6.htm#BABHDBJC
 - ii. Extract the contents to a directory of your choice; this directory will be referred to as the RCU_HOME directory. On Windows operating systems, make sure that you do not unzip the RCU file to a directory name that contains spaces.
 - iii. Run the command shell or console of the operating system.
 - iv. Start RCU from the bin directory inside the RCU_HOME directory.

Note:

If you are running RCU using a non-English database, set the following environment variables:

```
setenv LANG en_US.UTF8
```

```
setenv LC_ALL $LANG
```

```
setenv NLS_LANG american_america
```

On Linux:

```
./rcu
```

On Windows:

```
rcu.bat
```

- v. Start RCU from the bin directory inside the RCU_HOME directory.
- vi. Click **Next >** to pass through the Welcome screen.
- vii. On the Create Repository screen, select **Create**, then click **Next >**.
- viii. On the Database Connection Details screen, enter the connection details for your old environment's EDQ repository database and the login credentials of an EDQ repository database user who has DAB or SYSDBA privileges. Click **Next >**. The RCU will check prerequisites.
- ix. On the Select Components screen, ensure that you select the same schema prefix that is used for your old environment's EDQ Repository (for example, if the old environment's schemas started with **DEV**, then ensure you select **DEV**). This will usually entail checking the **Select existing prefix** option. Under Oracle AS Repository Components, expand **AS Common Schemas** and then select **Audit Services** and **Oracle Platform Security Services**. Click **Next >**.
- x. On the Schema Passwords screen, supply passwords for the schema owners. Click **Next >**.
- xi. On the Map Tablespaces screen, the default tablespace names are displayed. Either accept these, or select alternative tablespaces for the components. (Optionally, you can click **Manage Tablespaces** at this point to change the tablespaces sizes.) Click **Next >**.
- xii. On the Summary screen, click **Create** to complete the creation of the schema.

b. Create a JDBC Data Source for the OPSS Schema.

Use the WebLogic Server Administration Console to create a generic JDBC data source for the OPSS schema. For assistance or more information on any screen, click **Help**.

- i. In the Domain Structure tree, expand **Services**, then select **Data Sources**.
- ii. On the Summary of Data Sources screen, click **New** and select **Generic Data Source**.
- iii. On the JDBC Data Sources Properties screen, enter the data source name and the JNDI name of the OPSS schema that you created with the 11g RCU in Task 1. The OPSS DataSource name is **opss-data-source** and the JNDI name is similar to this example: **jdbc/OpssDataSource**. Select the database type as **Oracle**, and then click **Next**.
- iv. On the Create a New JDBC Data Source screen, in the Database Driver pull-down, select **Oracle's Driver (Thin) for Instance connections; Versions:9.0.1 and later** (this is a non-XA JDBC driver) and then click **Next**.

Note: You must install JDBC drivers before you can use them to create database connections. Some JDBC drivers are installed with WebLogic Server, but many are not installed.

- v. On the Transaction Options screen, make sure that **Supports Global Transactions** is **deselected** and then click **Next**.
- vi. On the Connection Properties screen, enter the connection details and credentials to connect to the OPSS schema that you created with the 11g RCU in the previous step.
- vii. On the Test Database Connection screen, review the connection parameters and click **Test Configuration**. Results from the connection test are displayed at the top of the page. If the test is unsuccessful, correct any configuration errors and retry the test. If the JDBC driver you selected is not installed on the WebLogic Server Administration Server, click **Next** to skip this step. Install the driver after finishing this procedure.
- viii. On the Select Targets page, select the WebLogic Server(s) or cluster(s) on which you want to deploy the data source. Select both the EDQ Managed WebLogic Server and the WebLogic Server Administration Server as targets. If you are using WebLogic clustering, deploy the data source to the cluster.
- ix. Click **Finish** to save the JDBC data source configuration and deploy the data source to the targets that you selected.
- x. If your WebLogic Server is working in production mode, click **Activate Changes** in the Change Center of the Administration Console. Not all changes take effect immediately, but all will take effect when you restart the server at the end of the EDQ upgrade process.
- xi. Restart the WebLogic Server Administration Server. (If the console tells you that no restart is necessary, ignore this message and restart anyway – the next step will not work unless you restart the Administration Server).

c. Reassociate the File-based Keystore to a DB-based Security Store

Use the WebLogic Server Scripting Tool (WLST) to reassociate the file-based keystore to the OPSS data source that you created with the WebLogic Server Administration Console.

- i. From the 11g \$EDQ_HOME/oracle_common/common/bin directory, issue the following command to run the WebLogic Server Scripting Tool (WLST):

UNIX/Linux:

```
./wlst.sh
```

Windows:

```
wlst.cmd
```

- ii. In WLST, connect to your WebLogic instance.

```
connect('username', 'password', 'host:port')
```

Where:

`username` is the name of the Oracle WebLogic Server admin user.

`password` is the Oracle WebLogic Server admin user's password.

`host` is the name of the machine that the WebLogic server is running on.

`port` is the connection port for the WebLogic Server Administration Server.

- iii. Reassociate the OPSS schema from the file-based format to the database format using the reassociate script.

```
reassociateSecurityStore(domain="upgradeDomain",
servertime="DB_ORACLE", jpsroot="cn=jpsroot",
datasourcename="name_of_datasource")
```

Where:

`upgradeDomain` is the name of the WebLogic domain used for your old instance of Enterprise Data Quality.

`name_of_datasource` is the name of the new OPSS JNDI data source that you created in the WebLogic Server Administration Console in the previous step.

For more information about the reassociate script, see

http://docs.oracle.com/cd/E29542_01/core.1111/e10043/cfgauthr.htm#JISEC2675

- iv. Once the reassociation has completed, restart the WebLogic Server Administration Server.

Do not restart the EDQ managed server(s) yet. Complete the rest of the upgrade steps. The changes you made are stored on the server and will propagate to the rest of the servers when you restart the servers after upgrade.

5. Create the Required 12c Schemas.

Oracle Fusion Middleware Release 12c introduces new schemas and tables that must be created with the Release 12.1.3 version of the Oracle Repository Creation Utility (RCU). This step of the upgrade must be performed before upgrading your old environment's EDQ Domain.

- a. To create the required 12.1.3 schemas, see Section 4.2, "Creating an EDQ Database Repository" of the Installing and Configuring Oracle Enterprise Data Quality guide for release 12.1.3. Follow those instructions, but with these four additions:
 - i. On the Database Connection Details screen, ensure that you supply the connection details for your old EDQ environment's database repository. This is because you are going to add new EDQ 12.1.3 schemas to your old repository.
 - ii. On the Select Components screen, ensure that you select the same schema prefix that is used for your old environment's EDQ Repository (for example, if the old

environment's schemas started with **DEV**, then ensure you select **DEV**). This will usually entail checking the **Select existing prefix** option.

- iii. On the Select Components screen, the **AS Common** Schemas option should be selected by default. Do not make any changes.
- iv. If, when you click **Next**, you see an error message similar to "RCU-6019: Invalid format for schema owner STB. Prefixable components should have their schema owner in the following format <prefix>_<Any String> where the prefix is DEV", deselect and then reselect the **Common Infrastructure Services** Component, and click **Next >** again.
- v. The effect of this should be that the RCU will create exactly one new schema:
 - Common Infrastructure Services.

You should be able to confirm that this new schema (and its associated tablespaces) are going to be created when you get to the Map Tablespaces screen, and also on the subsequent Summary screen. Before you click Create on the Summary screen, make absolutely sure that this schema will be created, and also ensure that it has the same prefix as your existing EDQ Repository schemas.

6. Follow the instructions below to upgrade the schemas in the EDQ repository database.

a. Start the Upgrade Assistant:

- i. Go to the following directory, where 12.1.3_FMW_HOME is the version 12.1.3 Oracle home directory.

On UNIX operating systems:

```
12.1.3_FMW_HOME/oracle_common/upgrade/bin
```

On Windows operating systems:

```
12.1.3_FMW_HOME\oracle_common\upgrade\bin
```

- ii. Run the following program:

On Linux or UNIX operating systems:

```
./ua
```

On Windows operating systems:

```
ua.bat
```

- b. At the Welcome screen, Review Important Reminders before Proceeding. The Welcome screen contains reminders to consider before proceeding with your upgrade. Make sure you read these and verify that you are ready to proceed. For assistance or more information on any screen, click Help. Otherwise, click **Next >**.

For more information about the following screens, see "Upgrading Schemas with the Upgrade Assistant" in *Upgrading with the Upgrade Assistant*.

- c. Select the type of upgrade you wish to perform.
 - i. Select **Schemas**.
- d. The Available Components screen will display a list of the components (or in other words, schemas) from your old EDQ environment's domain. The following components should always be listed:
 - Oracle Enterprise Data Quality Results Schema
 - Oracle Audit Services
 - Oracle Platform Security Services
 - Oracle Enterprise Data Quality Configuration Schema

Depending upon how your EDQ 11g environment was configured, it is possible that other components may also be listed.

Select all of the following components if they appear:

- **Common Infrastructure Services**
- **Oracle Audit Services.**
- **Oracle Enterprise Data Quality** (this should automatically select both Oracle Enterprise Data Quality Configuration Schema and Oracle Enterprise Data Quality Results Schema)
- **Oracle Platform Security Services**

- i. Click **Next >**.
- e. At the Domain Directory screen, enter your old EDQ environment's domain directory (for example, this could be similar to C:\Oracle11\Middleware\user_projects\domains\edq_domain). Click **Next >**.
- f. In the Prerequisites screen, verify that you have met the prerequisites, and, provided that you have, check the relevant boxes (note that The Upgrade Assistant will not verify that the prerequisites have been met.). Then click **Next >**.
- g. Specify Database and Schema Credentials
 - i. Screens for each of the components you selected, starting with The EDQRESULTS schema, are displayed in succession.
 - ii. On the first screen, specify connection details for your old environment's EDQ Repository (note that you may have to add **as sysdba** to the DBA User Name), and then click **Connect**. Once you have clicked Connect, the Schema User Name and Schema Password fields should be populated automatically for the EDQ Results schema (if they are not, enter this information manually).
 - iii. The remaining screens are automatically populated with the database connection and schema credentials that you supplied on the EDQ Results Schema screen. If these

- entries are not correct for any schema, change the entries and ensure that a database connection is made.
- h. Complete the upgrade validation.
 - i. On the Examine screen, the Upgrade Assistant performs a series of validations on the components. Validations for the Oracle Enterprise Data Quality Results component, the Oracle Enterprise Data Quality Configuration component and Oracle Platform Security Services component should succeed, and the Upgrade Assistant should report that an upgrade is not necessary for either of the Oracle Audit Services component or the Common Infrastructure Services Component. Click **Next >**.
 - i. Initiate the Upgrade.
 - i. Click **Upgrade** on the Upgrade Summary screen to initiate the upgrade. The Upgrade Progress screen shows information about the progress of the upgrade, and the Upgrade Success screen summarizes the upgrade.
- 7. Reconfigure the EDQ Domain.

In this step you will run the Oracle Fusion Middleware reconfiguration wizard to complete the upgrade of your WebLogic Server domain environment.

- a. Check that the Staging Mode is set to Nostage.
 - i. Log in to the WebLogic Server Administration Server console from any web browser.
 - ii. In the left pane, expand **Environment** and then click **Servers**. The Summary of Servers screen will be displayed.
 - iii. In the Summary of Servers screen, click your EDQ Managed WebLogic Server (for example: **edq_server1**).
 - iv. Select the **Configuration** tab, then the **Deployment** tab.
 - v. From the **Staging Mode** menu, ensure that **nostage** is selected, and, if it is not, select it. Do not change any other server settings.
 - vi. Click **Save**, and, if working in production mode, activate your settings.
- b. Start Node Manager if it is not already running

*Note that you only need to complete this step if the Node Manager is **not** already running.*

This task ensures that Node Manager has been started at least once since it was installed. Starting Node Manager creates a set of script files that must be moved from the old domain to the release 12.1.3 domain. Performing this task eliminates the need for extra steps when upgrading your domain.

- i. Using a command line tool (for example, the Windows Command prompt or a Linux or Unix terminal), navigate to the **bin** folder in your old EDQ environment's WebLogic base domain (this will be at a location similar to **FMW_HOME/user_projects/domains/edq_domain/bin**).
- ii. Run the following command to start Node Manager. For **hostname**, specify the name of the system where Node Manager is installed, and for **port**, specify the port number on which it runs.

Windows:

```
startNodeManager.cmd hostname port
```

UNIX:

```
./startNodeManager.sh hostname port
```

c. Stop Node Manager

- i. If the Node Manager was not already running, and you started it using the command above, then stop it by closing the command shell from which it is running.
- ii. If Node Manager was already running (for example, because it was already running as a service in your environment), then stop it using the appropriate means (for example, stop the Node Manager service).

d. Start the Reconfiguration Wizard

In this step you will launch the wizard that will upgrade the EDQ domain.

- i. Stop the WebLogic Server Administration Server and all managed servers.
- ii. Go to the following directory, where **12.1.3_FMW_HOME** is the release 12.1.3 Fusion Middleware home directory.

On Windows operating systems:

```
12.1.3_FMW_HOME\oracle_common\common\bin
```

On UNIX operating systems:

```
12.1.3_FMW_HOME/oracle_common/common/bin
```

- iii. Start the domain reconfiguration wizard.

On UNIX operating systems:

```
./reconfig.sh -log=log_file
```

On Windows operating systems:

```
reconfig.cmd -log=log_file
```

Specify the full path and file name in place of `log_file`. Creating a log file can be helpful if you need to troubleshoot the reconfiguration process.

Note that if the following error message is displayed, it indicates that the default cache directory is not valid:


```
*sys-package-mgr*: can't create package cache dir
```

You can change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`. For example:

```
CONFIG_JVM_ARGS=-Dpython.cachedir=valid_directory
```

e. Specify the domain.

- i. On the Select Domain screen, specify the full path to the location of your old EDQ domain (it is likely to be similar to `OLD_FMW_HOME/user_projects/domains/edq_domain` or



`OLD_FMW_HOME\user_projects\domains\edq_domain`). You can also click **Browse** and use the file manager window to help you select the domain location. (Be careful – you may find that the 12.1.3 Home is selected by default – remember to navigate to your old FMW home instead).

- f. View the Reconfiguration Setup Progress.
 - i. The Reconfiguration Setup Progress displays the reconfiguration progress and verifies whether the base domain that you selected can be reconfigured to the 12.1.3 domain. The message “Core WLS Infrastructure Reconfigured Successfully!” indicates that the domain can be reconfigured to the 12.1.3 domain, and you can click **Next** to go to the next step. If this message is not returned, the domain cannot be reconfigured to the 12.1.3 domain. If this happens, check to see if the EDQ version is earlier than version 11g; if so, then select the appropriate upgrade path for version 12.1.3, and start the entire upgrade process again.
- g. Select the Domain Mode and JDK.
 - i. It is not possible to change the Domain Mode from development to production or vice versa.
 - ii. On the Domain Mode and JDK screen, specify the location of the Java Development Kit (JDK) to use in the new EDQ domain. This must be a JDK that conforms to the system requirements of your new EDQ 12.1.3 environment.
- h. Check the JDBC Data Sources.
 - i. On the JDBC Data Sources screen, your old EDQ domain's Config and Results datasources should be listed in the lower half of the screen. If you need to make changes, select the check box next to the data source name and then make the changes.
- i. Test the JDBC Data Sources
 - i. On the JDBC Data Sources Test screen, test the data source connections that were detected. Select both schemas and then click **Test Selected Connections**.
- j. Select the Database Configuration Type.
 - i. In the Database Configuration Type screen, the **RCU Data** option should be selected by default. The connection details for your EDQ repository database should be partially populated, and the schema owner for the Common Infrastructure schema (STB) should be displayed. If this information is not all automatically populated, or is not correct, make any necessary entries or amendments. (In particular, make sure that the prefix of the schema owner is correct).
 - ii. Click **Get RCU Configuration**. The Reconfiguration Wizard should report that it can connect to the Database Server and that it can retrieve schema data and bind local schema components with the retrieved data. If these steps fail, amend the connection data and click **Get RCU Configuration** again. Otherwise, click **Next >**.
- k. Check the JDBC Component Schema.

- i. On the JDBC Component Schema screen, the schemas associated with your old EDQ domain are listed in the lower half of the screen. If you need to make changes, select the check box next to the data source name and then make the changes. Note that the connection details for some of the schemas may not be correct. If they are not, amend them so that they are correct.
- l. Test the JDBC Component Schema
 - i. On the JDBC Component Schema Test screen, test the data source connections that were detected. Select the schemas that you want to test and then click Test Selected Connections.

*To test a connection, the database to which you are connecting must be running. Note that the test for the OPSS Audit Viewer component schema may fail (all other tests should succeed). If the test for the OPSS Audit Viewer component schema fails, click **Next >** anyway, and click **OK** in the subsequent dialog box to proceed without verification.*

- m. Node Manager
 - i. Leave the default options for the Node Manager Type and Node Manager Configuration selected, and enter the Node Manager's Credentials.
- n. Select Optional Advanced Configuration Options

On the Advanced Configuration Screen, you can select additional domain options.

- i. Select **Managed Servers, Clusters and Coherence** and click **Next >**.
- o. Review Managed Servers.
 - i. On the Managed Servers screen, make no changes, and just click **Next >**.
- p. Review Clusters.
 - i. On the Clusters screen, make no changes (assuming that you do not want to change your cluster configuration), and just click **Next >**. (Note that this screen will be empty unless WebLogic clusters were configured in your old environment – leave it empty unless you want to use WebLogic clusters to configure EDQ for high availability).
- q. Review Coherence Clusters.
 - i. On the Coherence Clusters screen, make no changes (assuming that you do not want to change your Coherence cluster configuration), and just click **Next >**.
- r. Review Machines.
 - i. On the Machines screen, make no changes (assuming that you do not want to change your machine), and just click **Next >**.
- s. Assign Servers to Machines.
 - i. Assign the EDQ Managed WebLogic Server (probably called **edq_server1**) to your machine. Ensure that the AdminServer is assigned to the Machine too. Click **Next>**.

- t. Initiate the Domain Reconfiguration
 - i. On the Configuration Summary screen, review the configuration and then click **Reconfig** to start the reconfiguration process, or click **Back** to make changes.
 - u. Finish the Reconfiguration
 - i. Wait for the message that states, "Domain Reconfiguration Applied Successfully" and then click **Next**.
 - ii. A check mark and the message "Oracle WebLogic Server Reconfiguration Succeeded" indicate that the reconfiguration was successful. Click **Next >**, and then click **Finish** to dismiss the reconfiguration wizard.
8. Apply the Upgrade Changes to the Base Domain

These steps apply the changes that you made during the upgrade of the EDQ domain to the base domain.

- a. Start the WebLogic Node Manager for the upgraded EDQ domain.

*Take care that you start the right Node Manager, especially if your old (pre-upgrade) EDQ environment's Node Manager was started via a service: the service will almost certainly still be pointing at your old (not upgraded) Node Manager configuration. If you simply restart your service it will start your old environment's Node Manager, which will not work against your upgraded EDQ domain. A safe option is to start the upgraded Node Manager by running the startNodeManager script (startNodeManager.sh for Unix and Linux, or startNodeManager.cmd for Windows) that you will find in your EDQ domain's **bin** directory.*

- b. Start the WebLogic Server Administration Server for your upgraded EDQ domain. If it is running, stop and then restart it.

(UNIX) `DOMAIN_HOME/bin/startWebLogic.sh`

(Windows) `DOMAIN_HOME\bin\startWebLogic.cmd`

- c. Log in to the WebLogic Server Administration Server console from any web browser.
- d. In the Domain Structure list, click **Deployments**.
- e. Under Deployments, select **edq** and then click **Update**.
- f. Click **Next**.
- g. Click **Finish**.
- h. To complete the upgrade, start the EDQ managed server.

The startManagedWebLogic.sh script is located in your EDQ WebLogic domain's bin directory. Running this script is one method of starting the EDQ WebLogic Managed Server. Note that the startManagedWebLogic.sh script may be overwritten when the EDQ WebLogic Domain is reconfigured. Therefore, if you customized the script, any customizations may have been lost during the upgrade process. This is a particularly important point to note if you embedded EDQ's Server Parameters (in other words, its 'startup arguments' or 'Java options') within this script, as, without these, the EDQ Managed Server may not function correctly. See section 6 of the Installing and Configuring Oracle Enterprise Data Quality guide, "Setting Server Parameters to Support Enterprise Data Quality", for more information about EDQ's server parameters.

9. Open a Web Browser, and navigate to your upgraded EDQ environment's Launchpad (unless you changed any of the default options when installing EDQ, its address will be <http://your-server-name:8001/edq>).
10. From the launchpad, open the Director user interface and log in.
11. Verify that all your projects have been successfully migrated.
12. If any of your processes use Match Review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

Upgrade Path 2

From any Release of EDQ Installed Manually (i.e. Without Using the Oracle Universal Installer) on an Oracle / WebLogic Platform to EDQ 12.1.3 Using an Oracle / WebLogic Platform

Overview

Upgrading from a release of EDQ that was installed manually (that is, without the Oracle Universal Installer) and which uses the Oracle database and the WebLogic application server as its platform to EDQ 12.1.3 using the Oracle Database and Oracle WebLogic as its platform is essentially a two-part process:

- Part 1 – Install EDQ 12.1.3
- Part 2 – Migrate configuration data from your old EDQ environment to your new EDQ 12.1.3 environment. This entails using scripts provided with the EDQ 12.1.3 release, and some standard Oracle tools, to reshape and move data. It also entails manually copying some files.

Part 1: Install EDQ 12.1.3

1. Install Enterprise Data Quality 12.1.3 using the Oracle WebLogic Server as its Java Application Server and the Oracle Database as its structured query language (SQL) relational database management system (RDBMS); refer to <http://add-a-link-to-the-12.1.3-installation-doc> for more information about how to do this. During part 2, you will need to know the following information about your EDQ 12.1.3 environment:
 - a. The repository database's Host Name, Port Number and Database ID.
 - b. The schema owner and password for the config and results schemas.
 - c. The environment's domain location.

Part 2: Migrate the data from the old DIRECTOR schema to the new EDQCONFIG schema

1. Back up elements of your old EDQ environment.

During part 2, you will migrate configuration data from your old EDQ repository to your new EDQ repository, and you will also copy configuration files from your old environment to your new environment. You backup elements of your old environment (that is, the configuration data and the configuration files) so that you can easily restart this process should anything go seriously wrong.

- a. Before you back up your old EDQ environment, shut down its WebLogic Managed server.
- b. Take a copy of your old EDQ environment's configuration folder. By convention this is usually named either **config**, **oedq_local_home** or **oedq.local.home**. (note that EDQ 9.0 and below had only a single configuration folder; for release 11 and above, it is the local configuration folder that you need to back up).
- c. Backup your old EDQ environment's database repository using the appropriate Oracle tools.
- d. **Do not** restart your old EDQ environment's application server.

In case of a serious error during the upgrade you are about to perform (for example, if the migration script should fail to complete, and you are left with a schema that is stuck in between your old version of EDQ and your new version of EDQ), you can now fairly easily start the process again from scratch. How?

- Stop your old EDQ environment's WebLogic Managed server if it is running.
- Delete the original version of the folder you have backed up.
- Copy the backup into its original location.
- Restore the database repository from the backup.

...And then you can start the migration process again from scratch.

2. Run and examine the premigration report. This will reveal any problems with the Case Management schema (which is, strictly speaking, not a schema at all, but just a set of tables within EDQ's config schema). If any problems are revealed, you should fix them before proceeding to step 3.
 - a. Open a command line tool:
 - i. In a Windows environment, open a Windows Command Prompt, running it as administrator (for example, right-click Command Prompt in the All Programs > Accessories group, and select **Run as administrator**).
 - ii. In a Unix or Linux environment, open a terminal window.
 - b. Change directory to your Java Development Kit's bin folder. You can do this by issuing the command `cd path_to_the_jdk_bin_folder` (for example, in a Windows environment you might enter a command similar to: `cd C:\Program Files\Java\jdk1.8.0_40\bin`).
 - c. Run the pre-migration report by issuing the following command:

```
java -jar path_to_migration.jar premigrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with your new EDQ environment. This should be in `~ORACLE_HOME\edq\oracle.edq`
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the `SID` for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to run the premigration report in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar
preigrate oracle:mydatabase@localhost:1521/director/director
```

- d. If the Premigration report reveals any problems, fix them, and run the report again before proceeding further. (If the premigration report returns no output, by the way, that's a good thing: it means no problems were found.)

Examples of Problems Revealed by the Premigration Report, With Suggested Fixes

Example One: Let's say that you see the following error in the premigration report:

```
Parameter x in Source y for Case Type z using Selection Function aa "
" is using an attribute under input 'bb that is not mapped to case management.
This input will not be added automatically mapped and must be manually
configured."
```

In some earlier releases of EDQ it was possible for the value of a workflow parameter to be taken from a process attribute via a selection function even though the process attribute was not mapped to a case source attribute. For example, a workflow parameter called 'tel number' could be populated with the value of either 'cell-phone number' or 'landline number' via a First Non-empty Value selection function, even though neither 'cell-phone number' nor 'landline number' were mapped to an attribute in the case source. This is not permitted in later releases of EDQ. In order to fix this problem, you would need to create two new case source attributes and map 'cell-phone number' and 'landline number' to them. If you didn't do this, the migration would succeed, but in your upgraded system the 'tel number' parameter would not be populated, which may affect the correct workings of Case Management.

Example Two: Let's say that you see an error similar to this one in the premigration report:

```
Parameter X in Source Y for Case Type Z is defined differently across multiple
match processes. Please make these configurations consistent.
```

In later releases of EDQ, a parameter within a case source can only be mapped to a single attribute, but this was not the case in older releases. This error message is reporting that the same case source is used in two (or more) match processes, and that the same parameter has been mapped to more than one attribute in these match processes. Before migrating the data, you should resolve this by either:

- Mapping the parameter to the same attribute in each of the match processes.
 - Or creating a new parameter and mapping this to one of the attributes.
-

If you do not resolve this, the migration would succeed, but in your upgraded system the parameter would not be populated.

Example Three: Let's say that you see an error similar to this one in the premigration report:

```
No metadata for source XXX guid b491939a-e6fd-33d5-9752-241cbe26a378 (Code: 300,551)
```

```
Error executing script: java.lang.IllegalArgumentException
```

In older releases of EDQ, it was possible to import a Case Management configuration file (a .dxc file) that did not contain some historical structures, and this error indicates that this has been done in your old EDQ environment. If a Case Management configuration file that includes all historical case source structures in it, then importing this will solve the problem. If such a file is not available, contact Oracle Support before proceeding with your upgrade.

3. Upgrade the configuration schema.

- a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.

Note: If you are migrating from EDQ release 8.1 or below, then you will need to migrate the EDQ configuration schema in two steps:

- Step 1: migrate the configuration schema to release 9.0.
 - Step 2: migrate the configuration schema to your target 12c release.
-

- b. **Only complete this step if you are migrating from EDQ release 8.1 or below** (see the note directly above):

- i. Acquire a distribution of EDQ 9.0. (you will not need to fully install EDQ 9.0, but you will need access to its migration scripts.) The easiest way to acquire EDQ 9.0 is to download and then unzip the Windows Quick-Start installer for EDQ 9.0.
- ii. Upgrade the configuration schema to release 9.0 by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with the EDQ 9.0 distribution. This should be in `~edq-9_0_x(XXX)\9.0.x.(XXX)` (e.g. `edq-9_0_8_(1103)\9.0.8.(1103)`).
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.

- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to EDQ 9.0 in a Windows environment is:

```
java -jar C:\Users\Fred\Downloads\edq-9_0_8_(1103)\9.0.8.(1103)\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

- c. Upgrade the configuration schema to your EDQ 12c release by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ 12c environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to your new EDQ 12c release in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

4. Upgrade the Case Management schema.

- a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
- b. Upgrade the Case Management schema by issuing the following command:

```
java -jar path_to_migration.jar cm -db dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the Case Management schema in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar cm -db
oracle:mydatabase@localhost:1521/director/director
```

5. Migrate the upgraded configuration and Case Management data from your old EDQ environment's config repository to your new EDQ environment's repository.
 - a. Shutdown your new EDQ environment's WebLogic Managed server (by default called `edq_server1`). One way to do this is from the WebLogic Server Administration Console: within the Domain Structure, expand **Environment** and select **Servers**. Then navigate to the **Control** tab, select the EDQ Managed Server (by default, this is called `edq_server1` and select **Shutdown > When work completes**).
 - b. Use a standard Oracle tool, such as EXPDP, to export your old EDQ repository's configuration schema.

Here is an example of a command that will export your old EDQ repository's configuration schema:

```
expdp config8/config8@localhost:1521/EDQREPOSITORY
schemas=config8 DIRECTORY=DUMP_FILES DUMPFILE=config8-
schema.dmp LOGFILE=config8_2015-09-16.log
```

In this example:

`config8` is your old EDQ Repository's configuration schema (the password is also set to `config8`).

`localhost` is the machine the Oracle Database is running on.

`1521` is the Oracle Database port number.

`EDQREPOSITORY` is the database SID.

DUMP_FILES is your data pump directory.

config8-schema.dmp is the name of the dump file you are going to export.

- c. Use a standard Oracle tool, such as IMPDP, to import the tables (note: only the tables) from your old EDQ repository's configuration schema into your new EDQ repository's configuration schema, replacing any tables with the same name that already exist.

Here is an example of a command that will import the tables from your old EDQ repository's configuration schema into your new EDQ repository's configuration schema, replacing any tables that already exist:

```
impdp DEV_EDQCONFIG/edq@localhost:1521/EDQREPOSITORY
DIRECTORY=DUMP_FILES DUMPFILE=config8-schema.dmp
INCLUDE=TABLE table_exists_action=replace SCHEMAS=config8
REMAP_SCHEMA=config8:DEV_EDQCONFIG
```

In this example:

DEV_EDQCONFIG is your new EDQ Repository's configuration schema (the password is set to edq).

localhost is the machine the Oracle Database is running on.

1521 is the Oracle Database port number.

EDQREPOSITORY is the database SID.

DUMP_FILES is your data pump directory.

config8-schema.dmp is the name of the dump file you are going to import.

Note that the expdp and impdp examples above are just that – examples. You will need to adapt them to cater for the environment you are working in. At a minimum, you will also need to ensure that your database users have the appropriate privileges to run the impdp and expdp commands, and that your data pump directory is set. Consult Oracle Database documentation for more information about the expdp and impdp commands and how to use them.

6. Manually copy some of the configuration folders and files from your old EDQ environment to your new EDQ environment.

*The location of your old EDQ environment's local configuration folder will depend upon the release of EDQ that you are migrating from, which operating system it is installed on, and how it was installed. Note that EDQ release 9.x and lower only had a single configuration directory, which, by convention, was often named **config** (this is the folder you should copy from); EDQ release 11.x and above has two configuration directories – one 'base' and one 'local'. If upgrading from EDQ release 11.x or above, copy from the 'local' configuration directory.*

Your new EDQ environment's local configuration folder will have been created when you installed EDQ in part 1, above. By default, it will be called **oedq.local.home**, and it will be located somewhere under the domain location that was created in part 1. (For example:

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\base_domain\edq\oedq.local.home).

- a. Copy the entire contents of the following folders, including any sub-folders, from your old EDQ environment's local configuration folder (in EDQ releases prior to 11, there was only a single config folder, and you should copy from this) to your new EDQ environment's local configuration folder, merging them with any existing folders that have the same name:

- i. **autorun**
- ii. **av**
- iii. **blueprints**
- iv. **buckets**
- v. **businessrules**
- vi. **capscan**
- vii. **commandarea**
- viii. **dbconnectors**
- ix. **edqp**
- x. **extras**
- xi. **housekeeping**
- xii. **landingarea**
- xiii. **localselection**
- xiv. **localwidgets**
- xv. **match**
- xvi. **properties**
- xvii. **qas**
- xviii. **resources**
- xix. **runprofiles**
- xx. **security**
- xxi. **webservices**

Note that, depending upon the EDQ version that you are migrating from, and how it has been configured, some of the folders listed above may not exist.

- b. Treat the casemanagement folder as follows:

- ii. Once you have made any necessary changes, copy the entire contents of the **triggers** folder, including any sub-folders, from your old EDQ environment's configuration folder to your new EDQ environment's local configuration folder, merging it with any existing folder that has the same name.
 - b. Copy the following files from the root of your old EDQ environment's local configuration folder to the root of your new EDQ environment's local configuration folder:
 - iii. **kfile**
 - iv. **override.properties**
 7. Restart your new EDQ environment's WebLogic Managed server (in other words, the EDQ application server that is called **edq_server1** unless you changed the default name when you installed EDQ). Exactly how you do this depends on your operating system and how your new EDQ environment has been configured (for example, whether you have created a script to start EDQ's Managed WebLogic server).

If you have created a script to start EDQ's Managed WebLogic server, then it is usually possible to restart it via that script.

If you have not created a script to start EDQ's Managed WebLogic server, then follow these steps:

*- Ensure that the WebLogic Node Manager and the WebLogic Admin server are both running. You can stop and start the WebLogic Node Manager and the WebLogic Admin server using .cmd Windows files or Unix/Linux shell scripts that you can find in the **bin** folder of your new EDQ environment's **edq_domain** (An example of the path to this directory on a Windows machine is:*

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\edq_domain\bin). The relevant files will be called: **startNodeManager.cmd / startNodeManager.sh** and **startWebLogic.cmd / startWebLogic.sh**.

*- Open the WebLogic console in a web browser (the default address is usually **http://your-server-name:7001/console**).*

*- In the Domain Structure tree, on the left-side of the screen, select **Environments > Servers** to proceed to the **Summary of Servers** Screen.*

*- In the **Summary of Servers** screen, navigate to the **Control** tab.*

*- In the **Control** tab, select the checkbox for the EDQ Managed Server (by default, the EDQ Managed Server is called **edq_server1**). Then, from the **Shutdown** dropdown, select **When work completes**.*

*- Wait for the EDQ Managed Server to Shutdown, then ensure that its checkbox is selected, and click **Start**.*

8. Open a Web Browser, and navigate to your new EDQ environment's Launchpad (unless you changed any of the default options when installing EDQ, its address will be <http://your-server-name:8001/edq>).

9. From the launchpad, open the Director user interface and log in.

*Note that the user name and password for your new EDQ environment will not necessarily be the same as they were for your old EDQ environment. Unless you changed any of the default options when installing EDQ, the out-of-the-box user name for your new environment will be **weblogic**. You will have provided a password during the installation process.*

10. Verify that all your projects have been successfully migrated.
11. Run each of your jobs or standalone processes to regenerate their results (the results will not have been migrated).

Note that you should run jobs that require a run profile or run label from the Server Console user interface rather than from the Director user interface.

Note also that, if any of your processes use match review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

In some cases a job or process may fail with one of the following error messages:

```
Cannot load data for replacements list reference data "X". Duplicate key "Y".
```

```
Cannot load data for replacements list supplied reference data. Duplicate key "{0}".
```

Prior to release 11, EDQ was tolerant of reference data with duplicate lookup keys. EDQ 11 and higher is less tolerant of Reference Data with duplicate lookup keys. The errors above indicate that your migrated environment includes reference data that has duplicate lookup keys.

In this situation, Oracle recommends that you should deduplicate the reference data in question (on a case-insensitive basis). You can do this either manually if the volume of reference data is small, or by creating an EDQ process and using the Group and Merge processor if the volume of reference data is large.

*Alternatively - although it is not recommended - if previous behavior is required, you can add the following property to your new EDQ environment's **director.properties** file:*

```
replace.transform.allow.duplicate.key = true
```

*Note that the **director.properties** file is located in your new EDQ environment's local configuration directory (by default, this will be called **oedq.local.home**).*

Enterprise Data Quality has a limit of 4000 characters when writing data to the Oracle Database. When EDQ runs snapshots, it truncates fields to ensure that they are within this limit. However, in very rare cases, EDQ processes can generate strings that are longer than the character limit. Unpatched instances of Oracle Watchlist Screening 11.1.1.7 can, for example, generate very long strings that contain delimited lists of watchlist ids matched by a single customer record. When processes produce strings that are longer than the limit, they will fail. This means that it is possible that processes that ran previously under Postgres will fail on Oracle. In such cases the EDQ log would contain an error such as `ORA-01461: can bind a LONG value only for insert into a LONG column`. If you experience this error, contact Oracle support.

12. If you are using Case Management, then, from the Launchpad, open the Case Management user interface, log in and verify that all of your cases and alerts have been successfully migrated.

If Case Management appears to be missing cases and alerts (for example, if a filter that should return several cases and alerts in fact returns none), it is possible that the case management indexes need to be re-generated. Follow this procedure to re-generate Case Management indexes:

- Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is **C:\Program Files\Java\jdk1.8.0_40\bin**).
 - Run the `jconsole` application (how you run this will depend upon your operating system – you may simply be able to double-click the application).
 - The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Local Process** radio button, and then highlight the top-most instance of the **weblogic.server** that appears in the list of local processes (this should be EDQ's Managed WebLogic Server). Input the username for your new EDQ environment (by default, this is **weblogic**), and password. Then click **Connect**.
 - Navigate to the **Mbeans** tab.
 - Within the tree, on the left-side of the Window, expand **edq** -> **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.
 - Ensure that **clear** is set to **true**.
 - Click **reindex**. An Info dialog box will report that the Method was successfully invoked.
-

13. Within Case Management, verify that you can successfully search for and find Cases and Alerts.

*If you cannot successfully search for and find Cases and Alerts, this is likely to indicate that your flags.xml file is incorrect. The flags.xml file is located in your new EDQ environment's casemanagement directory, which is itself within the local configuration directory (called **oedq.local.home** by default). flags.xml contains information about extended attributes. A fresh EDQ installation comes with a flags.xml file that includes a number of default extended attributes. You can add new extended attributes to flags.xml, but if you removed or changed the default extended attributes that came with the original file, then EDQ will not be able to properly index Cases and Alerts, and you will therefore not be able to successfully search for them. You should resolve this by putting the default extended attributes back into flags.xml, and reindexing cases and alerts. (How to reindex is explained in the note above this one.). One way to do this is to copy the default extended attributes from a fresh version of flags.xml. You will find a fresh version of flags.xml in your new EDQ environment's base configuration folder, which is called **oedq.home** by default.*

14. Out-of-the-box, your new EDQ environment's Launchpad may not display links to all of your EDQ user interfaces. To configure the launchpad, click **Log In**, enter your EDQ user name and password, and then select **Administration > Launchpad Configuration**. You can then select the user interfaces that you want to display on the Launchpad.
15. All of the user groups from your old EDQ environment will have been migrated to your new EDQ environment. However, the users from your old environment will no longer be active. This is because in your new EDQ environment, users are managed within your new WebLogic environment. You will need to:
 - a. Use the WebLogic Server Administration Console to create a series of WebLogic user groups to mirror each of your old environment's user groups.

*To get to the part of the WebLogic Server Administration Console that you will require, in the Domain Structure select **Security Realms**. Then click your security realm (by default this is called **myrealm**) and navigate to the **Users and Groups** tab.*

- b. Use the WebLogic Server Administration Console to recreate all of your old environment's users, assigning each of these users to one of the WebLogic user groups you created in the previous step.
- c. Use the EDQ Launchpad (**Administration > External Groups**) to map each of the WebLogic user groups you created (which EDQ will see as **External Groups**) to the appropriate EDQ user group.

Upgrade Path 3:

From any Release of EDQ Using an Oracle / Tomcat Platform to EDQ 12.1.3 Using an Oracle / WebLogic Platform

Overview

Upgrading from a release of EDQ that uses the Oracle database and the Tomcat application server as its platform to EDQ 12.1.3 using the Oracle Database and Oracle WebLogic as its platform is essentially a two-part process:

- Part 1 – Install EDQ 12.1.3
- Part 2 – Migrate configuration data from your old EDQ environment to your new EDQ 12.1.3 environment. This entails using scripts provided with the EDQ 12.1.3 release to reshape and move data. It also entails manually copying some files.

Part 1: Install EDQ 12.1.3

1. Install Enterprise Data Quality 12.1.3 using the Oracle WebLogic Server as its Java Application Server and the Oracle Database as its structured query language (SQL) relational database management system (RDBMS); refer to <http://add-a-link-to-the-12.1.3-installation-doc> for more information about how to do this. During part 2, you will need to know the following information about your EDQ 12.1.3 environment:
 - a. The repository database's Host Name, Port Number and Database ID.
 - b. The schema owner and password for the config and results schemas.
 - c. The environment's domain location.

Part 2: Migrate the data from the old DIRECTOR schema to the new EDQCONFIG schema

1. Back up elements of your old EDQ environment.

During part 2, you will migrate configuration data from your old EDQ repository to your new EDQ repository, and you will also copy configuration files from your old environment to your new environment. You backup elements of your old environment (that is, the configuration data and the configuration files) so that you can easily restart this process should anything go seriously wrong.

- a. Before you back up your old EDQ environment you need to shut down EDQ's Tomcat application server. How you do this will vary by operating system.
 - i. In Windows there is usually a Windows service for the application server. You can stop this via the **Control Panel > Administrative Tools > Services**.

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the Windows service will be called **Datanomic Application Server**.*

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the Windows service will be called **Oracle EDQ Application Server**.

- ii. In Linux or Unix environments, it is likely that the application is started via an init.d service (it's quite likely you will find them in **/etc/rc.d/init.d**). The service name and method of stopping it will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name stop** from a terminal window will often work.
- b. Take a copy of your old EDQ environment's configuration folder, which is likely to be called either **config** or **oedq_local_home**.
 - i. For Windows environments, see the note below.

If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the configuration folder will be called **config**, and its default location is **C:\Program Files\Datanomic\dnDirector\config**.

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the configuration folder will be called **oedq_local_home**. Its location will depend upon the version of Windows you have installed on. On a Windows 7 machine, for example, the default location of **oedq_local_home** is:
C:\ProgramData\Oracle\Enterprise Data Quality\oedq_local_home.

- ii. In Linux or Unix environments, there is no default location or name for the EDQ configuration folder, and you may need to ask your system administrator to tell you what the path is. Having said that, by convention, the folder is often called either **config** or **oedq.local.home** or **oedq_local_home**.
- c. Backup your old EDQ environment's database repository using the appropriate Oracle tools.
- d. **Do not** restart your old EDQ environment's application server.

In case of a serious error during the upgrade you are about to perform (for example, if the migration script should fail to complete, and you are left with a schema that is stuck in between your old version of EDQ and your new version of EDQ), you can now fairly easily start the process again from scratch. How?

- Stop your old EDQ environment's application server.
 - Delete the original versions of the folder you have backed up.
 - Copy the backup into its original location.
 - Restore the database repository from the backup.
- ...And then you can start the migration process again from scratch.
-

- 2. Run and examine the premigration report. This will reveal any problems with the Case Management schema (which is, strictly speaking, not a schema at all, but just a set of tables within EDQ's config schema). If any problems are revealed, you should fix them before proceeding to step 3.

- a. Open a command line tool:
 - i. In a Windows environment, open a Windows Command Prompt, running it as administrator (for example, right-click Command Prompt in the All Programs > Accessories group, and select **Run as administrator**).
 - ii. In a Unix or Linux environment, open a terminal window.
- b. Change directory to your Java Development Kit's bin folder. You can do this by issuing the command `cd path_to_the_jdk_bin_folder` (for example, in a Windows environment you might enter a command similar to: `cd C:\Program Files\Java\jdk1.8.0_40\bin`).
- c. Run the pre-migration report by issuing the following command:

```
java -jar path_to_migration.jar premigrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to run the premigration report in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar  
preigrate oracle:mydatabase@localhost:1521/director/director
```

- d. If the Premigration report reveals any problems, fix them, and run the report again before proceeding further. (If the premigration report returns no output, by the way, that's a good thing: it means no problems were found.)

Examples of Problems Revealed by the Premigration Report, With Suggested Fixes

Example One: Let's say that you see the following error in the premigration report:

```
Parameter x in Source y for Case Type z using Selection Function aa "  
" is using an attribute under input 'bb that is not mapped to case management.  
This input will not be added automatically mapped and must be manually  
configured."
```

In some earlier releases of EDQ it was possible for the value of a workflow parameter to be taken from a process attribute via a selection function even though the process attribute was not mapped to a case source attribute. For example, a workflow parameter called 'tel number' could be populated with the value of either 'cell-phone number' or 'landline number' via a First Non-empty Value selection function, even though neither 'cell-phone number' nor 'landline number' were mapped to an attribute in the case source. This is not permitted in later releases of EDQ. In order to fix this problem, you would need to create two new case source attributes and map 'cell-phone number' and 'landline number' to them. If you didn't do this, the migration would succeed, but in your upgraded system the 'tel number' parameter would not be populated, which may affect the correct workings of Case Management..

Example Two: Let's say that you see an error similar to this one in the premigration report:

```
Parameter X in Source Y for Case Type Z is defined differently across multiple  
match processes. Please make these configurations consistent.
```

In later releases of EDQ, a parameter within a case source can only be mapped to a single attribute, but this was not the case in older releases. This error message is reporting that the same case source is used in two (or more) match processes, and that the same parameter has been mapped to more than one attribute in these match processes. Before migrating the data, you should resolve this by either:

- Mapping the parameter to the same attribute in each of the match processes.
- Or creating a new parameter and mapping this to one of the attributes.

If you do not resolve this, the migration would succeed, but in your upgraded system the parameter would not be populated.

Example Three: Let's say that you see an error similar to this one in the premigration report:

```
No metadata for source XXX guid b491939a-e6fd-33d5-9752-241cbe26a378 (Code:
300,551)
Error executing script: java.lang.IllegalArgumentException
```

In older releases of EDQ, it was possible to import a Case Management configuration file (a .dxc file) that did not contain some historical structures, and this error indicates that this has been done in your old EDQ environment. If a Case Management configuration file that includes all historical case source structures in it, then importing this will solve the problem. If such a file is not available, contact Oracle Support before proceeding with your upgrade.

3. Upgrade the configuration schema.
 - a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
-

Note: If you are migrating from EDQ release 8.1 or below, then you will need to migrate the EDQ configuration schema in two steps:

- Step 1: migrate the configuration schema to release 9.0.
 - Step 2: migrate the configuration schema to your target 12c release.
-

- b. **Only complete this step if you are migrating from EDQ release 8.1 or below** (see the note directly above):

- i. Acquire a distribution of EDQ 9.0. (you will not need to fully install EDQ 9.0, but you will need access to its migration scripts.) The easiest way to acquire EDQ 9.0 is to download and then unzip the Windows Quick-Start installer for EDQ 9.0.
 - ii. Upgrade the configuration schema to release 9.0 by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with the EDQ 9.0 distribution. This should be in `~edq-9_0_x(XXX)\9.0.x.(XXX)` (e.g. `edq-9_0_8(1103)\9.0.8.(1103)`).
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.

- password is the password of the database user.

An example of a command to migrate the configuration schema to EDQ 9.0 in a Windows environment is:

```
java -jar C:\Users\Fred\Downloads\edq-9_0_8_(1103)\9.0.8.(1103)\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

- c. Upgrade the configuration schema to your EDQ 12c release by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- path_to_migration.jar is the path to the **migration.jar** file that was installed with your new EDQ 12c environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- dbtype is postgresql for PostgreSQL or oracle for Oracle.
- dbid is the database name for PostgreSQL or the SID for Oracle.
- host is the name of the database host machine.
- port is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- user is the user ID that EDQ uses to connect to the Director database.
- password is the password of the database user.

An example of a command to migrate the configuration schema to your new EDQ 12c release in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

4. Upgrade the Case Management schema.

- Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
- Upgrade the Case Management schema by issuing the following command:

```
java -jar path_to_migration.jar cm -db dbtype:dbid@host:port/user/password
```

Where:

- path_to_migration.jar is the path to the **migration.jar** file that was installed with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- dbtype is postgresql for PostgreSQL or oracle for Oracle.

- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the Case Management schema in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar cm -db oracle:mydatabase@localhost:1521/director/director
```

5. Migrate the upgraded configuration and Case Management data from your old EDQ environment's config repository to your new EDQ environment's repository.
 - d. Shutdown your new EDQ environment's WebLogic Managed server (by default called `edq_server1`). One way to do this is from the WebLogic Server Administration Console: within the Domain Structure, expand **Environment** and select **Servers**. Then navigate to the **Control** tab, select the EDQ Managed Server (by default, this is called `edq_server1` and select **Shutdown > When work completes**).
 - e. Use a standard Oracle tool, such as EXPDP, to export your old EDQ repository's configuration schema.

Here is an example of a command that will export your old EDQ repository's configuration schema:

```
expdp config8/config8@localhost:1521/EDQREPOSITORY schemas=config8 DIRECTORY=DUMP_FILES DUMPFILE=config8-schema.dmp LOGFILE=config8_2015-09-16.log
```

In this example:

`config8` is your old EDQ Repository's configuration schema (the password is also set to `config8`).

`localhost` is the machine the Oracle Database is running on.

`1521` is the Oracle Database port number.

`EDQREPOSITORY` is the database SID.

`DUMP_FILES` is your data pump directory.

`config8-schema.dmp` is the name of the dump file you are going to export.

- f. Use a standard Oracle tool, such as IMPDP, to import the tables (note: only the tables) from your old EDQ repository's configuration schema into your new EDQ repository's configuration schema, replacing any tables with the same name that already exist.

Here is an example of a command that will import the tables from your old EDQ repository's configuration schema into your new EDQ repository's configuration schema, replacing any tables that already exist:

```
impdp DEV_EDQCONFIG/edq@localhost:1521/EDQREPOSITORY
DIRECTORY=DUMP_FILES DUMPFILE=config8-schema.dmp
INCLUDE=TABLE table_exists_action=replace SCHEMAS=config8
REMAP_SCHEMA=config8:DEV_EDQCONFIG
```

In this example:

DEV_EDQCONFIG is your new EDQ Repository's configuration schema (the password is set to edq).

localhost is the machine the Oracle Database is running on.

1521 is the Oracle Database port number.

EDQREPOSITORY is the database SID.

DUMP_FILES is your data pump directory.

config8-schema.dmp is the name of the dump file you are going to import.


Note that the expdp and impdp examples above are just that – examples. You will need to adapt them to cater for the environment you are working in. At a minimum, you will also need to ensure that your database users have the appropriate privileges to run the impdp and expdp commands, and that your data pump directory is set. Consult Oracle Database documentation for more information about the expdp and impdp commands and how to use them.

6. Manually copy some of the configuration folders and files from your old EDQ environment to your new EDQ environment.

*The location of your old EDQ environment's local configuration folder will depend upon the release of EDQ that you are migrating from, which operating system it is installed on, and how it was installed. You should already know it, as you entered it during step 5d, above. Note that EDQ release 9.x and lower only had a single configuration directory, which was usually called **config** (this is the folder you should copy from); EDQ release 11.x and above has two configuration directories – one 'base' and one 'local'. If upgrading from EDQ release 11.x or above, copy from the 'local' configuration directory.*

*Your new EDQ environment's local configuration folder will have been created when you installed EDQ in part 1, above. By default, it will be called **oedq.local.home**, and it will be located somewhere under the domain location that was created in part 1. (For example:*

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\base_domain\config\fmwconfig\edq\oedq.local.home).

- 
- a. Copy the entire contents of the following folders, including any sub-folders, from your old EDQ environment's local configuration folder (in EDQ releases prior to 11, there was only a single config folder, and you should copy from this) to your new EDQ environment's local configuration folder, merging them with any existing folders that have the same name:

- i. **autorun**
- ii. **av**
- iii. **blueprints**
- iv. **buckets**
- v. **businessrules**
- vi. **capscan**
- vii. **commandarea**
- viii. **dbconnectors**
- ix. **edqp**
- x. **extras**
- xi. **housekeeping**
- xii. **landingarea**
- xiii. **localselection**
- xiv. **localwidgets**
- xv. **match**
- xvi. **properties**
- xvii. **qas**
- xviii. **resources**
- xix. **runprofiles**
- xx. **security**
- xxi. **webservices**

Note that, depending upon the EDQ version that you are migrating from, and how it has been configured, some of the folders listed above may not exist.

- b. Treat the casemanagement folder as follows:
- i. Copy entire contents of the **casemanagement** folder, **except** for the indexes folder and the **flags.xml** file.
 - ii. If you **have not** made any changes to the **flags.xml** file, copy that file too.

- iii. If you **have** made changes to the **flags.xml** file, manually copy any new extended attributes that you added, and paste them into the version of **flags.xml** that you will find in your new EDQ environment's **casemanagement** folder. However, ensure that you do not remove or modify any of the default extended attributes that you will find within your new EDQ environment's **casemanagement** folder.
- c. Treat the **localgadgets** folder as follows:
- i. If you are migrating from EDQ release 8.1.x or above, copy entire contents of the **localgadgets** folder.
 - ii. If you are migrating from EDQ release 8.0.x or below, then copy the entire contents of the **localgadgets** folder except for any selection functions. You will need to create new versions of any tailored selection functions and place them in the **localselection** sub-folder of your new EDQ environment's local configuration folder. Existing custom selection functions will not work after upgrade.
- d. Treat the **notification** folder as follows:
- i. If you are migrating from EDQ release 9.0.x or above, copy the entire contents of the **notification** folder.
 - ii. If you are migrating from EDQ release 8.1.x or below, copy the entire contents of the **notification** folder except for the **jobs** and **eventlog** sub-folders.

*The **jobs** and **eventlog** folders cannot be copied from EDQ release 8.1.x or lower because job template code was restructured in EDQ 9.0 and therefore modified versions of the job templates created prior to release 9.0 will not work. (If you do erroneously copy these folders, you will find that job notification e-mails will not work, and that the job notifications will not be visible in the event log.) If the contents of these folders were not modified in your old EDQ environment, then you need do nothing: the default files provided with your new EDQ environment will work out-of-the-box. But if you have carried out any modifications, then any changes you made will need to be re-applied to the new versions of the templates - copying over the old versions will not work.*

- c. Treat the **triggers** folder as follows:
- v. Check through each of the files in the triggers directory, and in any sub-directories. If you find the following line in any file:

```
importClass("com.datanomic.director.guihelper.users.server.UserCache");
```


Change it to:

```
importClass("com.datanomic.userauth.server.cache.UserCache");
```
 - vi. Once you have made any necessary changes, copy the entire contents of the **triggers** folder, including any sub-folders, from your old EDQ environment's configuration folder to your new EDQ environment's local configuration folder, merging it with any existing folder that has the same name.

- d. Copy the following files from the root of your old EDQ environment's local configuration folder to the root of your new EDQ environment's local configuration folder:

vii. kfile

viii. override.properties

7. Restart your new EDQ environment's WebLogic Managed server (in other words, the EDQ application server that is called **edq_server1** unless you changed the default name when you installed EDQ). Exactly how you do this depends on your operating system and how your new EDQ environment has been configured (for example, whether you have created a script to start EDQ's Managed WebLogic server).

If you have created a script to start EDQ's Managed WebLogic server, then it is usually possible to restart it via that script.

If you have not created a script to start EDQ's Managed WebLogic server, then follow these steps:

*- Ensure that the WebLogic Node Manager and the WebLogic Admin server are both running. You can stop and start the WebLogic Node Manager and the WebLogic Admin server using .cmd Windows files or Unix/Linux shell scripts that you can find in the **bin** folder of your new EDQ environment's edq_domain (An example of the path to this directory on a Windows machine is:*

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\edq_domain\bin). The relevant files will be called: **startNodeManager.cmd / startNodeManager.sh** and **startWebLogic.cmd / startWebLogic.sh**.

*- Open the WebLogic console in a web browser (the default address is usually **http://your-server-name:7001/console**).*

- In the Domain Structure tree, on the left-side of the screen, select Environments > Servers to proceed to the Summary of Servers Screen.

*- In the Summary of Servers screen, navigate to the **Control** tab.*

*- In the Control tab, select the checkbox for the EDQ Managed Server (by default, the EDQ Managed Server is called **edq_server1**). Then, from the **Shutdown** dropdown, select **When work completes**.*

*- Wait for the EDQ Managed Server to Shutdown, then ensure that its checkbox is selected, and click **Start**.*

8. Open a Web Browser, and navigate to your new EDQ environment's Launchpad (unless you changed any of the default options when installing EDQ, its address will be <http://your-server-name:8001/edq>).

9. From the launchpad, open the Director user interface and log in.

*Note that the user name and password for your new EDQ environment will not necessarily be the same as they were for your old EDQ environment. Unless you changed any of the default options when installing EDQ, the out-of-the-box user name for your new environment will be **weblogic**. You will have provided a password during the installation process.*

10. Verify that all your projects have been successfully migrated.
11. Run each of your jobs or standalone processes to regenerate their results (the results will not have been migrated).

Note that you should run jobs that require a run profile or run label from the Server Console user interface rather than from the Director user interface.

Note also that, if any of your processes use match review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

In some cases a job or process may fail with one of the following error messages:

```
Cannot load data for replacements list reference data "X". Duplicate key "Y".
```

```
Cannot load data for replacements list supplied reference data. Duplicate key "{0}".
```

Prior to release 11, EDQ was tolerant of reference data with duplicate lookup keys. EDQ 11 and higher is less tolerant of Reference Data with duplicate lookup keys. The errors above indicate that your migrated environment includes reference data that has duplicate lookup keys.

In this situation, Oracle recommends that you should deduplicate the reference data in question (on a case-insensitive basis). You can do this either manually if the volume of reference data is small, or by creating an EDQ process and using the Group and Merge processor if the volume of reference data is large.

Alternatively - although it is not recommended - if previous behavior is required, you can add the following property to your new EDQ environment's **director.properties** file:

```
replace.transform.allow.duplicate.key = true
```

Note that the **director.properties** file is located in your new EDQ environment's local configuration directory (by default, this will be called **oedq.local.home**).

Enterprise Data Quality has a limit of 4000 characters when writing data to the Oracle Database. When EDQ runs snapshots, it truncates fields to ensure that they are within this limit. However, in very rare cases, EDQ processes can generate strings that are longer than the character limit. Unpatched instances of Oracle Watchlist Screening 11.1.1.7 can, for example, generate very long strings that contain delimited lists of watchlist ids matched by a single customer record. When processes produce strings that are longer than the limit, they will fail. This means that it is possible that processes that ran previously under Postgres will fail on Oracle. In such cases the EDQ log would contain an error such as ORA-01461: can bind a LONG value only for insert into a LONG column. If you experience this error, contact Oracle support.

12. If you are using Case Management, then, from the Launchpad, open the Case Management user interface, log in and verify that all of your cases and alerts have been successfully migrated.


If Case Management appears to be missing cases and alerts (for example, if a filter that should return several cases and alerts in fact returns none), it is possible that the case management indexes need to be re-generated. Follow this procedure to re-generate Case Management indexes:

- *Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is **C:\Program Files\Java\jdk1.8.0_40\bin**).*
 - *Run the jconsole application (how you run this will depend upon your operating system – you may simply be able to double-click the application).*
 - *The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Local Process** radio button, and then highlight the top-most instance of the **weblogic.server** that appears in the list of local processes (this should be EDQ's Managed WebLogic Server). Input the username for your new EDQ environment (by default, this is **weblogic**), and password. Then click **Connect**.*
 - *Navigate to the **Mbeans** tab.*
 - *Within the tree, on the left-side of the Window, expand **edq** - > **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.*
 - *Ensure that **clear** is set to **true**.*
 - *Click **reindex**. An Info dialog box will report that the Method was successfully invoked.*
-

13. Within Case Management, verify that you can successfully search for and find Cases and Alerts.

*If you cannot successfully search for and find Cases and Alerts, this is likely to indicate that your flags.xml file is incorrect. The flags.xml file is located in your new EDQ environment's casemanagement directory, which is itself within the local configuration directory (called **oedq.local.home** by default). flags.xml contains information about extended attributes. A fresh EDQ installation comes with a flags.xml file that includes a number of default extended attributes. You can add new extended attributes to flags.xml, but if you removed or changed the default extended attributes that came with the original file, then EDQ will not be able to properly index Cases and Alerts, and you will therefore not be able to successfully search for them. You should resolve this by putting the default extended attributes back into flags.xml, and reindexing cases and alerts. (How to reindex is explained in the note above this one.). One way to do this is to copy the default extended attributes from a fresh version of flags.xml. You will find a fresh version of flags.xml in your new EDQ environment's base configuration folder, which is called **oedq.home** by default.*

14. Out-of-the-box, your new EDQ environment's Launchpad may not display links to all of your EDQ user interfaces. To configure the launchpad, click **Log In**, enter your EDQ user name and password, and then



select **Administration > Launchpad Configuration**. You can then select the user interfaces that you want to display on the Launchpad.

15. All of the user groups from your old EDQ environment will have been migrated to your new EDQ environment. However, the users from your old environment will no longer be active. This is because in your new EDQ environment, users are managed within WebLogic rather than within EDQ itself. You will need to:
 - a. Use the WebLogic Server Administration Console to create a series of WebLogic user groups to mirror each of your EDQ user groups.

*To get to the part of the WebLogic Server Administration Console that you will require, in the Domain Structure select **Security Realms**. Then click your security realm (by default this is called **myrealm**) and navigate to the **Users and Groups** tab.*

- b. Use the WebLogic Server Administration Console to recreate all of your old EDQ users, assigning each of these users to one of the WebLogic user groups you created in the previous step.
 - c. Use the EDQ Launchpad (**Administration > External Groups**) to map each of the WebLogic user groups you created (which EDQ will see as **External Groups**) to the appropriate EDQ user group.

Upgrade Path 4:

From any Release of EDQ on a PostgreSQL / Tomcat Platform to EDQ 12.1.3 Using a PostgreSQL / Tomcat Platform

Overview

Upgrading from an installation of EDQ that uses the PostgreSQL database and the Tomcat application server as its platform to EDQ 12.1.3 using the PostgreSQL Database and the Tomcat application server is not recommended, but is possible. It is essentially a two-part process:

- Part 1 – Perform a Partial Installation of EDQ 12.1.3
- Part 2 – Upgrade Your Old EDQ Environment.

Note that the recommended path is to upgrade to EDQ 12.1.3 or 12.2.1 using the Oracle Database and the WebLogic application server as its platform

Part 1: Perform a Partial Installation of EDQ 12.1.3

Note that when installing EDQ it is often possible to complete the same task in different ways. For example, when you need to manipulate settings in the Oracle Database, you can usually do this through the Enterprise Manager user interface, or by issuing a SQL statement at the command line. In the instructions below, an example of one possible method of completing a task is often given. However, so long as the task is completed, you should feel free to use the approach that you are most comfortable with.

1. Ensure that your environment satisfies all EDQ 12.1.3 System Requirements, as outlined in section 1.4 of the Installing and Configuring Oracle Enterprise Data Quality guide, and upgrade your environment if necessary.
2. Download EDQ 12.1.3, as described in section 1.5 of the Installing and Configuring Oracle Enterprise Data Quality guide.
3. If your environment's current Java Development Kit does not support EDQ 12.1.3, install a Java Development Kit to support EDQ 12.1.3, as described in section 2.2 of the Installing and Configuring Oracle Enterprise Data Quality guide.
4. If your environment's PostgreSQL database does not support EDQ 12.1.3, upgrade it to a version that will support EDQ 12.1.3. See section 2.4.2 of the Installing and Configuring Oracle Enterprise Data Quality guide for more information about how to configure PostgreSQL for EDQ.
5. Install Enterprise Data Quality 12.1.3 assets. This step entails installing the Oracle Fusion Middleware EDQ distribution. This is usually downloaded as a zip file that contains a single file with a name similar to **fmw_12.1.3.x.x_edq.jar**. Running this jar file invokes the Oracle Universal Installer.
 - a. Open a command line tool and navigate to your Java Development Kit's bin folder.
 - b. Open the Oracle Universal Installer to install Enterprise Data Quality assets by issuing the following command:

```
java -jar EDQ_installation_jar_file.jar
```

Where:

- `EDQ_installation_jar_file.jar` is the path to the Enterprise Data Quality Installation file.

An example of a command to Open the Oracle Universal Installer to install Enterprise Data Quality in a Windows environment is:

```
java -jar C:\Users\peter\Downloads\fmw_12.1.3_edq.jar
```

After some time, the Oracle Enterprise Data Quality 12c Installation window will open at the Welcome screen.

- At the Welcome screen click **Next >**.
- At the Auto Updates screen, make a choice about whether or not to auto-update (and if so how), and click **Next >**.
- At the Installation Location screen, select a directory where you would like to install EDQ, and then click **Next >**. For the purpose of this document, we will refer to this directory as the EDQ installation folder.

*Note that, in order to install EDQ for the Tomcat application server, the EDQ installation folder **must not be** located within any existing Fusion Middleware home folder (if your system has one).*

- At the Installation Type screen, select **Installation for Other Platforms**, and click **Next >**.
 - At the Prerequisite Checks screen, wait until all of the prerequisite are passed (if any fail, address the issues that caused them to fail, and try this step again until all prerequisite checks are passed), and then click **Next >**.
 - At the Installation Summary screen, click **Install**.
 - At the Installation Progress screen, wait until the installation has completed, and then click **Finish**.
You should find that your EDQ installation folder has been populated with a number of sub-folders.
- Create your new EDQ environment's configuration directories.
 - Create two empty directories to contain your new EDQ environment's configuration files. For the purposes of this document we will assume that you call these **oedq.home** and **oedq.local.home** respectively, though they do not necessarily have to be given those names.

For example:

On Linux and UNIX operating systems:

```
/opt/edqconfig/oedq.home
```

/opt/edqconfig/oedq.local.home

On the Windows operating system:

C:\edqconfig\oedq.home

C:\edqconfig\oedq.local.home

The first directory (**oedq.home**) is the base configuration directory and will contain the configuration files that should not be changed post-installation. The second directory (**oedq.local.home**) is the local configuration directory and will contain any custom settings that you create.

- b. Ensure that your application server user has read and write access to the two directories you create.
7. Run the EDQ Configapp to populate the configuration directories.
- a. Make certain that your PostgreSQL database is running.
 - b. Open a command line tool and navigate to your Java Development Kit's bin folder.
 - c. Start the Configapp by issuing the following command:

```
java -jar path_to_configapp.jar
```

Where:

- `path_to_configapp.jar` is the path to the Configapp.

Note that the configapp is located in your EDQ installation folder's **edq/oracle.edq** directory.

An example of a command to start the Configapp in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\configapp.jar
```

The Configapp will open.

- d. Click **Begin**.
 - a. At the Configuration directory screen, leave **Create a new configuration directory** selected, and select the **oedq.home** directory that you in the previous step as the Base Configuration directory location, and select the **oedq.local.home** directory that you created above as the Local Configuration directory location. Then click **Next >**.
 - b. At the Functional Pack screen, select the functional packs you want and are licensed to use, and then click **Next >** (ensure that you scroll right to the bottom of this screen to see all of the functional packs).

All Functional Packs are needed if you want to use the full capabilities of the EDQ Customer Data Services Pack on your EDQ server, such as for integration with Siebel Customer Relationship Management or Universal Customer Master.

All Functional Packs are needed if you want to install Oracle Watchlist Screening on your EDQ server.

- c. At the Configure Config repository database screen, leave the **Selection Type** set to **Properties**. Enter the user name and password for your old EDQ environment's configuration schema. Select the option for the type of database (Postgres). Enter the host name of the system that contains the database, its port number, and the Database Id. Ensure that you configured the connection to the database correctly by clicking **Test**. Then click **Next >** to continue.
- d. At the Configure Results repository database screen, leave the **Selection Type** set to **Properties**. Enter the user name and password for your old EDQ environment's results schema. Select the option for the type of database (Postgres). Enter the host name of the system that contains the database, its port number, and the Database Id. Ensure that you configured the connection to the database correctly by clicking **Test**. Then click **Next >** to continue.
- e. At the Selected options screen, Review the summary of your selections. Use **Back** to modify any of your selections, or click **Finish** to complete the configuration and continue.

Part 2: Upgrade your Old EDQ Environment

1. Back up your old EDQ environment before you make any changes to it.
 - a. Before you back up your old EDQ environment you need to shut down EDQ's Tomcat application server and the EDQ repository (which is held in Postgresql databases). How you do this will vary by operating system.
 - i. In Windows there are usually three Windows services: one for the application server, and one for each of the Director and Results databases. You can stop these via the **Control Panel > Administrative Tools > Services**.

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the three Windows services will be called **Datanomic Application Server**, **Datanomic dnDirector Store** and **Datanomic dnResults Store**.*

*If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the three Windows services will be called **Oracle EDQ Application Server**, **Oracle EDQ Config Store** and **Oracle EDQ Results Store** respectively.*

- ii. In Linux or Unix environments, it is likely that the application server and repository database(s) are started via init.d services (it's quite likely you will find them in **/etc/rc.d/init.d**). The service name and method of stopping them will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name stop** from a terminal window will often work.
 - b. Take a copy of the folder that contains the Postgresql data, which is usually called **data**. The location and name of this folder may vary depending on your operating system and type of installation.
 - iii. For Windows environments, see the note below.

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the folder that contains the Postgresql data will be **C:\Program Files\Datanomic\dnDirector\data**.*

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the folder that contains the Postgresql data will depend upon which version of Windows you have installed EDQ on. In Windows 7, for example, the default location is **C:\ProgramData\Oracle\Enterprise Data Quality\data** (note that you may have to change Windows Folder Options to view hidden files and folders before you can see this location).

- iv. In Linux or Unix environments, a common location is **/opt/PostgreSQL/version_number/data**. However, there is no single standard location, and you may need to ask your system administrator to tell you what the path is.
- c. Take a copy of your old EDQ environment's configuration folder, which is likely to be called either **config** or **oedq_local_home**.
- v. For Windows environments, see the note below.

If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the configuration folder will be called **config**, and its default location is **C:\Program Files\Datanomic\dnDirector\config**.

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the configuration folder will be called **oedq_local_home**. Its location will depend upon the version of Windows you have installed on. On a Windows 7 machine, for example, the default location of **oedq_local_home** is:
C:\ProgramData\Oracle\Enterprise Data Quality\oedq_local_home.

- vi. In Linux or Unix environments, there is no default location or name for the EDQ configuration folder, and you may need to ask your system administrator to tell you what the path is. Having said that, by convention, the folder is often called either **config** or **oedq.local.home** or **oedq_local_home**.
- d. Restart your old EDQ environment's Respository (that is to say, the Results and Config databases), but **do not** restart its application server.
 - vii. For Windows environments, start two of the three Windows services that you stopped above (do not restart the service for the application server though).
 - viii. In Linux or Unix environments, start the init.d service for the repository database, but do not start the init.d service for the application server. The service name and method of starting the repository database will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name start** from a terminal window will often work.

In case of a serious error during the upgrade you are about to perform (for example, if the migration script should fail to complete, and you are left with a schema that is stuck in between your old version of EDQ and your new version of EDQ), you can now fairly easily start the process again from scratch. How?

- Stop your old EDQ environment's application server and repository.
 - Delete the original versions of the folders you have backed up.
 - Copy the backups into the original locations.
 - Restart your old EDQ environment's repository.
- ...And then you can start the migration process again from scratch.*
-

2. Run and examine the premigration report. This will reveal any problems with the Case Management schema (which is, strictly speaking, not a schema at all, but just a set of tables within EDQ's config schema). If any problems are revealed, you should fix them before proceeding to step 3.
 - d. Open a command line tool:
 - i. In a Windows environment, open a Windows Command Prompt, running it as administrator (for example, right-click Command Prompt in the All Programs > Accessories group, and select **Run as administrator**).
 - ii. In a Unix or Linux environment, open a terminal window.
 - e. Change directory to your Java Development Kit's bin folder. You can do this by issuing the command `cd path_to_the_jdk_bin_folder` (for example, in a Windows environment you might enter a command similar to: `cd C:\Program Files\Java\jdk1.8.0_40\bin`).
 - f. Run the pre-migration report by issuing the following command:

```
java -jar path_to_migration.jar premigrate dbtype:dbid@host:port/user/password
```

Where:
 - `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with your new EDQ environment. This should be in `~ORACLE_HOME\edq\oracle.edq`
 - `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
 - `dbid` is the database name for PostgreSQL or the `SID` for Oracle.
 - `host` is the name of the database host machine.
 - `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
 - `user` is the user ID that EDQ uses to connect to the Director database.
 - `password` is the password of the database user.

An example of a command to run the premigration report in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar
premigrate pgsql:mydatabase@localhost:5432/director/director
```

- g. If the Premigration report reveals any problems, fix them, and run the report again before proceeding further. (If the premigration report returns no output, by the way, that's a good thing: it means no problems were found.)

Examples of Problems Revealed by the Premigration Report, With Suggested Fixes

Example One: Let's say that you see the following error in the premigration report:

```
Parameter x in Source y for Case Type z using Selection Function aa "
" is using an attribute under input 'bb that is not mapped to case management.
This input will not be added automatically mapped and must be manually
configured."
```

In some earlier releases of EDQ it was possible for the value of a workflow parameter to be taken from a process attribute via a selection function even though the process attribute was not mapped to a case source attribute. For example, a workflow parameter called 'tel number' could be populated with the value of either 'cell-phone number' or 'landline number' via a First Non-empty Value selection function, even though neither 'cell-phone number' nor 'landline number' were mapped to an attribute in the case source. This is not permitted in later releases of EDQ. In order to fix this problem, you would need to create two new case source attributes and map 'cell-phone number' and 'landline number' to them. If you didn't do this, the migration would succeed, but in your upgraded system the 'tel number' parameter would not be populated, which may affect the correct workings of Case Management..

Example Two: Let's say that you see an error similar to this one in the premigration report:

```
Parameter X in Source Y for Case Type Z is defined differently across multiple
match processes. Please make these configurations consistent.
```

In later releases of EDQ, a parameter within a case source can only be mapped to a single attribute, but this was not the case in older releases. This error message is reporting that the same case source is used in two (or more) match processes, and that the same parameter has been mapped to more than one attribute in these match processes. Before migrating the data, you should resolve this by either:

- Mapping the parameter to the same attribute in each of the match processes.
 - Or creating a new parameter and mapping this to one of the attributes.
-

If you do not resolve this, the migration would succeed, but in your upgraded system the parameter would not be populated.

Example Three: Let's say that you see an error similar to this one in the premigration report:

```
No metadata for source XXX guid b491939a-e6fd-33d5-9752-241cbe26a378 (Code: 300,551)
```

```
Error executing script: java.lang.IllegalArgumentException
```

In older releases of EDQ, it was possible to import a Case Management configuration file (a .dxc file) that did not contain some historical structures, and this error indicates that this has been done in your old EDQ environment. If a Case Management configuration file that includes all historical case source structures in it, then importing this will solve the problem. If such a file is not available, contact Oracle Support before proceeding with your upgrade.

3. Upgrade the configuration schema.

- h. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.

Note: If you are migrating from EDQ release 8.1 or below, then you will need to migrate the EDQ configuration schema in two steps:

- Step 1: migrate the configuration schema to release 9.0.
 - Step 2: migrate the configuration schema to your target 12c release.
-

- i. **Only complete this step if you are migrating from EDQ release 8.1 or below** (see the note directly above):

- i. Acquire a distribution of EDQ 9.0. (you will not need to fully install EDQ 9.0, but you will need access to its migration scripts.) The easiest way to acquire EDQ 9.0 is to download and then unzip the Windows Quick-Start installer for EDQ 9.0.
- ii. Upgrade the configuration schema to release 9.0 by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with the EDQ 9.0 distribution. This should be in `~edq-9_0_x(XXX)\9.0.x.(XXX)` (e.g. `edq-9_0_8_(1103)\9.0.8.(1103)`).
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.

- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to EDQ 9.0 in a Windows environment is:

```
java -jar C:\Users\Fred\Downloads\edq-9_0_8_(1103)\9.0.8.(1103)\migration.jar migrate postgresql:mydatabase@localhost:5432/director/director
```

- j. Upgrade the configuration schema to your EDQ 12c release by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ 12c environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `postgresql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to your new EDQ 12c release in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar migrate postgresql:mydatabase@localhost:5432/director/director
```

4. Upgrade the Case Management schema.

- k. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
- l. Upgrade the Case Management schema by issuing the following command:

```
java -jar path_to_migration.jar cm -db dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the Case Management schema in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar cm -db
pgsql:mydatabase@localhost:5432/director/director
```

5. If your environment's current Tomcat application server does not support EDQ 12.1.3, install a release of the Tomcat Application server that does support EDQ 12.1.3.
6. Configure the Tomcat Application Server.
 - a. Create the necessary Oracle Application Development Framework (ADF) shared library:
 - i. Create a **lib.adf** ADF shared directory in your Tomcat directory. If, for example, Tomcat is installed at C:\Program Files\Apache Software Foundation\Tomcat 8.0, create a folder called **lib.adf** within that folder.
 - ii. Ensure that the operating system user who will run the EDQ application server has full permissions to the **lib.adf** folder.
 - iii. Unzip the **adf-essentials.zip** file, delivered with the EDQ product, into the **lib.adf** directory. The JAR files in this zip file must be in the **lib.adf** directory, not the **adf-essentials** directory that it unzips to by default. You will find the **adf-essentials.zip** file in your EDQ installation folder's **edq\adf-essentials** sub-folder.
 - iv. Configure the Tomcat `common.loader` property (typically located in the `catalina.properties` file within Tomcat's conf directory) to load Oracle ADF from their respective shared library paths. For example:

```
common.loader=${catalina.base}/lib,${catalina.base}/lib/*.jar,${
catalina.home}/lib,${catalina.home}/lib/*.jar,${catalina.home}/l
ib.adf/*.jar
```
7. Manually copy some of the configuration folders and files from your old EDQ environment to your new EDQ environment.

*The location of your old EDQ environment's local configuration folder will depend upon the release of EDQ that you are migrating from, which operating system it is installed on, and how it was installed. Note that EDQ release 9.x and lower only had a single configuration directory, which was usually called **config** (this is the folder you should copy from); EDQ release 11.x and above has two configuration directories – one 'base' and one 'local'. If upgrading from EDQ release 11.x or above, copy from the 'local' configuration directory.*

- b. Copy the entire contents of the following folders, including any sub-folders, from your old EDQ environment's local configuration folder (in EDQ releases prior to 11, there was only a single config folder, and you should copy from this) to your new EDQ environment's local configuration folder that you created earlier in this process (in the example used in this document, this was called, oedq.local.home) merging them with any existing folders that have the same name:
- i. **autorun**
 - ii. **av**
 - iii. **blueprints**
 - iv. **buckets**
 - v. **businessrules**
 - vi. **capscan**
 - vii. **commandarea**
 - viii. **dbconnectors**
 - ix. **edqp**
 - x. **extras**
 - xi. **housekeeping**
 - xii. **landingarea**
 - xiii. **localselection**
 - xiv. **localwidgets**
 - xv. **match**
 - xvi. **properties**
 - xvii. **qas**
 - xviii. **resources**
 - xix. **runprofiles**
 - xx. **security**
 - xxi. **webservices**

Note that, depending upon the EDQ version that you are migrating from, and how it has been configured, some of the folders listed above may not exist.

- c. Treat the **casemanagement** folder as follows:
 - i. Copy entire contents of the **casemanagement** folder, **except** for the **indexes** folder and the **flags.xml** file.
 - ii. If you **have not** made any changes to the **flags.xml** file, copy that file too.
 - iii. If you **have** made changes to the **flags.xml** file, manually copy any new extended attributes that you added, and paste them into the version of **flags.xml** that you will find in your new EDQ environment's **casemanagement** folder. However, ensure that you do not remove or modify any of the default extended attributes that you will find within your new EDQ environment's **casemanagement** folder.
- d. Treat the **localgadgets** folder as follows:
 - i. If you are migrating from EDQ release 8.1.x or above, copy entire contents of the **localgadgets** folder.
 - ii. If you are migrating from EDQ release 8.0.x or below, then copy the entire contents of the **localgadgets** folder except for any selection functions. You will need to create new versions of any tailored selection functions and place them in the **localselection** sub-folder of your new EDQ environment's local configuration folder. Existing custom selection functions will not work after upgrade.
- e. Treat the **notification** folder as follows:
 - i. If you are migrating from EDQ release 9.0.x or above, copy the entire contents of the **notification** folder.
 - ii. If you are migrating from EDQ release 8.1.x or below, copy the entire contents of the **notification** folder except for the **jobs** and **eventlog** sub-folders.

*The **jobs** and **eventlog** folders cannot be copied from EDQ release 8.1.x or lower because job template code was restructured in EDQ 9.0 and therefore modified versions of the job templates created prior to release 9.0 will not work. (If you do erroneously copy these folders, you will find that job notification e-mails will not work, and that the job notifications will not be visible in the event log.) If the contents of these folders were not modified in your old EDQ environment, then you need do nothing: the default files provided with your new EDQ environment will work out-of-the-box. But if you have carried out any modifications, then any changes you made will need to be re-applied to the new versions of the templates - copying over the old versions will not work.*

- e. Treat the **triggers** folder as follows:
 - ix. Check through each of the files in the triggers directory, and in any sub-directories. If you find the following line in any file:

```
importClass("com.datanomic.director.guihelper.users.server.UserCache");
```

Change it to:

```
importClass("com.datanomic.userauth.server.cache.UserCache");
```

- x. Once you have made any necessary changes, copy the entire contents of the **triggers** folder, including any sub-folders, from your old EDQ environment's configuration folder to your new EDQ environment's local configuration folder, merging it with any existing folder that has the same name.
- f. Copy the following files from the root of your old EDQ environment's local configuration folder to the root of your new EDQ environment's local configuration folder:

- x*i. kfile**

- x*ii. override.properties**

8. Deploy the EDQ application on the Tomcat application server.

- b. You must manually deploy the EDQ application on Tomcat:

- i. Stop the Tomcat application server.
 - ii. Deploy the **edq.war** file on your application server. You will find the edq.war file in your EDQ installation folder's **edqoracle.edq** sub-folder.

For more information, see the Tomcat Web Application Deployment web page at <http://tomcat.apache.org/tomcat-8.0-doc/deployer-howto.html>

- c. Link EDQ to the base and local configuration directories that you created in one of the following:

- i. On any of the EDQ-supported platforms, you can use a Java property named -Dedq.config.path that specifies the paths to those directories. The syntax is as follows, where `path_to_base_config` is the path to the base configuration directory and `path_to_local_config` is the path to the local configuration directory.

Linux and UNIX (colon separates the paths):

```
-Dedq.config.path=path_to_base_config:path_to_local_config -  
Doracle.mds.cache=simple
```

Windows (semi-colon separates the paths):

```
-Dedq.config.path=path_to_base_config;path_to_local_config -  
Doracle.mds.cache=simple
```

- ii. In Windows environments only, you can append an environment variable named **EDQ_CONFIG_PATH** to your **JAVA_OPTS** environment variable, as shown in the following, where `path_to_base_config` is the path to the base configuration directory and `path_to_local_config` is the path to the local configuration directory.

```
EDQ_CONFIG_PATH="path_to_base_config;path_to_local_config"
```

For more information about setting Java parameters, see the Apache Tomcat Documentation website at <http://tomcat.apache.org/>

- d. Restart your application server and ensure that edq.war is successfully deployed.

Note that in some cases, the application server may report errors relating to Case Management indexes, such as this one: `SEVERE: 17-Sep-2015 11:57:32: [EDQ-01032] CaseManagement: Indexing thread failed org.apache.lucene.store.AlreadyClosedException: this IndexWriter is closed`

EDQ Errors relating to Case Management indexes should not prevent the application server from restarting successfully, but they are an indication that you may need to regenerate the Case Management indexes. If your application server reports such errors, follow this procedure to regenerate Case Management indexes:

- Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is **C:\Program Files\Java\jdk1.7.0_79\bin**).
 - Run the jconsole application (how you run this will depend upon your operating system – you may simply be able to double-click the application).
 - The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Remote Process** radio button. Enter a connection string for remote access to your Tomcat application server (`hostname:JMX_PORT_NO`), and enter your EDQ environment administration user's username (for example, `dnadmin`) and password. Then click **Connect**.
 - Navigate to the **Mbeans** tab.
 - Within the tree, on the left-side of the Window, expand **edq** - > **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.
 - Ensure that **clear** is set to **true**.
-

Click **reindex**. An Info dialog box will report that the Method was successfully invoked.

9. Open a Web Browser, and navigate to your new EDQ environment's Launchpad (unless you changed any of the default options when installing Tomcat, its address will be port numbers when installing Tomcat its URL will be <http://your-server-name:8080/edq>).
10. From the launchpad, open the Director user interface and log in.
11. Verify that all your projects have been successfully migrated.

Note that, if any of your processes use match review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

Note that in some cases a job or process may fail with one of the following error messages:

```
Cannot load data for replacements list reference data "X". Duplicate key "Y".
```

```
Cannot load data for replacements list supplied reference data. Duplicate key "{0}".
```

Prior to release 11, EDQ was tolerant of reference data with duplicate lookup keys. EDQ 11 and higher is less tolerant of Reference Data with duplicate lookup keys. The errors above indicate that your migrated environment includes reference data that has duplicate lookup keys.

In this situation, Oracle recommends that you should deduplicate the reference data in question (on a case-insensitive basis). You can do this either manually if the volume of reference data is small, or by creating an EDQ process and using the Group and Merge processor if the volume of reference data is large.

Alternatively - although it is not recommended - if previous behavior is required, you can add the following property to your new EDQ environment's **director.properties** file:

```
replace.transform.allow.duplicate.key = true
```

Note that the **director.properties** file is located in your new EDQ environment's local configuration directory (by default, this will be called **oedq.local.home**).

12. If you are using Case Management, then, from the Launchpad, open the Case Management user interface, log in and verify that all of your cases and alerts have been successfully migrated.

If Case Management appears to be missing cases and alerts (for example, if a filter that should return several cases and alerts in fact returns none), it is possible that the case management indexes need to be re-generated. Follow this procedure to re-generate Case Management indexes:

- Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is **C:\Program Files\Java\jdk1.7.0_79\bin**).
 - Run the jconsole application (how you run this will depend upon your operating system – you may simply be able to double-click the application).
-

-
- The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Remote Process** radio button. Enter a connection string for remote access to your Tomcat application server (`hostname:JMX_PORT_NO`), and enter your EDQ environment administration user's username (for example, `dnadmin`) and password. Then click **Connect**.
An example of a connection string is: `localhost:8090`.
 - Navigate to the **Mbeans** tab.
 - Within the tree, on the left-side of the Window, expand **edq** -> **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.
 - Ensure that **clear** is set to **true**.
 - Click **reindex**. An Info dialog box will report that the Method was successfully invoked.
-

13. Within Case Management, verify that you can successfully search for and find Cases and Alerts.

If you cannot successfully search for and find Cases and Alerts, this is likely to indicate that your `flags.xml` file is incorrect. The `flags.xml` file is located in your new EDQ environment's `casemanagement` directory, which is itself within the local configuration directory (called **oedq.local.home** by default). `flags.xml` contains information about extended attributes. A fresh EDQ installation comes with a `flags.xml` file that includes a number of default extended attributes. You can add new extended attributes to `flags.xml`, but if you removed or changed the default extended attributes that came with the original file, then EDQ will not be able to properly index Cases and Alerts, and you will therefore not be able to successfully search for them. You should resolve this by putting the default extended attributes back into `flags.xml`, and reindexing cases and alerts. (How to reindex is explained in the note above this one.). One way to do this is to copy the default extended attributes from a fresh version of `flags.xml`. You will find a fresh version of `flags.xml` in your new EDQ environment's base configuration folder, which is called **oedq.home** by default.

Upgrade Path 5:

From any Release of EDQ on an Oracle / Tomcat Platform to EDQ 12.1.3 Using an Oracle / Tomcat Platform

Overview

Upgrading from an installation of EDQ that uses the Oracle database and the Tomcat application server as its platform to EDQ 12.1.3 using the Oracle Database and the Tomcat application server is not recommended, but is possible. It is essentially a two-part process:

- Part 1 – Perform a Partial Installation of EDQ 12.1.3
- Part 2 – Upgrade Your Old EDQ Environment.

Note that the recommended path is to upgrade to EDQ 12.1.3 using the Oracle Database and the WebLogic application server as its platform

Part 1: Perform a Partial Installation of EDQ 12.1.3

Note that when installing EDQ it is often possible to complete the same task in different ways. For example, when you need to manipulate settings in the Oracle Database, you can usually do this through the Enterprise Manager user interface, or by issuing a SQL statement at the command line. In the instructions below, an example of one possible method of completing a task is often given. However, so long as the task is completed, you should feel free to use the approach that you are most comfortable with.

8. Ensure that your environment satisfies all EDQ 12.1.3 System Requirements, as outlined in section 1.4 of the Installing and Configuring Oracle Enterprise Data Quality guide, and upgrade your environment if necessary.
9. Download EDQ 12.1.3, as described in section 1.5 of the Installing and Configuring Oracle Enterprise Data Quality guide.
10. If your environment's current Java Development Kit does not support EDQ 12.1.3, install a Java Development Kit to support EDQ 12.1.3, as described in section 2.2 of the Installing and Configuring Oracle Enterprise Data Quality guide.
11. Install Enterprise Data Quality 12.1.3 assets. This step entails installing the Oracle Fusion Middleware EDQ distribution. This is usually downloaded as a zip file that contains a single file with a name similar to **fmw_12.1.3.x.x_edq.jar**. Running this jar file invokes the Oracle Universal Installer.
 - j. Open a command line tool and navigate to your Java Development Kit's bin folder.
 - k. Open the Oracle Universal Installer to install Enterprise Data Quality assets by issuing the following command:

```
java -jar EDQ_installation_jar_file.jar
```

Where:

- `EDQ_installation_jar_file.jar` is the path to the Enterprise Data Quality Installation file.

An example of a command to Open the Oracle Universal Installer to install Enterprise Data Quality in a Windows environment is:

```
java -jar C:\Users\peter\Downloads\fmw_12.1.3_edq.jar
```

After some time, the Oracle Enterprise Data Quality 12c Installation window will open at the Welcome screen.

- l. At the Welcome screen click **Next >**.
- m. At the Auto Updates screen, make a choice about whether or not to auto-update (and if so how), and click **Next >**.
- n. At the Installation Location screen, select a directory where you would like to install EDQ, and then click **Next >**. For the purpose of this document, we will refer to this directory as the EDQ installation folder.

*Note that, in order to install EDQ for the Tomcat application server, the EDQ installation folder **must not be** located within any existing Fusion Middleware home folder (if your system has one).*

- o. At the Installation Type screen, select **Installation for Other Platforms**, and click **Next >**.
 - p. At the Prerequisite Checks screen, wait until all of the prerequisite are passed (if any fail, address the issues that caused them to fail, and try this step again until all prerequisite checks are passed), and then click **Next >**.
 - q. At the Installation Summary screen, click **Install**.
 - r. At the Installation Progress screen, wait until the installation has completed, and then click **Finish**.
You should find that your EDQ installation folder has been populated with a number of sub-folders.
12. If your environment's current Oracle Database does not support EDQ 12.1.3, upgrade the Oracle Database to a release that will support EDQ 12.1.3.
 13. Create two new Oracle database accounts named **edqconfig** and **edqresults** or other names of your choosing. When selecting names, consider that one schema will be the configuration schema and the other will be the results schema (the names you choose must obviously be different to those used in your old EDQ environment).
 14. Grant the following privileges to each of these accounts:
 - **CREATE SESSION** (normally granted by CONNECT)
 - **CREATE INDEXTYPE** (normally granted by RESOURCE)
 - **CREATE SEQUENCE** (normally granted by RESOURCE)
 - **CREATE TABLE** (normally granted by RESOURCE)

- **CREATE TYPE** (normally granted by RESOURCE)
- **CREATE INDEX**
- **DROP TABLE**
- **DROP INDEX**
- **UNLIMITED TABLESPACE** (recommended to allow the auto-extension of tablespace)

15. Create your new EDQ environment's configuration directories.

- a. Create two empty directories to contain your new EDQ environment's configuration files. For the purposes of this document we will assume that you call these **oedq.home** and **oedq.local.home** respectively, though they do not necessarily have to be given those names.

For example:

On Linux and UNIX operating systems:

```
/opt/edqconfig/oedq.home
```

```
/opt/edqconfig/oedq.local.home
```

On the Windows operating system:

```
C:\edqconfig\oedq.home
```

```
C:\edqconfig\oedq.local.home
```

The first directory (**oedq.home**) is the base configuration directory and will contain the configuration files that should not be changed post-installation. The second directory (**oedq.local.home**) is the local configuration directory and will contain any custom settings that you create.

- b. Ensure that your application server user has read and write access to the two directories you create.

16. Run the EDQ Configapp to populate the configuration directories and the EDQ repository.

- e. Make certain that your Oracle database is running and open.
- f. Open a command line tool and navigate to your Java Development Kit's bin folder.
- g. Start the Configapp by issuing the following command:

```
java -jar path_to_configapp.jar
```

Where:

- `path_to_configapp.jar` is the path to the Configapp.

Note that the configapp is located in your EDQ installation folder's **edq/oracle.edq** directory.

An example of a command to start the Configapp in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\configapp.jar
```

The Configapp will open.

- h. Click **Begin**.
- f. At the Configuration directory screen, leave **Create a new configuration directory** selected, and select the **oedq.home** directory that you in the previous step as the Base Configuration directory location, and select the **oedq.local.home** directory that you created above as the Local Configuration directory location. Then click **Next >**.
- g. At the Functional Pack screen, select the functional packs you want and are licensed to use, and then click **Next >** (ensure that you scroll right to the bottom of this screen to see all of the functional packs).

All Functional Packs are needed if you want to use the full capabilities of the EDQ Customer Data Services Pack on your EDQ server, such as for integration with Siebel Customer Relationship Management or Universal Customer Master.

All Functional Packs are needed if you want to install Oracle Watchlist Screening on your EDQ server.

- h. At the Configure Config repository database screen, leave the **Selection Type** set to **Properties**. Enter the user name and password for your new EDQ environment's configuration schema (which you created in a previous step). Select the option for the type of database (Oracle). Enter the host name of the system that contains the database, its port number, and the database unique id. (For Oracle the unique id is the SID not the service name. If you want to use a pluggable database, which requires a service name, you must use a raw JDBC URL or JNDI source.). Ensure that you configured the connection to the database correctly by clicking **Test**. Then click **Next >** to continue.
- i. At the Configure Results repository database screen, leave the **Selection Type** set to **Properties**. Enter the user name and password for your new EDQ environment's Results schema (which you created in a previous step). Select the option for the type of database (Oracle). Enter the host name of the system that contains the database, its port number, and the database unique id. (For Oracle the unique id is the SID not the service name. If you want to use a pluggable database, which requires a service name, you must use a raw JDBC URL or JNDI source.). Ensure that you configured the connection to the database correctly by clicking **Test**. Then click **Next >** to continue.
- i. At the Selected options screen, Review the summary of your selections. Use **Back** to modify any of your selections, or click **Finish** to complete the configuration and continue.

In order for EDQ to function efficiently and in order for it to be able to process large volumes of data, sufficient storage space must be allocated to the tablespaces for the following Oracle database schemas:

- *EDQCONFIG*
 - *EDQRESULTS*
-

For production purposes, Oracle recommends the following minimum tablespace configuration, which in some cases may need to be extended:

- 30 GB total for the EDQCONFIG, held in a single file with autoextend turned off.
 - 120 GB total for the EDQRESULTS, made up of four 30GB files with autoextend turned off.
-

Part 2: Upgrade your Old EDQ Environment

2. Back up your old EDQ environment before you make any changes to it.
 - e. Before you back up your old EDQ environment you need to shut down EDQ's Tomcat application server. How you do this will vary by operating system.
 - iii. In Windows there is usually a Windows service for the application server. You can stop this via the **Control Panel > Administrative Tools > Services**.
-

If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the Windows service will be called **Datanomic Application Server**.

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the Windows service will be called **Oracle EDQ Application Server**.

- iv. In Linux or Unix environments, it is likely that the application server is started via init.d services (it's quite likely you will find them in **/etc/rc.d/init.d**). The service name and method of stopping it will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name stop** from a terminal window will often work.
 - f. Take a copy of your old EDQ environment's configuration folder, which is likely to be called either **config** or **oedq_local_home**.
 - v. For Windows environments, see the note below.
-

If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the configuration folder will be called **config**, and its default location is **C:\Program Files\Datanomic\dnDirector\config**.

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the configuration folder will be called **oedq_local_home**. Its location will depend upon the version of Windows you have installed on. On a Windows 7 machine, for example, the default location of **oedq_local_home** is:
C:\ProgramData\Oracle\Enterprise Data Quality\oedq_local_home.

- vi. In Linux or Unix environments, there is no default location or name for the EDQ configuration folder, and you may need to ask your system administrator to tell you what the path is. Having said that, by convention, the folder is often called either **config** or **oedq.local.home** or **oedq_local_home**.
- g. Backup the two EDQ Repository schemas in your Oracle database. The names of these schemas will have been decided at installation time, and can vary. You can find their connection strings (which will include their names) in the **director.properties** file. The properties you need to look at are called **dataSource.username** and **resultsDataSource.username** respectively.

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the default location of the **director.properties** file is **C:\Program Files\Datanomic\dnDirector\config**.*

*If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the **director.properties** file will be within a folder called **oedq_local_home**. Its location will depend upon the version of Windows you have installed on. On a Windows 7 machine, for example, the default location of **oedq_local_home** is: **C:\ProgramData\Oracle\Enterprise Data Quality\oedq_local_home**.*

- h. **Do not** restart your old environment's application server.
3. Run and examine the premigration report. This will reveal any problems with the Case Management schema (which is, strictly speaking, not a schema at all, but just a set of tables within EDQ's config schema). If any problems are revealed, you should fix them before proceeding to step 3.
- m. Open a command line tool:
 - i. In a Windows environment, open a Windows Command Prompt, running it as administrator (for example, right-click Command Prompt in the All Programs > Accessories group, and select **Run as administrator**).
 - ii. In a Unix or Linux environment, open a terminal window.
 - n. Change directory to your Java Development Kit's bin folder. You can do this by issuing the command `cd path_to_the_jdk_bin_folder` (for example, in a Windows environment you might enter a command similar to: `cd C:\Program Files\Java\jdk1.8.0_40\bin`).
 - o. Run the pre-migration report by issuing the following command:

```
java -jar path_to_migration.jar premigrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.

- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to run the premigration report in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar
preigrate oracle:mydatabase@localhost:1521/director/director
```

- p. If the Premigration report reveals any problems, fix them, and run the report again before proceeding further. (If the premigration report returns no output, by the way, that's a good thing: it means no problems were found.)

Examples of Problems Revealed by the Premigration Report, With Suggested Fixes

Example One: Let's say that you see the following error in the premigration report:

```
Parameter x in Source y for Case Type z using Selection Function aa "
" is using an attribute under input 'bb that is not mapped to case management.
This input will not be added automatically mapped and must be manually
configured."
```

In some earlier releases of EDQ it was possible for the value of a workflow parameter to be taken from a process attribute via a selection function even though the process attribute was not mapped to a case source attribute. For example, a workflow parameter called 'tel number' could be populated with the value of either 'cell-phone number' or 'landline number' via a First Non-empty Value selection function, even though neither 'cell-phone number' nor 'landline number' were mapped to an attribute in the case source. This is not permitted in later releases of EDQ. In order to fix this problem, you would need to create two new case source attributes and map 'cell-phone number' and 'landline number' to them. If you didn't do this, the migration would succeed, but in your upgraded system the 'tel number' parameter would not be populated, which may affect the correct workings of Case Management.

Example Two: Let's say that you see an error similar to this one in the premigration report:

Parameter X in Source Y for Case Type Z is defined differently across multiple match processes. Please make these configurations consistent.

In later releases of EDQ, a parameter within a case source can only be mapped to a single attribute, but this was not the case in older releases. This error message is reporting that the same case source is used in two (or more) match processes, and that the same parameter has been mapped to more than one attribute in these match processes. Before migrating the data, you should resolve this by either:

- Mapping the parameter to the same attribute in each of the match processes.
- Or creating a new parameter and mapping this to one of the attributes.

If you do not resolve this, the migration would succeed, but in your upgraded system the parameter would not be populated.

Example Three: Let's say that you see an error similar to this one in the premigration report:

No metadata for source XXX guid b491939a-e6fd-33d5-9752-241cbe26a378 (Code: 300,551)

Error executing script: java.lang.IllegalArgumentException

In older releases of EDQ, it was possible to import a Case Management configuration file (a .dxc file) that did not contain some historical structures, and this error indicates that this has been done in your old EDQ environment. If a Case Management configuration file that includes all historical case source structures in it, then importing this will solve the problem. If such a file is not available, contact Oracle Support before proceeding with your upgrade.

4. Upgrade the configuration schema.

- q. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.

Note: If you are migrating from EDQ release 8.1 or below, then you will need to migrate the EDQ configuration schema in two steps:

- Step 1: migrate the configuration schema to release 9.0.
- Step 2: migrate the configuration schema to your target 12c release.

-
- r. **Only complete this step if you are migrating from EDQ release 8.1 or below** (see the note directly above):

- i. Acquire a distribution of EDQ 9.0. (you will not need to fully install EDQ 9.0, but you will need access to its migration scripts.) The easiest way to acquire EDQ 9.0 is to download and then unzip the Windows Quick-Start installer for EDQ 9.0.
- ii. Upgrade the configuration schema to release 9.0 by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with the EDQ 9.0 distribution. This should be in **~edq-9_0_x(xxxx)\9.0.x.(xxxx)** (e.g. `edq-9_0_8_(1103)\9.0.8.(1103)`).
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to EDQ 9.0 in a Windows environment is:

```
java -jar C:\Users\Fred\Downloads\edq-9_0_8_(1103)\9.0.8.(1103)\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

- s. Upgrade the configuration schema to your EDQ 12c release by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ 12c environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to your new EDQ 12c release in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

5. Upgrade the Case Management schema.

- t. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
- u. Upgrade the Case Management schema by issuing the following command:

```
java -jar path_to_migration.jar cm -db dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the Case Management schema in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar cm -db oracle:mydatabase@localhost:1521/director/director
```

6. Migrate the upgraded configuration and Case Management data from your old EDQ environment's config repository to your new EDQ environment's repository.

- g. Use a standard Oracle tool, such as EXPDP, to take export your old EDQ repository's configuration schema.

Here is an example of a command that will export your old EDQ repository's configuration schema:

```
expdp config8/config8@localhost:1521/EDQREPOSITORY schemas=config8 DIRECTORY=DUMP_FILES DUMPFILE=config8-schema.dmp LOGFILE=config8_2015-09-16.log
```

In this example:

`config8` is your old EDQ Repository's configuration schema (the password is also set to `config8`).

`localhost` is the machine the Oracle Database is running on.

`1521` is the Oracle Database port number.

`EDQREPOSITORY` is the database SID.

`DUMP_FILES` is your data pump directory.

`config8-schema.dmp` is the name of the dump file you are going to export.

- h. Use a standard Oracle tool, such as IMPDP, to import the tables (note: only the tables) from your old EDQ repository's configuration schema into your new EDQ repository's configuration schema, replacing any tables with the same name that already exist.

Here is an example of a command that will import the tables from your old EDQ repository's configuration schema into your new EDQ repository's configuration schema, replacing any tables that already exist:

```
impdp config12/config12@localhost:1521/EDQREPOSITORY
DIRECTORY=DUMP_FILES DUMPFILE=config8-schema.dmp
INCLUDE=TABLE table_exists_action=replace SCHEMAS=config8
REMAP_SCHEMA=config8:config12
```

In this example:

`Config12` is your new EDQ Repository's configuration schema (the password is also set to `config12`).

`localhost` is the machine the Oracle Database is running on.

`1521` is the Oracle Database port number.

`EDQREPOSITORY` is the database SID.

`DUMP_FILES` is your data pump directory.

`config8-schema.dmp` is the name of the dump file you are going to import.

Note that the `expdp` and `impdp` examples above are just that – examples. You will need to adapt them to cater for the environment you are working in. At a minimum, you will also need to ensure that your database users have the appropriate privileges to run the `impdp` and `expdp` commands, and that your data pump directory is set. Consult Oracle Database documentation for more information about the `expdp` and `impdp` commands and how to use them.

7. If your environment's current Tomcat application server does not support EDQ 12.1.3, install a release of the Tomcat Application server that does support EDQ 12.1.3.

8. Configure the Tomcat Application Server.

- a. Create the necessary Oracle Application Development Framework (ADF) shared library:

- i. Create a **lib.adf** ADF shared directory in your Tomcat directory. If, for example, Tomcat is installed at C:\Program Files\Apache Software Foundation\Tomcat 8.0, create a folder called **lib.adf** within that folder.
- ii. Ensure that the operating system user who will run the EDQ application server has full permissions to the lib.adf folder.
- iii. Unzip the **adf-essentials.zip** file, delivered with the EDQ product, into the **lib.adf** directory. The JAR files in this zip file must be in the **lib.adf** directory, not the adf-essentials directory that it unzips to by default. You will find the **adf-essentials.zip** file in your EDQ installation folder's **edq\adf-essentials** sub-folder.
- iv. Configure the Tomcat common.loader property (typically located in the catalina.properties file within Tomcat's conf directory) to load Oracle ADF from their respective shared library paths. For example:

```
common.loader=${catalina.base}/lib,${catalina.base}/lib/*.jar,${catalina.home}/lib,${catalina.home}/lib/*.jar,${catalina.home}/lib.adf/*.jar
```

9. Manually copy some of the configuration folders and files from your old EDQ environment to your new EDQ environment.

*The location of your old EDQ environment's local configuration folder will depend upon the release of EDQ that you are migrating from, which operating system it is installed on, and how it was installed. Note that EDQ release 9.x and lower only had a single configuration directory, which was usually called **config** (this is the folder you should copy from); EDQ release 11.x and above has two configuration directories – one 'base' and one 'local'. If upgrading from EDQ release 11.x or above, copy from the 'local' configuration directory.*

- f. Copy the entire contents of the following folders, including any sub-folders, from your old EDQ environment's local configuration folder (in EDQ releases prior to 11, there was only a single config folder, and you should copy from this) to your new EDQ environment's local configuration folder that you created earlier in this process (in the example used in this document, this was called, oedq.local.home) merging them with any existing folders that have the same name:
 - i. **autorun**
 - ii. **av**
 - iii. **blueprints**
 - iv. **buckets**
 - v. **businessrules**
 - vi. **capscan**
 - vii. **commandarea**
 - viii. **dbconnectors**
 - ix. **edqp**

- x. **extras**
- xi. **housekeeping**
- xii. **landingarea**
- xiii. **localselection**
- xiv. **localwidgets**
- xv. **match**
- xvi. **properties**
- xvii. **qas**
- xviii. **resources**
- xix. **runprofiles**
- xx. **security**
- xxi. **webservices**

Note that, depending upon the EDQ version that you are migrating from, and how it has been configured, some of the folders listed above may not exist.

- g. Treat the **casemanagement** folder as follows:
 - i. Copy entire contents of the **casemanagement** folder, **except** for the **indexes** folder and the **flags.xml** file.
 - ii. If you **have not** made any changes to the **flags.xml** file, copy that file too.
 - iii. If you **have** made changes to the **flags.xml** file, manually copy any new extended attributes that you added, and paste them into the version of **flags.xml** that you will find in your new EDQ environment's **casemanagement** folder. However, ensure that you do not remove or modify any of the default extended attributes that you will find within your new EDQ environment's **casemanagement** folder.
- h. Treat the **localgadgets** folder as follows:
 - i. If you are migrating from EDQ release 8.1.x or above, copy entire contents of the **localgadgets** folder.
 - ii. If you are migrating from EDQ release 8.0.x or below, then copy the entire contents of the **localgadgets** folder except for any selection functions. You will need to create new versions of any tailored selection functions and place them in the **localselection** sub-folder of your new EDQ environment's local configuration folder. Existing custom selection functions will not work after upgrade.
- i. Treat the **notification** folder as follows:
 - i. If you are migrating from EDQ release 9.0.x or above, copy the entire contents of the **notification** folder.

- ii. If you are migrating from EDQ release 8.1.x or below, copy the entire contents of the **notification** folder except for the **jobs** and **eventlog** sub-folders.

*The **jobs** and **eventlog** folders cannot be copied from EDQ release 8.1.x or lower because job template code was restructured in EDQ 9.0 and therefore modified versions of the job templates created prior to release 9.0 will not work. (If you do erroneously copy these folders, you will find that job notification e-mails will not work, and that the job notifications will not be visible in the event log.) If the contents of these folders were not modified in your old EDQ environment, then you need do nothing: the default files provided with your new EDQ environment will work out-of-the-box. But if you have carried out any modifications, then any changes you made will need to be re-applied to the new versions of the templates - copying over the old versions will not work.*

- g. Treat the **triggers** folder as follows:

- xiii. Check through each of the files in the triggers directory, and in any sub-directories. If you find the following line in any file:

```
importClass("com.datanomic.director.guihelper.users.server.UserCache");
```

Change it to:

```
importClass("com.datanomic.userauth.server.cache.UserCache");
```

- xiv. Once you have made any necessary changes, copy the entire contents of the **triggers** folder, including any sub-folders, from your old EDQ environment's configuration folder to your new EDQ environment's local configuration folder, merging it with any existing folder that has the same name.

- h. Copy the following files from the root of your old EDQ environment's local configuration folder to the root of your new EDQ environment's local configuration folder:

- xv. kfile**

- xvi. override.properties**

- 10. Deploy the EDQ application on the Tomcat application server.

- a. You must manually deploy the EDQ application on Tomcat:

- i. Stop the Tomcat application server.
- ii. Deploy the **edq.war** file on your application server. You will find the edq.war file in your EDQ installation folder's **edq\oracle.edq** sub-folder.

For more information, see the Tomcat Web Application Deployment web page at <http://tomcat.apache.org/tomcat-8.0-doc/deployer-howto.html>

- b. Link EDQ to the base and local configuration directories that you created in one of the following:

- i. On any of the EDQ-supported platforms, you can use a Java property named - **Dedq.config.path** that specifies the paths to those directories. The syntax is as follows,

where `path_to_base_config` is the path to the base configuration directory and `path_to_local_config` is the path to the local configuration directory.

Linux and UNIX (colon separates the paths):

```
-Dedq.config.path=path_to_base_config:path_to_local_config -  
Doracle.mds.cache=simple
```

Windows (semi-colon separates the paths):

```
-Dedq.config.path=path_to_base_config;path_to_local_config -  
Doracle.mds.cache=simple
```

- ii. In Windows environments only, you can append an environment variable named **EDQ_CONFIG_PATH** to your **JAVA_OPTS** environment variable, as shown in the following, where `path_to_base_config` is the path to the base configuration directory and `path_to_local_config` is the path to the local configuration directory.

```
EDQ_CONFIG_PATH="path_to_base_config;path_to_local_config"
```

For more information about setting Java parameters, see the Apache Tomcat Documentation website at <http://tomcat.apache.org/>

- c. Restart your application server service and ensure that `edq.war` is successfully deployed.
11. Open a Web Browser, and navigate to your new EDQ environment's Launchpad (unless you changed any of the default options when installing Tomcat, its address will be port numbers when installing Tomcat its URL will be <http://your-server-name:8080/edq>).
12. From the launchpad, open the Director user interface and log in.
13. Verify that all your projects have been successfully migrated.
14. Run each of your jobs or standalone processes to regenerate their results (the results will not have been migrated).

Note that you should run jobs that require a run profile or run label from the Server Console user interface rather than from the Director user interface.

Note also that, if any of your processes use match review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

In some cases a job or process may fail with one of the following error messages:

```
Cannot load data for replacements list reference data "X". Duplicate key "Y".
```

Cannot load data for replacements list supplied reference data. Duplicate key "{0}" .

Prior to release 11, EDQ was tolerant of reference data with duplicate lookup keys. EDQ 11 and higher is less tolerant of Reference Data with duplicate lookup keys. The errors above indicate that your migrated environment includes reference data that has duplicate lookup keys.

In this situation, Oracle recommends that you should deduplicate the reference data in question (on a case-insensitive basis). You can do this either manually if the volume of reference data is small, or by creating an EDQ process and using the Group and Merge processor if the volume of reference data is large.

*Alternatively - although it is not recommended - if previous behavior is required, you can add the following property to your new EDQ environment's **director.properties** file:*

```
replace.transform.allow.duplicate.key = true
```

*Note that the **director.properties** file is located in your new EDQ environment's local configuration directory (by default, this will be called **oedq.local.home**).*

15. If you are using Case Management, then, from the Launchpad, open the Case Management user interface, log in and verify that all of your cases and alerts have been successfully migrated.
-

If Case Management appears to be missing cases and alerts (for example, if a filter that should return several cases and alerts in fact returns none), it is possible that the case management indexes need to be re-generated. Follow this procedure to re-generate Case Management indexes:

- *Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is **C:\Program Files\Java\jdk1.7.0_79\bin**).*
 - *Run the jconsole application (how you run this will depend upon your operating system – you may simply be able to double-click the application).*
 - *The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Remote Process** radio button. Enter a connection string for remote access to your Tomcat application server (*hostname:JMX_PORT_NO*), and enter your EDQ environment administration user's username (for example, *dnadmin*) and password. Then click **Connect**.*
 - *Navigate to the **Mbeans** tab.*
 - *Within the tree, on the left-side of the Window, expand **edq** - > **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.*
 - *Ensure that **clear** is set to **true**.*
-

-
- Click **reindex**. An Info dialog box will report that the Method was successfully invoked.
-

16. Within Case Management, verify that you can successfully search for and find Cases and Alerts.

*If you cannot successfully search for and find Cases and Alerts, this is likely to indicate that your flags.xml file is incorrect. The flags.xml file is located in your new EDQ environment's casemanagement directory, which is itself within the local configuration directory (called **oedq.local.home** by default). flags.xml contains information about extended attributes. A fresh EDQ installation comes with a flags.xml file that includes a number of default extended attributes. You can add new extended attributes to flags.xml, but if you removed or changed the default extended attributes that came with the original file, then EDQ will not be able to properly index Cases and Alerts, and you will therefore not be able to successfully search for them. You should resolve this by putting the default extended attributes back into flags.xml, and reindexing cases and alerts. (How to reindex is explained in the note above this one.). One way to do this is to copy the default extended attributes from a fresh version of flags.xml. You will find a fresh version of flags.xml in your new EDQ environment's base configuration folder, which is called **oedq.home** by default.*

Upgrade Path 6:

From EDQ 11g Using an Oracle / WebLogic Platform Installed Using the OUI to EDQ 12.2.1 Using an Oracle / WebLogic Platform

Overview

Upgrading from an installation of EDQ 11g that was installed using the OUI and that uses the Oracle database and the WebLogic application server as its platform to EDQ 12.2.1 using the Oracle Database and Oracle WebLogic as its platform is essentially a two-part process:

- Part 1 – Perform a Partial Installation of EDQ 12.2.1
- Part 2 – Upgrade Your Old EDQ Environment.

Part 1: Perform a Partial Installation of EDQ 12.2.1

1. Ensure that your environment satisfies all EDQ 12.2.1 System Requirements, as outlined in section 1.4 of the Installing and Configuring Oracle Enterprise Data Quality guide, and upgrade your environment if necessary.
2. Download EDQ 12.2.1, as described in section 1.5 of the Installing and Configuring Oracle Enterprise Data Quality guide.
3. Download the Fusion Middleware Infrastructure release 12.2.1.
4. If your environment's current Java Development Kit does not support EDQ 12.2.1, install a Java Development Kit to support EDQ 12.2.1, as described in section 2.2 of the Installing and Configuring Oracle Enterprise Data Quality guide.
5. If your environment's current Oracle Fusion Middleware Infrastructure does not support EDQ 12.2.1, install Oracle Fusion Middleware Infrastructure to support EDQ 12.2.1, as described in section 2.3.1 of the Installing and Configuring Oracle Enterprise Data Quality guide.

Note that the Oracle Fusion Middleware Infrastructure that you used to support your old release of EDQ will not support your new release of EDQ. Therefore, unless, you have already installed the new release of the Oracle Fusion Middleware Infrastructure for some other purpose, you will have to install it now. Note also that the new release of the Fusion Middleware Infrastructure will have to be installed in a different Fusion Middleware Home to your old release.

6. If your environment's current Oracle Database does not support EDQ 12.2.1, upgrade the Oracle Database to a release that will support EDQ 12.2.1.
 7. Install Enterprise Data Quality 12.2.1 assets, as described in section 3 (Installing Enterprise Data Quality) of the Installing and Configuring Oracle Enterprise Data Quality guide, ensuring that you install EDQ in your new Fusion Middleware Infrastructure's Fusion Middleware Home. For the avoidance of doubt, this step entails installing the Oracle Fusion Middleware EDQ distribution. This is usually downloaded as a zip file that contains a single file with a name similar to **fmw_12.2.1.x.x_edq.jar**. Running this jar file invokes the Oracle Universal Installer.
-

This step places elements of Enterprise Data Quality in your environment's Fusion Middleware Home location. Some of these will be used later in the installation process. Note, however, that you will not have a working EDQ environment at this stage.

Part 2: Upgrade your Old EDQ Environment

A note on Upgrading EDQ instances that are configured for high availability

Enterprise Data Quality 11g and higher can be enabled for high availability by configuring several WebLogic Managed Servers within the same WebLogic Cluster. In releases of EDQ 11, and also in EDQ 12.1.3, it was necessary for each of these Managed Servers to have its own its own dedicated database repository (in other words, its own pair of Config and Results datasources), and its own dedicated config directory.

In EDQ 12.2.1, configuration for High Availability was simplified. At release 12.2.1 and above, all of the WebLogic Managed Servers in a Cluster can share the same database repository (in other words, only one pair of Config and Results datasources is required). And all of the WebLogic Managed Servers in a Cluster can share the same base and local config directories.

Oracle recommends that when you upgrade an instance of EDQ 11g or 12.1.3 that is configured for high availability, you change its configuration to take advantage of the simplified approach to High Availability introduced in EDQ 12.2.1. To do this you should:

- *Upgrade only one database repository (in other words, only one pair of Config and Results schemas) using the Upgrade Assistant.*
-

-
- *Run the Oracle Fusion Middleware Reconfiguration Wizard a single time, supplying the connection details of the database repository that you upgraded. As you run the Wizard, at the Advanced Options screen, ensure that you select the **Managed Servers, Clusters and Coherence** option, and then, referring to the EDQ High Availability Guide for release 12.2.1, configure your upgraded EDQ instance for High Availability using the new, simplified architecture.*
 - *Ensure that all of your WebLogic Managed Server's Server Parameters (in other words, its 'startup arguments' or 'Java options') are configured to point towards the same base and local config directories.*
-

Although not recommended, it is also possible to maintain the existing high availability configuration after an upgrade, by continuing to use multiple schemas and config areas. To do this:

- *Run the Upgrade Assistant separately against each pair of Config and Results schemas.*
 - *Disable coherence by setting `coherence.clustering = false` in the **director.properties** file.*
-

1. Back up your old EDQ environment.
 - a. Before you begin, Oracle recommends that you make a complete backup of your old EDQ environment, including your EDQ domain. For instructions, see "Backup and Recovery Strategies for Upgrade" in the Planning an Upgrade of Oracle Fusion Middleware documentation.
2. Shutdown your old EDQ environment's Managed WebLogic Server.
3. Ensure that your old EDQ environment's WebLogic Server Administration Server remains running.
4. If your old EDQ environment used a File-Based Keystore, migrate it to an OPSS Security Store.

At release 12c, all Oracle Fusion Middleware products store their security credentials within an Oracle Platform Security Services (OPSS) schema within an Oracle database. The Oracle term for this is security store. If you were using a file-based keystore in the configuration directories, you must now use an OPSS DB-based security store.

To convert to an OPSS DB-based security store, you create the OPSS, Audit Services, and Audit Services Viewer schemas. The Audit Services and Audit Services Viewer schemas are required by the Oracle Platform Security Services schema. After creating these schemas, you then create a WebLogic Server data source for the OPSS schema, and then re-associate the EDQ keystore to the OPSS schema.

- a. Create the OPSS Schema
 - i. Download the 11g Repository Creation Utility (RCU) from Oracle. For instructions, see http://docs.oracle.com/cd/E23104_01/download_readme_ps6/download_readme_ps6.htm#BABHDBJC
 - ii. Extract the contents to a directory of your choice; this directory will be referred to as the RCU_HOME directory. On Windows operating systems, make sure that you do not unzip the RCU file to a directory name that contains spaces.
 - iii. Run the command shell or console of the operating system.

- iv. Start RCU from the bin directory inside the RCU_HOME directory.

Note:

If you are running RCU using a non-English database, set the following environment variables:

```
setenv LANG en_US.UTF8
```

```
setenv LC_ALL $LANG
```

```
setenv NLS_LANG american_america
```

On Linux:

```
./rcu
```

On Windows:

```
rcu.bat
```

- v. Start RCU from the bin directory inside the RCU_HOME directory.
- vi. Click **Next >** to pass through the Welcome screen.
- vii. On the Create Repository screen, select **Create**, then click **Next >**.
- viii. On the Database Connection Details screen, enter the connection details for your old environment's EDQ repository database and the login credentials of an EDQ repository database user who has DAB or SYSDBA privileges. Click **Next >**. The RCU will check prerequisites.
- ix. On the Select Components screen, ensure that you select the same schema prefix that is used for your old environment's EDQ Repository (for example, if the old environment's schemas started with **DEV**, then ensure you select **DEV**). This will usually entail checking the **Select existing prefix** option. Under Oracle AS Repository Components, expand **AS Common Schemas** and then select **Audit Services** and **Oracle Platform Security Services**. Click **Next >**.
- x. On the Schema Passwords screen, supply passwords for the schema owners. Click **Next >**.
- xi. On the Map Tablespaces screen, the default tablespace names are displayed. Either accept these, or select alternative tablespaces for the components. (Optionally, you can click **Manage Tablespaces** at this point to change the tablespaces sizes.) Click **Next >**.
- xii. On the Summary screen, click **Create** to complete the creation of the schema.

- b. Create a JDBC Data Source for the OPSS Schema.

Use the WebLogic Server Administration Console to create a generic JDBC data source for the OPSS schema. For assistance or more information on any screen, click **Help**.

- i. In the Domain Structure tree, expand **Services**, then select **Data Sources**.
- ii. On the Summary of Data Sources screen, click **New** and select **Generic Data Source**.
- iii. On the JDBC Data Sources Properties screen, enter the data source name and the JNDI name of the OPSS schema that you created with the 11g RCU in Task 1. The OPSS DataSource name is **opss-data-source** and the JNDI name is similar to this example: **jdbc/OpssDataSource**. Select the database type as **Oracle**, and then click **Next**.
- iv. On the Create a New JDBC Data Source screen, in the Database Driver pull-down, select **Oracle's Driver (Thin) for Instance connections; Versions:9.0.1 and later** (this is a non-XA JDBC driver) and then click **Next**.

Note: You must install JDBC drivers before you can use them to create database connections. Some JDBC drivers are installed with WebLogic Server, but many are not installed.

- v. On the Transaction Options screen, make sure that **Supports Global Transactions** is **deselected** and then click **Next**.
- vi. On the Connection Properties screen, enter the connection details and credentials to connect to the OPSS schema that you created with the 11g RCU in the previous step.
- vii. On the Test Database Connection screen, review the connection parameters and click **Test Configuration**. Results from the connection test are displayed at the top of the page. If the test is unsuccessful, correct any configuration errors and retry the test. If the JDBC driver you selected is not installed on the WebLogic Server Administration Server, click **Next** to skip this step. Install the driver after finishing this procedure.
- viii. On the Select Targets page, select the WebLogic Server(s) or cluster(s) on which you want to deploy the data source. Select both the EDQ Managed WebLogic Server and the WebLogic Server Administration Server as targets. If you are using WebLogic clustering, deploy the data source to the cluster.
- ix. Click **Finish** to save the JDBC data source configuration and deploy the data source to the targets that you selected.
- x. If your WebLogic Server is working in production mode, click **Activate Changes** in the Change Center of the Administration Console. Not all changes take effect immediately, but all will take effect when you restart the server at the end of the EDQ upgrade process.
- xi. Restart the WebLogic Server Administration Server. (If the console tells you that no restart is necessary, ignore this message and restart anyway – the next step will not work unless you restart the Administration Server).

- c. Reassociate the File-based Keystore to a DB-based Security Store

Use the WebLogic Server Scripting Tool (WLST) to reassociate the file-based keystore to the OPSS data source that you created with the WebLogic Server Administration Console.

- i. From the 11g \$EDQ_HOME/oracle_common/common/bin directory, issue the following command to run the WebLogic Server Scripting Tool (WLST):

UNIX/Linux:

```
./wlst.sh
```

Windows:

```
wlst.cmd
```

- ii. In WLST, connect to your WebLogic instance.

```
connect('username', 'password', 'host:port')
```

Where:

`username` is the name of the Oracle WebLogic Server admin user.

`password` is the Oracle WebLogic Server admin user's password.

`host` is the name of the machine that the WebLogic server is running on.

`port` is the connection port for the WebLogic Server Administration Server.

- iii. Reassociate the OPSS schema from the file-based format to the database format using the reassociate script.

```
reassociateSecurityStore(domain="upgradeDomain",  
servertype="DB_ORACLE", jpsroot="cn=jpsroot",  
datasourcename="name_of_datasource")
```

Where:

`upgradeDomain` is the name of the WebLogic domain used for your old instance of Enterprise Data Quality.

`name_of_datasource` is the name of the new OPSS JNDI data source that you created in the WebLogic Server Administration Console in the previous step.


For more information about the reassociate script, see

http://docs.oracle.com/cd/E29542_01/core.1111/e10043/cfgauthr.htm#JISEC2675

- iv. Once the reassociation has completed, restart the WebLogic Server Administration Server.

Do not restart the EDQ managed server(s) yet. Complete the rest of the upgrade steps. The changes you made are stored on the server and will propagate to the rest of the servers when you restart the servers after upgrade.

5. Create the Required 12c Schemas.



Oracle Fusion Middleware Release 12c introduces new schemas and tables that must be created with the Release 12.2.1 version of the Oracle Repository Creation Utility (RCU). This step of the upgrade must be performed before upgrading your old environment's EDQ Domain.

- a. To create the required 12.2.1 schemas, see Section 4.2, "Creating an EDQ Database Repository" of the Installing and Configuring Oracle Enterprise Data Quality guide for release 12.2.1. Follow those instructions, but with these four additions:
 - i. On the Database Connection Details screen, ensure that you supply the connection details for your old EDQ environment's database repository. This is because you are going to add new EDQ 12.2.1 schemas to your old repository.
 - ii. On the Select Components screen, ensure that you select the same schema prefix that is used for your old environment's EDQ Repository (for example, if the old environment's schemas started with **DEV**, then ensure you select **DEV**). This will usually entail checking the **Select existing prefix** option.
 - iii. On the Select Components screen, the **AS Common** Schemas option should be selected by default. Make only a single change on this screen: Expand **Oracle EDQ** if necessary, and select the **EDQ Staging Repository** schema.
 - iv. The effect of this should be that the RCU will create exactly two new schemas:
 - EDQ Staging Repository.
 - Common Infrastructure Services.

You should be able to confirm that these two new schemas (and their associated tablespaces) are going to be created when you get to the Map Tablespaces screen, and also on the subsequent Summary screen. Before you click Create on the Summary screen, make absolutely sure that these two schemas will be created, and also ensure that they have the same prefix as your existing EDQ Repository schemas.

6. Follow the instructions below to upgrade the schemas in the EDQ repository database.
 - a. Start the Upgrade Assistant:
 - i. Go to the following directory, where 12.2.1_FMW_HOME is the version 12.2.1 Oracle home directory.

On UNIX operating systems:

```
12.2.1_FMW_HOME/oracle_common/upgrade/bin
```

On Windows operating systems:

```
12.2.1_FMW_HOME\oracle_common\upgrade\bin
```

- ii. Run the following program:

On Linux or UNIX operating systems:

```
./ua
```

On Windows operating systems:

ua.bat

- b. At the Welcome screen, Review Important Reminders before Proceeding. The Welcome screen contains reminders to consider before proceeding with your upgrade. Make sure you read these and verify that you are ready to proceed. For assistance or more information on any screen, click Help. Otherwise, click **Next >**.

For more information about the following screens, see "Upgrading Schemas with the Upgrade Assistant" in Upgrading with the Upgrade Assistant.

- c. Select the type of upgrade you wish to perform.
 - i. Select **All Schemas Used By a Domain**.
 - ii. In the Domain Directory field, select your old EDQ environment's domain directory (this is likely to be similar to **/OLD_FMW_HOME/user_projects/domains/edq_domain**).
 - iii. Click **Next >**.

- d. The Component List screen will display a list of the components (or in other words, schemas) to be upgraded in your old EDQ environment's domain. The following components should always be listed:

Oracle Enterprise Data Quality Results

Oracle Audit Services

Oracle Platform Security Services

Oracle Enterprise Data Quality Configuration

Depending upon how your EDQ 11g environment was configured, it is possible that other components may also be listed.

- i. Click **Next >**.
- e. In the Prerequisites screen, verify that you have met the prerequisites, and, provided that you have, check the relevant boxes (note that The Upgrade Assistant will not verify that the prerequisites have been met.). Then click **Next >**.
- f. Specify Database and Schema Credentials
 - i. The EDQ Results schema, OPSS, and EDQ Configuration schema screens are displayed in succession.
 - ii. On the first screen, specify connection details for your old environment's EDQ Repository (note that you may have to add **as sysdba** to the DBA User Name), and then click **Connect**. Once you have clicked Connect, the Schema User Name and Schema Password fields should be populated automatically for the EDQ Results schema (if they are not, enter this information manually).
 - iii. The remaining screens are automatically populated with the database connection and schema credentials that you supplied on the EDQ Results Schema screen. If these

- entries are not correct for any schema, change the entries and ensure that a database connection is made.
 - g. Complete the upgrade validation.
 - i. On the Examine screen, the Upgrade Assistant performs a series of validations on the components. Validations for the Oracle Enterprise Data Quality Results component, the Oracle Audit Services component, and the Oracle Enterprise Data Quality Configuration component should succeed, and the Upgrade Assistant should report that an upgrade is not necessary for the Oracle Audit Services component. Click **Next** >.
 - h. Initiate the Upgrade.
 - i. Click **Upgrade** on the Upgrade Summary screen to initiate the upgrade. The Upgrade Progress screen shows information about the progress of the upgrade, and the Upgrade Success screen summarizes the upgrade.
7. Reconfigure the EDQ Domain.

In this step you will run the Oracle Fusion Middleware reconfiguration wizard to complete the upgrade of your WebLogic Server domain environment.

- a. Check that the Staging Mode is set to Nostage.
 - i. Log in to the WebLogic Server Administration Server console from any web browser.
 - ii. In the left pane, expand **Environment** and then click **Servers**. The Summary of Servers screen will be displayed.
 - iii. In the Summary of Servers screen, click your EDQ Managed WebLogic Server (for example: **edq_server1**).
 - iv. Select the **Configuration** tab, then the **Deployment** tab.
 - v. From the Staging Mode menu, ensure that **nostage** is selected, and, if it is not, select it. Do not change any other server settings.
 - vi. Click **Save**, and, if working in production mode, activate your settings.
- b. Start Node Manager if it is not already running

*Note that you only need to complete this step if the Node Manager is **not** already running.*

This task ensures that Node Manager has been started at least once since it was installed. Starting Node Manager creates a set of script files that must be moved from the old domain to the release 12.2.1 domain. Performing this task eliminates the need for extra steps when upgrading your domain.

- i. Using a command line tool (for example, the Windows Command prompt or a Linux or Unix terminal), navigate to the **bin** folder in your old EDQ environment's WebLogic base domain (this will be at a location similar to **FMW_HOME/user_projects/domains/edq_domain/bin**).
- ii. Run the following command to start Node Manager. For **hostname**, specify the name of the system where Node Manager is installed, and for **port**, specify the port number on which it runs.

Windows:

```
startNodeManager.cmd hostname port
```

UNIX:

```
./startNodeManager.sh hostname port
```

c. Stop Node Manager

- i. If the Node Manager was not already running, and you started it using the command above, then stop it by closing the command shell from which it is running.
- ii. If Node Manager was already running (for example, because it was already running as a service in your environment), then stop it using the appropriate means (for example, stop the Node Manager service).

d. Start the Reconfiguration Wizard

In this step you will launch the wizard that will upgrade the EDQ domain.

- i. Stop the WebLogic Server Administration Server and all managed servers.
- ii. Go to the following directory, where **12.2.1_FMW_HOME** is the release 12.2.1 Fusion Middleware home directory.

On Windows operating systems:

```
12.2.1_FMW_HOME\oracle_common\common\bin
```

On UNIX operating systems:

```
12.2.1_FMW_HOME/oracle_common/common/bin
```

- iii. Start the domain reconfiguration wizard.

On UNIX operating systems:

```
./reconfig.sh -log=log_file
```

On Windows operating systems:

```
reconfig.cmd -log=log_file
```

Specify the full path and file name in place of `log_file`. Creating a log file can be helpful if you need to troubleshoot the reconfiguration process.

Note that if the following error message is displayed, it indicates that the default cache directory is not valid:


```
*sys-package-mgr*: can't create package cache dir
```

You can change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`. For example:

```
CONFIG_JVM_ARGS=-Dpython.cachedir=valid_directory
```

e. Specify the domain.

- i. On the Select Domain screen, specify the full path to the location of your old EDQ domain (it is likely to be similar to `OLD_FMW_HOME/user_projects/domains/edq_domain` or



`OLD_FMW_HOME\user_projects\domains\edq_domain`). You can also click **Browse** and use the file manager window to help you select the domain location. (Be careful – you may find that the 12.2.1 Home is selected by default – remember to navigate to your old FMW home instead).

- f. View the Reconfiguration Setup Progress.
 - i. The Reconfiguration Setup Progress displays the reconfiguration progress and verifies whether the base domain that you selected can be reconfigured to the 12.2.1 domain. The message “Core WLS Infrastructure Reconfigured Successfully!” indicates that the domain can be reconfigured to the 12.2.1 domain, and you can click **Next** to go to the next step. If this message is not returned, the domain cannot be reconfigured to the 12.2.1 domain. If this happens, check to see if the EDQ version is earlier than version 11g; if so, then select the appropriate upgrade path for version 12.2.1, and start the entire upgrade process again.
- g. Select the Domain Mode and JDK.
 - i. It is not possible to change the Domain Mode from development to production or vice versa.
 - ii. On the Domain Mode and JDK screen, specify the location of the Java Development Kit (JDK) to use in the new EDQ domain. This must be a JDK that conforms to the system requirements of your new EDQ 12.2.1 environment.
- h. Check the JDBC Data Sources.
 - i. On the JDBC Data Sources screen, your old EDQ domain's Config and Results datasources should be listed in the lower half of the screen. If you need to make changes, select the check box next to the data source name and then make the changes.
- i. Test the JDBC Data Sources
 - i. On the JDBC Data Sources Test screen, test the data source connections that were detected. Select both schemas and then click **Test Selected Connections**.
- j. Select the Database Configuration Type.
 - i. In the Database Configuration Type screen, the **RCU Data** option should be selected by default. The connection details for your EDQ repository database should be partially populated, and the schema owner for the Common Infrastructure schema (STB) should be displayed. If this information is not all automatically populated, or is not correct, make any necessary entries or amendments. (In particular, make sure that the prefix of the schema owner is correct).
 - ii. Click **Get RCU Configuration**. The Reconfiguration Wizard should report that it can connect to the Database Server and that it can retrieve schema data and bind local schema components with the retrieved data. If these steps fail, amend the connection data and click **Get RCU Configuration** again. Otherwise, click **Next >**.
- k. Check the JDBC Component Schema.

- i. On the JDBC Component Schema screen, the schemas associated with your old EDQ domain are listed in the lower half of the screen. If you need to make changes, select the check box next to the data source name and then make the changes. Note that the connection details for some of the schemas may not be correct. If they are not, amend them so that they are correct.
- l. Test the JDBC Component Schema
 - i. On the JDBC Component Schema Test screen, test the data source connections that were detected. Select the schemas that you want to test and then click Test Selected Connections.

*To test a connection, the database to which you are connecting must be running. Note that the test for the OPSS Audit Viewer component schema may fail (all other tests should succeed). If the test for the OPSS Audit Viewer component schema fails, click **Next >** anyway, and click **OK** in the subsequent dialog box to proceed without verification.*

- m. Node Manager
 - i. Leave the default options for the Node Manager Type and Node Manager Configuration selected, and enter the Node Manager's Credentials.
- n. Select Optional Advanced Configuration Options

On the Advanced Configuration Screen, you can select additional domain options.

- i. Select **Managed Servers, Clusters and Coherence** and click **Next >**.
- o. Review Managed Servers.
 - i. On the Managed Servers screen, make no changes, and just click **Next >**.
- p. Review Clusters.
 - i. On the Clusters screen, make no changes (assuming that you do not want to change your cluster configuration), and just click **Next >**. (Note that this screen will be empty unless WebLogic clusters were configured in your old environment – leave it empty unless you want to use WebLogic clusters to configure EDQ for high availability).
- q. Review Coherence Clusters.
 - i. On the Coherence Clusters screen, make no changes (assuming that you do not want to change your Coherence cluster configuration), and just click **Next >**.
- r. Review Machines.
 - i. On the Machines screen, make no changes (assuming that you do not want to change your machine), and just click **Next >**.
- s. Assign Servers to Machines.
 - i. Assign the EDQ Managed WebLogic Server (probably called **edq_server1**) to your machine. Ensure that the AdminServer is assigned to the Machine too. Click **Next>**.

- t. Initiate the Domain Reconfiguration
 - i. On the Configuration Summary screen, review the configuration and then click **Reconfig** to start the reconfiguration process, or click **Back** to make changes.
 - u. Finish the Reconfiguration
 - i. Wait for the message that states, "Domain Reconfiguration Applied Successfully" and then click **Next**.
 - ii. A check mark and the message "Oracle WebLogic Server Reconfiguration Succeeded" indicate that the reconfiguration was successful. Click **Next >**, and then click **Finish** to dismiss the reconfiguration wizard.
8. Follow the instructions below to upgrade EDQ's domain configuration.
- a. Start the Upgrade Assistant:
 - i. Go to the following directory, where 12.2.1_FMW_HOME is the version 12.2.1 Oracle home directory.
 - On UNIX operating systems:
`12.2.1_FMW_HOME/oracle_common/upgrade/bin`
 - On Windows operating systems:
`12.2.1_FMW_HOME\oracle_common\upgrade\bin`
 - ii. Run the following program:
 - On Linux or UNIX operating systems:
`./ua`
 - On Windows operating systems:
`ua.bat`
 - b. At the Welcome screen, Review Important Reminders before Proceeding. The Welcome screen contains reminders to consider before proceeding with your upgrade. Make sure you read these and verify that you are ready to proceed. For assistance or more information on any screen, click Help. Otherwise, click **Next >**.

For more information about the following screens, see "Upgrading Oracle WebLogic Component Configurations" in Upgrading with the Upgrade Assistant.

- c. At the Selected Schemas screen, select the **All Configurations Used By a Domain** option. At this point the name of the screen will change to All Configurations. In the **Domain Directory** field, select your old EDQ environment's domain directory (this is likely to be similar to **/OLD_FMW_HOME/user_projects/domains/edq_domain**), and click **Next >**.

- d. At the Component List screen, review the list of components to be upgraded (it should consist of Oracle JRF, System Components Infrastructure, and Oracle Enterprise Data Quality Configuration), and click **Next >**.
 - e. At the Prerequisites screen, verify that you have met the prerequisites, and, provided that you have, check the relevant boxes (note that The Upgrade Assistant will not verify that the prerequisites have been met.). Then click **Next >**.
 - f. At the Examine screen, the Upgrade Assistant will validate that the components can be upgraded, returning statuses for either upgrade not necessary or succeeded for each of them. When the examination has finished, click **Next>**.
 - g. At the Upgrade Summary screen, click **Upgrade**.
 - h. At the Upgrade Progress screen, wait until the upgrade has finished, and then click **Next>**.
 - i. At the Upgrade Success screen, click **Close**.
9. Apply the Upgrade Changes to the Base Domain

These steps apply the changes that you made during the upgrade of the EDQ domain to the base domain.

- a. Start the WebLogic Node Manager for the upgraded EDQ domain.

*Take care that you start the right Node Manager, especially if your old (pre-upgrade) EDQ environment's Node Manager was started via a service: the service will almost certainly still be pointing at your old (not upgraded) Node Manager configuration. If you simply restart your service it will start your old environment's Node Manager, which will not work against your upgraded EDQ domain. A safe option is to start the upgraded Node Manager by running the startNodeManager script (startNodeManager.sh for Unix and Linux, or startNodeManager.cmd for Windows) that you will find in your EDQ domain's **bin** directory.*

- b. Start the WebLogic Server Administration Server for your upgraded EDQ domain. If it is running, stop and then restart it.

(UNIX) `DOMAIN_HOME/bin/startWebLogic.sh`

(Windows) `DOMAIN_HOME\bin\startWebLogic.cmd`

- c. Log in to the WebLogic Server Administration Server console from any web browser.
- d. In the Domain Structure list, click **Deployments**.
- e. Under Deployments, select **edq** and then click **Update**.
- f. Click **Next**.
- g. Click **Finish**.
- h. To complete the upgrade, start the EDQ managed server.

The startManagedWebLogic.sh script is located in your EDQ WebLogic domain's bin directory. Running this script is one method of starting the EDQ WebLogic Managed Server. Note that the startManagedWebLogic.sh script may be overwritten when the EDQ WebLogic Domain is reconfigured. Therefore, if you customized the script, any customizations may have been lost during the upgrade process. This is a particularly important point to note if you embedded EDQ's Server Parameters (in other words, its 'startup arguments' or 'Java options') within this script, as, without these, the EDQ Managed Server may not function correctly. See section 6 of the Installing and Configuring Oracle Enterprise Data Quality guide, "Setting Server Parameters to Support Enterprise Data Quality", for more information about EDQ's server parameters.

10. Open a Web Browser, and navigate to your upgraded EDQ environment's Launchpad (unless you changed any of the default options when installing EDQ, its address will be <http://your-server-name:8001/edq>).
11. From the launchpad, open the Director user interface and log in.
12. Verify that all your projects have been successfully migrated.
13. If any of your processes use Match Review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

Upgrade Path 7:

From EDQ 12.1.3 Using an Oracle / WebLogic Platform Installed Using the OUI to EDQ 12.2.1 Using an Oracle / WebLogic Platform

Overview

Upgrading from an installation of EDQ 12.1.3 that was installed using the OUI and that uses the Oracle database and the WebLogic application server as its platform to EDQ 12.2.1 using the Oracle Database and Oracle WebLogic as its platform is essentially a two-part process:

- Part 1 – Perform a Partial Installation of EDQ 12.2.1
- Part 2 – Upgrade Your Old EDQ Environment.

Part 1: Perform a Partial Installation of EDQ 12.2.1

1. Ensure that your environment satisfies all EDQ 12.2.1 System Requirements, as outlined in section 1.4 of the Installing and Configuring Oracle Enterprise Data Quality guide, and upgrade your environment if necessary.
2. Download EDQ 12.2.1, as described in section 1.5 of the Installing and Configuring Oracle Enterprise Data Quality guide.
3. Download the Fusion Middleware Infrastructure release 12.2.1.
4. If your environment's current Java Development Kit does not support EDQ 12.2.1, install a Java Development Kit to support EDQ 12.2.1, as described in section 2.2 of the Installing and Configuring Oracle Enterprise Data Quality guide.
5. If your environment's current Oracle Fusion Middleware Infrastructure does not support EDQ 12.2.1, install Oracle Fusion Middleware Infrastructure to support EDQ 12.2.1, as described in section 2.3.1 of the Installing and Configuring Oracle Enterprise Data Quality guide.

Note that the Oracle Fusion Middleware Infrastructure that you used to support your old release of EDQ will not support your new release of EDQ. Therefore, unless, you have already installed the new release of the Oracle Fusion Middleware Infrastructure for some other purpose, you will have to install it now. Note also that the new release of the Fusion Middleware Infrastructure will have to be installed in a different Fusion Middleware Home to your old release.

6. If your environment's current Oracle Database does not support EDQ 12.2.1, upgrade the Oracle Database to a release that will support EDQ 12.2.1.
7. Install Enterprise Data Quality 12.2.1 assets, as described in section 3 (Installing Enterprise Data Quality) of the Installing and Configuring Oracle Enterprise Data Quality guide, ensuring that you install EDQ in your new Fusion Middleware Infrastructure's Fusion Middleware Home. For the avoidance of doubt, this step entails installing the Oracle Fusion Middleware EDQ distribution. This is usually downloaded as a zip file that contains a single file with a name similar to **fmw_12.2.1.x.x_edq.jar**. Running this jar file invokes the Oracle Universal Installer. You do not need to configure or extend a WebLogic domain now, since you will upgrade the WebLogic domain using the reconfiguration wizard at a later stage.

This step places elements of Enterprise Data Quality in your environment's Fusion Middleware Home location. Some of these will be used later in the installation process. Note, however, that you will not have a working EDQ environment at this stage.

Part 2: Upgrade your Old EDQ Environment

A note on Upgrading EDQ instances that are configured for high availability

Enterprise Data Quality 11g and higher can be enabled for high availability by configuring several WebLogic Managed Servers within the same WebLogic Cluster. In releases of EDQ 11, and also in EDQ 12.1.3, it was necessary for each of these Managed Servers to have its own its own dedicated database repository (in other words, its own pair of Config and Results datasources), and its own dedicated config directory.

In EDQ 12.2.1, configuration for High Availability was simplified. At release 12.2.1 and above, all of the WebLogic Managed Servers in a Cluster can share the same database repository (in other words, only one pair of Config and Results datasources is required). And all of the WebLogic Managed Servers in a Cluster can share the same base and local config directories.


Oracle recommends that when you upgrade an instance of EDQ 11g or 12.1.3 that is configured for high availability, you change its configuration to take advantage of the simplified approach to High Availability introduced in EDQ 12.2.1. To do this you should:

- *Upgrade only one database repository (in other words, only one pair of Config and Results schemas) using the Upgrade Assistant.*
 - *Run the Oracle Fusion Middleware Reconfiguration Wizard a single time, supplying the connection details of the database repository that you upgraded. As you run the Wizard, at the Advanced Options screen, ensure that you select the **Managed Servers, Clusters and Coherence** option, and then, referring to the EDQ High Availability Guide for release 12.2.1, configure your upgraded EDQ instance for High Availability using the new, simplified architecture.*
 - *Ensure that all of your WebLogic Managed Server's Server Parameters (in other words, its 'startup arguments' or 'Java options') are configured to point towards the same base and local config directories.*
-

Although not recommended, it is also possible to maintain the existing high availability configuration after an upgrade, by continuing to use multiple schemas and config areas. To do this:

- *Run the Upgrade Assistant separately against each pair of Config and Results schemas.*
 - *Disable coherence by setting `coherence.clustering = false` in the **director.properties** file.*
-

1. Back up your old EDQ environment.

- 
- a. Before you begin, Oracle recommends that you make a complete backup of your old EDQ environment, including your EDQ domain. For instructions, see “Backup and Recovery Strategies for Upgrade” in the Planning an Upgrade of Oracle Fusion Middleware documentation.
 2. Shutdown your old EDQ environment’s Managed WebLogic Server.
 3. Ensure that your old EDQ environment’s WebLogic Server Administration Server remains running.
 4. Follow the instructions below to upgrade your **EDQCONFIG** (configuration), **EDQRESULTS** (results), and **OPSS**, **IAU** and **STB** schemas in the EDQ repository database.

- a. Start the Upgrade Assistant:

- i. Go to the following directory, where 12.2.1_FMW_HOME is the version 12.2.1 Oracle home directory.

On UNIX operating systems:

```
12.2.1_FMW_HOME/oracle_common/upgrade/bin
```

On Windows operating systems:

```
12.2.1_FMW_HOME\oracle_common\upgrade\bin
```

- ii. Run the following program:

On Linux or UNIX operating systems:

```
./ua
```

On Windows operating systems:

```
ua.bat
```

- b. At the Welcome screen, Review Important Reminders before Proceeding. The Welcome screen contains reminders to consider before proceeding with your upgrade. Make sure you read these and verify that you are ready to proceed. For assistance or more information on any screen, click Help. Otherwise, click **Next >**.

For more information about the following screens, see “Upgrading Schemas with the Upgrade Assistant” in Upgrading with the Upgrade Assistant.

- c. Select the type of upgrade you wish to perform.
 - i. Select **All Schemas Used By a Domain**.
 - ii. In the Domain Directory field, select your old EDQ environment’s domain directory (this is likely to be similar to **/OLD_FMW_HOME/user_projects/domains/edq_domain**).
 - iii. Click **Next >**.

- d. The Component List screen will display a list of the components (or in other words, schemas) to be upgraded in your old EDQ environment's domain. The following components will be listed:

- Oracle Enterprise Data Quality Results

- Oracle Audit Services

- Oracle Platform Security Services

- Common Infrastructure Services

- Oracle Enterprise Data Quality Configuration

- i. - Click **Next >**.

- e. In the Prerequisites screen, verify that you have met the prerequisites, and, provided that you have, check the relevant boxes (note that The Upgrade Assistant will not verify that the prerequisites have been met.). Then click **Next >**.

- f. Specify Database and Schema Credentials

- i. The EDQ Results schema, IAU, OPSS, STB, and EDQ Configuration schema screens are displayed in succession.
 - ii. On the first screen, specify connection details for the database that contains the EDQ Results (EDQRESULTS) schema, and then click **Connect**. You are then prompted for the password for the schema user.
 - iii. The remaining screens are automatically populated with the database connection and schema credentials that you supplied on the EDQ Results Schema screen. If these entries are not correct for any schema, change the entries and ensure that a database connection is made.

- g. Complete the upgrade validation.

- i. On the Examine screen, the Upgrade Assistant performs a series of validations before upgrading the selected components. Ensure that all validations have succeeded.

- h. Initiate the Upgrade.

- i. Click **Upgrade** on the Upgrade Summary screen to initiate the upgrade. The Upgrade Progress screens shows information about the progress of the upgrade, and the Upgrade Success screen summarizes the upgrade.


- 5. Create the Required 12c Schemas.

Oracle Fusion Middleware Release 12.2.1 introduces new schemas and tables that must be created with the Release 12.2.1 version of the Oracle Repository Creation Utility (RCU). This step of the upgrade must be performed before Reconfiguring the EDQ Domain.

- a. To create the required 12.2.1 schemas, see Section 4.2, "Creating an EDQ Database Repository" of the Installing and Configuring Oracle Enterprise Data Quality guide for release 12.2.1. Follow all of those instructions, but only select the following schemas on the Select Components screen:

- EDQ Staging Repository (note that you will find this under the Oracle EDQ option).

- 6. Reconfigure the EDQ Domain.



In this step you will run the Oracle Fusion Middleware reconfiguration wizard to complete the upgrade of your WebLogic Server domain environment.

- a. Check that the Staging Mode is set to Nostage.
 - i. Log in to the WebLogic Server Administration Server console from any web browser.
 - ii. In the left pane, expand **Environment** and then click **Servers**. The Summary of Servers screen will be displayed.
 - iii. In the Summary of Servers screen, click your EDQ Managed WebLogic Server (for example: **edq_server1**).
 - iv. Select the **Configuration** tab, then the **Deployment** tab.
 - v. From the Staging Mode menu, ensure that select nostage is selected, and, if it is not, select it. Do not change any other server settings.
 - vi. Click **Save**.
- b. Start Node Manager if it is not already running

*Note that you only need to complete this step if the Node Manager is **not** already running.*

This task ensures that Node Manager has been started at least once since it was installed. Starting Node Manager creates a set of script files that must be moved from the old domain to the release 12.2.1 domain. Performing this task eliminates the need for extra steps when upgrading your domain.

- i. Using a command line tool (for example, the Windows Command prompt or a Linux or Unix terminal), navigate to the **bin** folder in your old EDQ environment's WebLogic base domain (this will be at a location similar to **FMW_HOME/user_projects/domains/edq_domain/bin**).
- ii. Run the following command to start Node Manager. For **hostname**, specify the name of the system where Node Manager is installed, and for **port**, specify the port number on which it runs.

Windows:

```
startNodeManager.cmd hostname port
```

UNIX:

```
./startNodeManager.sh hostname port
```

- c. Stop Node Manager
 - i. If the Node Manager was not already running, and you started it using the command above, then stop it by closing the command shell from which it is running.
 - ii. If Node Manager was already running (for example, because it was already running as a service in your environment), then stop it using the appropriate means (for example, stop the Node Manager service).
- d. Start the Reconfiguration Wizard

In this step you will launch the wizard that will upgrade the EDQ domain.

- i. Stop the WebLogic Server Administration Server and all managed servers.
- ii. Go to the following directory, where **12.2.1_FMW_HOME** is the release 12.2.1 Fusion Middleware home directory.

On Windows operating systems:

```
12.2.1_FMW_HOME\oracle_common\common\bin
```

On UNIX operating systems:

```
12.2.1_FMW_HOME/oracle_common/common/bin
```

- iii. Start the domain reconfiguration wizard.

On Windows operating systems:

```
reconfig.cmd -log=log_file
```

On UNIX operating systems:

```
./reconfig.sh -log=log_file
```

Specify the full path and file name in place of `log_file`. Creating a log file can be helpful if you need to troubleshoot the reconfiguration process.

Note that if the following error message is displayed, it indicates that the default cache directory is not valid:

```
*sys-package-mgr*: can't create package cache dir
```

You can change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`. For example:

```
CONFIG_JVM_ARGS=-Dpython.cachedir=valid_directory
```

- e. Specify the domain.
 - i. On the Select Domain screen, specify the full path to the location of your old EDQ domain (it is likely to be similar to `FMW_HOME/user_projects/domains/edq_domain` or `FMW_HOME\user_projects\domains\edq_domain`). You can also click **Browse** and use the file manager window to help you select the domain location.
- f. View the Reconfiguration Setup Progress.
 - i. The Reconfiguration Setup Progress displays the reconfiguration progress and verifies whether the base domain that you selected can be reconfigured to the 12.2.1 domain. The message "Core WLS Infrastructure Reconfigured Successfully!" indicates that the domain can be reconfigured to the 12.2.1 domain, and you can click **Next** to go to the next step. If this message is not returned, the domain cannot be reconfigured to the 12.2.1 domain. If this happens, check to see if the EDQ version is earlier than version 11g; if so, then you must first upgrade to EDQ 11g and then upgrade to version 12.2.1.
- g. Select the Domain Mode and JDK.
 - i. It is not possible to change the Domain Mode from development to production or vice versa.

- ii. On the Domain Mode and JDK screen, specify the location of the Java Development Kit (JDK) to use in the new EDQ domain. This must be a JDK that conforms to the system requirements of your new EDQ environment.
- h. Select the Database Configuration Type.
 - i. In the Database Configuration Type screen, the **RCU Data** option should be selected by default. The connection details for your EDQ repository database should be automatically populated, and the schema owner and password for the Common Infrastructure schema (STB) should be displayed. If this information is not automatically populated, or is not correct, make any necessary amendments.
 - ii. Click **Get RCU Configuration**. The Reconfiguration Wizard should report that it can connect to the Database Server and that it can retrieve schema data and bind local schema components with the retrieved data. If these steps fail, amend the connection data and click **Get RCU Configuration** again. Otherwise, click **Next >**.
- i. Check the JDBC Component Schema.
 - i. On the JDBC Component Schema screen, the schemas associated with your old EDQ domain are listed in the lower half of the screen. If you need to make changes, select the check box next to the data source name and then make the changes. Note that the correct connection details for the EDQ Staging schema that you created using the Repository Creation Utility in step 5, above, may not be displayed. If they are not, amend them so that they are correct.
- j. Test the JDBC Component Schema
 - i. On the JDBC Component Schema Test screen, test the data source connections that were detected. Select the schemas that you want to test and then click Test Selected Connections.

To test a connection, the database to which you are connecting must be running.

- k. Select Optional Advanced Configuration Options

On the Advanced Configuration Screen, you can select additional domain options.

- ii. To make changes to the Managed Server configuration, select **Managed Servers, Clusters and Coherence**. For help with any of the options, click the **Help** button.
- ii. To skip making these changes, click **Next**.
- l. Initiate the Domain Reconfiguration
 - i. On the Configuration Summary screen, review the configuration and then click **Reconfig** to start the reconfiguration process, or click **Back** to make changes.
- m. Finish the Reconfiguration
 - i. Wait for the message that states, "Domain Reconfiguration Applied Successfully" and then click **Next**.

- ii. A check mark and the message “Oracle WebLogic Server Reconfiguration Succeeded” indicate that the reconfiguration was successful. Click **Next >**, and then click **Finish** to dismiss the reconfiguration wizard.

7. Follow the instructions below to upgrade EDQ's domain configuration.

a. Start the Upgrade Assistant:

- i. Go to the following directory, where 12.2.1_FMW_HOME is the version 12.2.1 Oracle home directory.

On UNIX operating systems:

```
12.2.1_FMW_HOME/oracle_common/upgrade/bin
```

On Windows operating systems:

```
12.2.1_FMW_HOME\oracle_common\upgrade\bin
```

- ii. Run the following program:

On Linux or UNIX operating systems:

```
./ua
```

On Windows operating systems:

```
ua.bat
```

- b. At the Welcome screen, Review Important Reminders before Proceeding. The Welcome screen contains reminders to consider before proceeding with your upgrade. Make sure you read these and verify that you are ready to proceed. For assistance or more information on any screen, click Help. Otherwise, click **Next >**.

For more information about the following screens, see “Upgrading Oracle WebLogic Component Configurations” in Upgrading with the Upgrade Assistant.

- c. At the Selected Schemas screen, select the **All Configurations Used By a Domain** option. At this point the name of the screen will change to All Configurations. In the **Domain Directory** field, select your old EDQ environment's domain directory (this is likely to be similar to **/OLD_FMW_HOME/user_projects/domains/edq_domain**), and click **Next >**.
- d. At the Component List screen, review the list of components to be upgraded (it should consist of Oracle JRF, System Components Infrastructure, and Oracle Enterprise Data Quality Configuration), and click **Next >**.
- e. At the Prerequisites screen, verify that you have met the prerequisites, and, provided that you have, check the relevant boxes (note that The Upgrade Assistant will not verify that the prerequisites have been met.). Then click **Next >**.

- f. At the Examine screen, the Upgrade Assistant will validate that the components can be upgraded, returning statuses for either upgrade not necessary or succeeded for each of them. When the examination has finished, click **Next>**.
 - g. At the Upgrade Summary screen, click **Upgrade**.
 - h. At the Upgrade Progress screen, wait until the upgrade has finished, and then click **Next>**.
 - i. At the Upgrade Success screen, click **Close**.
- 8.
9. Apply the Upgrade Changes to the Base Domain

These steps apply the changes that you made during the upgrade of the EDQ domain to the base domain.

- a. Start the WebLogic Node Manager for the upgraded EDQ domain.
- b. Start the WebLogic Server Administration Server from the upgraded EDQ domain. If it is running, stop and then restart it.


(UNIX) `DOMAIN_HOME/bin/startWebLogic.sh`

(Windows) `DOMAIN_HOME\bin\startWebLogic.cmd`

- c. Log in to the WebLogic Server Administration Server console from any web browser.
- d. In the Domain Structure list, click **Deployments**.
- e. Under Deployments, select **edq** and then click **Update**.
- f. Click **Next**.
- g. Click **Finish**.
- h. To complete the upgrade, start the EDQ managed server.

*The `startManagedWebLogic.sh` script is located in your EDQ WebLogic domain's bin directory. Running this script is one method of starting the EDQ WebLogic Managed Server. Note that the `startManagedWebLogic.sh` script may be overwritten when the EDQ WebLogic Domain is reconfigured. Therefore, if you customized the script, any customizations may have been lost during the upgrade process. This is a particularly important point to note if you embedded EDQ's Server Parameters (in other words, its 'startup arguments' or 'Java options') within this script, as, without these, the EDQ Managed Server may not function correctly. See section 6 of the *Installing and Configuring Oracle Enterprise Data Quality* guide, "Setting Server Parameters to Support Enterprise Data Quality", for more information about EDQ's server parameters.*

10. Open a Web Browser, and navigate to your upgraded EDQ environment's Launchpad (unless you changed any of the default options when installing EDQ, its address will be <http://your-server-name:8001/edq>).
11. From the launchpad, open the Director user interface and log in.
12. Verify that all your projects have been successfully migrated.

- 
13. If any of your processes use Match Review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

Upgrade Path 8

From any Release of EDQ Installed Manually (i.e. Without Using the Oracle Universal Installer) on an Oracle / WebLogic Platform to EDQ 12.2.1 Using an Oracle / WebLogic Platform

Overview

Upgrading from a release of EDQ that was installed manually (that is, without the Oracle Universal Installer) and which uses the Oracle database and the WebLogic application server as its platform to EDQ 12.2.1 using the Oracle Database and Oracle WebLogic as its platform is essentially a two-part process:

- Part 1 – Install EDQ 12.2.1
- Part 2 – Migrate configuration data from your old EDQ environment to your new EDQ 12.2.1 environment. This entails using scripts provided with the EDQ 12.2.1 release to reshape and move data. It also entails manually copying some files.

Part 1: Install EDQ 12.2.1

2. Install Enterprise Data Quality 12.2.1 using the Oracle WebLogic Server as its Java Application Server and the Oracle Database as its structured query language (SQL) relational database management system (RDBMS); refer to <http://add-a-link-to-the-12.2.1-installation-doc> for more information about how to do this. During part 2, you will need to know the following information about your EDQ 12.2.1 environment:
 - a. The repository database's Host Name, Port Number and Database ID.
 - b. The schema owner and password for the config and results schemas.
 - c. The environment's domain location.

Part 2: Migrate the data from the old DIRECTOR schema to the new EDQCONFIG schema

16. Back up elements of your old EDQ environment.

During part 2, you will migrate configuration data from your old EDQ repository to your new EDQ repository, and you will also copy configuration files from your old environment to your new environment. You backup elements of your old environment (that is, the configuration data and the configuration files) so that you can easily restart this process should anything go seriously wrong.

- a. Before you back up your old EDQ environment, shut down its WebLogic Managed server.
- b. Take a copy of your old EDQ environment's configuration folder. By convention this is usually named either **config**, **oedq_local_home** or **oedq.local.home**. (note that EDQ 9.0 and below had only a single configuration folder; for release 11 and above, it is the local configuration folder that you need to back up).
- c. Backup your old EDQ environment's database repository using the appropriate Oracle tools.
- d. **Do not** restart your old EDQ environment's application server.

In case of a serious error during the upgrade you are about to perform (for example, if the migration script should fail to complete, and you are left with a schema that is stuck in between your old version of EDQ and your new version of EDQ), you can now fairly easily start the process again from scratch. How?

- Stop your old EDQ environment's WebLogic Managed server if it is running.
- Delete the original version of the folder you have backed up.
- Copy the backup into its original location.
- Restore the database repository from the backup.

...And then you can start the migration process again from scratch.

17. Run and examine the premigration report. This will reveal any problems with the Case Management schema (which is, strictly speaking, not a schema at all, but just a set of tables within EDQ's config schema). If any problems are revealed, you should fix them before proceeding to step 3.

- a. Open a command line tool:
 - i. In a Windows environment, open a Windows Command Prompt, running it as administrator (for example, right-click Command Prompt in the All Programs > Accessories group, and select **Run as administrator**).
 - ii. In a Unix or Linux environment, open a terminal window.
- b. Change directory to your Java Development Kit's bin folder. You can do this by issuing the command `cd path_to_the_jdk_bin_folder` (for example, in a Windows environment you might enter a command similar to: `cd C:\Program Files\Java\jdk1.8.0_40\bin`).
- c. Run the pre-migration report by issuing the following command:

```
java -jar path_to_migration.jar premigrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with your new EDQ environment. This should be in `~ORACLE_HOME\edq\oracle.edq`
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the `SID` for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to run the premigration report in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar
preigrate oracle:mydatabase@localhost:1521/director/director
```

- d. If the Premigration report reveals any problems, fix them, and run the report again before proceeding further. (If the premigration report returns no output, by the way, that's a good thing: it means no problems were found.)

Examples of Problems Revealed by the Premigration Report, With Suggested Fixes

Example One: Let's say that you see the following error in the premigration report:

```
Parameter x in Source y for Case Type z using Selection Function aa "
" is using an attribute under input 'bb that is not mapped to case management.
This input will not be added automatically mapped and must be manually
configured."
```

In some earlier releases of EDQ it was possible for the value of a workflow parameter to be taken from a process attribute via a selection function even though the process attribute was not mapped to a case source attribute. For example, a workflow parameter called 'tel number' could be populated with the value of either 'cell-phone number' or 'landline number' via a First Non-empty Value selection function, even though neither 'cell-phone number' nor 'landline number' were mapped to an attribute in the case source. This is not permitted in later releases of EDQ. In order to fix this problem, you would need to create two new case source attributes and map 'cell-phone number' and 'landline number' to them. If you didn't do this, the migration would succeed, but in your upgraded system the 'tel number' parameter would not be populated, which may affect the correct workings of Case Management..

Example Two: Let's say that you see an error similar to this one in the premigration report:

```
Parameter X in Source Y for Case Type Z is defined differently across multiple
match processes. Please make these configurations consistent.
```

In later releases of EDQ, a parameter within a case source can only be mapped to a single attribute, but this was not the case in older releases. This error message is reporting that the same case source is used in two (or more) match processes, and that the same parameter has been mapped to more than one attribute in these match processes. Before migrating the data, you should resolve this by either:

- Mapping the parameter to the same attribute in each of the match processes.
 - Or creating a new parameter and mapping this to one of the attributes.
-

If you do not resolve this, the migration would succeed, but in your upgraded system the parameter would not be populated.

Example Three: Let's say that you see an error similar to this one in the premigration report:

```
No metadata for source XXX guid b491939a-e6fd-33d5-9752-241cbe26a378 (Code: 300,551)
```

```
Error executing script: java.lang.IllegalArgumentException
```

In older releases of EDQ, it was possible to import a Case Management configuration file (a .dxc file) that did not contain some historical structures, and this error indicates that this has been done in your old EDQ environment. If a Case Management configuration file that includes all historical case source structures in it, then importing this will solve the problem. If such a file is not available, contact Oracle Support before proceeding with your upgrade.

18. Upgrade the configuration schema.

- a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.

Note: If you are migrating from EDQ release 8.1 or below, then you will need to migrate the EDQ configuration schema in two steps:

- Step 1: migrate the configuration schema to release 9.0.
 - Step 2: migrate the configuration schema to your target 12c release.
-

- b. **Only complete this step if you are migrating from EDQ release 8.1 or below** (see the note directly above):

- i. Acquire a distribution of EDQ 9.0. (you will not need to fully install EDQ 9.0, but you will need access to its migration scripts.) The easiest way to acquire EDQ 9.0 is to download and then unzip the Windows Quick-Start installer for EDQ 9.0.
- ii. Upgrade the configuration schema to release 9.0 by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with the EDQ 9.0 distribution. This should be in `~edq-9_0_x(XXX)\9.0.x.(XXX)` (e.g. `edq-9_0_8_(1103)\9.0.8.(1103)`).
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.

- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to EDQ 9.0 in a Windows environment is:

```
java -jar C:\Users\Fred\Downloads\edq-9_0_8_(1103)\9.0.8.(1103)\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

- c. Upgrade the configuration schema to your EDQ 12c release by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ 12c environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to your new EDQ 12c release in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

19. Upgrade the Case Management schema.

- a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
- b. Upgrade the Case Management schema by issuing the following command:

```
java -jar path_to_migration.jar cm -db dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the Case Management schema in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar cm -db
oracle:mydatabase@localhost:1521/director/director
```

20. Migrate the upgraded configuration and Case Management data from your old EDQ environment's repository to your new EDQ environment's repository.
 - a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
 - b. Start the Database Copy tool by issuing the following command:

```
java -jar path_to_rcumigration.jar
```

Where:

- `path_to_rcumigration.jar` is the path to the **rcumigration.jar** file that was installed with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**

An example of a command to start the Database Copy tool in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\rcumigration.jar
```

The Database Copy Tool window will open at the Welcome screen.

- c. At the welcome screen, click **Next >** to proceed to the Configuration Directories screen.
- d. At the Configuration Directories screen:
 - i. If upgrading from EDQ 9.0.x or below, leave the **Base Configuration directory location** field empty, and select the location of your old EDQ environment's

configuration directory (by convention, this is often called **config**) in the **Local Configuration directory location** field.

- ii. If upgrading from EDQ 11.x or above, select your old EDQ environment's base configuration directory location (this folder often called either **edq_home** or **edq.home**) in the **Base Configuration directory location** field; then select the location of your old EDQ environment's local configuration directory in the **Local Configuration directory location** field (this folder is often called either **edq_local_home** or **edq.local.home**).

The reason for this step is so that the Database Copy tool can find the connection details for your old EDQ environment's repository, as this information is stored in the configuration directory.

- e. Click **Next >** to proceed to the Source Config Database screen.
- f. The Source Config Database screen should be populated with the connection details for the config schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Source Results Database screen.
- g. The Source Results Database screen should be populated with the connection details for the results schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Destination Config Database screen.
- h. In the Destination Config Database screen, enter the connection details for your new EDQ environment's config schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Destination Results Database screen.
- i. In the Destination Results Database screen, enter the connection details for your new EDQ environment's results schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Summary screen.

Your new EDQ environment's config and results schemas were created by the Repository Creation Utility (RCU) when you installed EDQ during part 1 above. You will have supplied the RCU with the information you need now (Datasource Username, Datasource Password, HOST NAME, Port Number and Database ID). The RCU will suggest default Schema Owners of DEV_EDQCONFIG and DEV_EDQRESULTS respectively for your new EDQ environment's config and results schemas. Unless you changed these defaults, these are the values that you should now enter into the Datasource username fields.

- j. In the Summary screen, select the **Skip Empty Check** checkbox (Your new EDQ environment the destination schemas will not be empty). Also select the **DATA IN DESTINATION DATABASES WILL BE DELETED** check box (in this case, we don't want to keep any of the default data that was inserted when our new EDQ environment was created).
- k. Click **Next >..** A dialog box informs you that you are about to commence the copying of data, and that all data in the destination databases will be deleted. Click **OK**.

- I. The Database Copy Tool now migrates the configuration (and some results) data from your old EDQ environment to your new EDQ environment, reporting its progress as it does so. When it has completed this operation, the **Finished** screen is displayed. This reports that your new database is ready to use, and provides the location of database copy tool's logfile. Click **Finish**.

21. Manually copy some of the configuration folders and files from your old EDQ environment to your new EDQ environment.

*The location of your old EDQ environment's local configuration folder will depend upon the release of EDQ that you are migrating from, which operating system it is installed on, and how it was installed. You should already know it, as you entered it during step 5d, above. Note that EDQ release 9.x and lower only had a single configuration directory, which, by convention, was often named **config** (this is the folder you should copy from); EDQ release 11.x and above has two configuration directories – one 'base' and one 'local'. If upgrading from EDQ release 11.x or above, copy from the 'local' configuration directory.*

*Your new EDQ environment's local configuration folder will have been created when you installed EDQ in part 1, above. By default, it will be called **oedq.local.home**, and it will be located somewhere under the domain location that was created in part 1. (For example:*

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\base_domain\config\fmwconfig\edq\oedq.local.home).

- a. Copy the entire contents of the following folders, including any sub-folders, from your old EDQ environment's local configuration folder (in EDQ releases prior to 11, there was only a single config folder, and you should copy from this) to your new EDQ environment's local configuration folder, merging them with any existing folders that have the same name:
 - i. **autorun**
 - ii. **av**
 - iii. **blueprints**
 - iv. **buckets**
 - v. **businessrules**
 - vi. **capscan**
 - vii. **commandarea**
 - viii. **dbconnectors**
 - ix. **edqp**
 - x. **extras**
 - xi. **housekeeping**
 - xii. **landingarea**
 - xiii. **localeselection**
 - xiv. **localwidgets**

- xv. **match**
- xvi. **properties**
- xvii. **qas**
- xviii. **resources**
- xix. **runprofiles**
- xx. **security**
- xxi. **webservices**

Note that, depending upon the EDQ version that you are migrating from, and how it has been configured, some of the folders listed above may not exist.

- b. Treat the **casemanagement** folder as follows:
 - i. Copy entire contents of the **casemanagement** folder, **except** for the indexes folder and the **flags.xml** file.
 - ii. If you **have not** made any changes to the **flags.xml** file, copy that file too.
 - iii. If you **have** made changes to the **flags.xml** file, manually copy any new extended attributes that you added, and paste them into the version of **flags.xml** that you will find in your new EDQ environment's **casemanagement** folder. However, ensure that you do not remove or modify any of the default extended attributes that you will find within your new EDQ environment's **casemanagement** folder.
- c. Treat the **localgadgets** folder as follows:
 - i. If you are migrating from EDQ release 8.1.x or above, copy entire contents of the **localgadgets** folder.
 - ii. If you are migrating from EDQ release 8.0.x or below, then copy the entire contents of the **localgadgets** folder except for any selection functions. You will need to create new versions of any tailored selection functions and place them in the **localselection** sub-folder of your new EDQ environment's local configuration folder. Existing custom selection functions will not work after upgrade.
- d. Treat the **notification** folder as follows:
 - i. If you are migrating from EDQ release 9.0.x or above, copy the entire contents of the **notification** folder.
 - ii. If you are migrating from EDQ release 8.1.x or below, copy the entire contents of the **notification** folder except for the **jobs** and **eventlog** sub-folders.

The **jobs** and **eventlog** folders cannot be copied from EDQ release 8.1.x or lower because job template code was restructured in EDQ 9.0 and therefore modified versions of the job templates created prior to release 9.0 will not work. (If you do erroneously copy these folders, you will find that job notification e-mails will not work, and that the job notifications will not be visible in the event log.) If the contents of these folders were not modified in your old EDQ environment, then you need do nothing: the default files provided with your new EDQ environment will work out-of-the-box. But if you have carried out any modifications, then any changes you made will need to be re-applied to the new versions of the templates - copying over the old versions will not work.

i. Treat the **triggers** folder as follows:

xvii. Check through each of the files in the triggers directory, and in any sub-directories.

If you find the following line in any file:

```
importClass("com.datanomic.director.guihelper.users.server.UserCache");
```

Change it to:

```
importClass("com.datanomic.userauth.server.cache.UserCache");
```

xviii. Once you have made any necessary changes, copy the entire contents of the **triggers** folder, including any sub-folders, from your old EDQ environment's configuration folder to your new EDQ environment's local configuration folder, merging it with any existing folder that has the same name.

j. Copy the following files from the root of your old EDQ environment's local configuration folder to the root of your new EDQ environment's local configuration folder:

xix. kfile

xx. override.properties

22. Restart your new EDQ environment's WebLogic Managed server (in other words, the EDQ application server that is called **edq_server1** unless you changed the default name when you installed EDQ). Exactly how you do this depends on your operating system and how your new EDQ environment has been configured (for example, whether you have created a script to start EDQ's Managed WebLogic server).

If you have created a script to start EDQ's Managed WebLogic server, then it is usually possible to restart it via that script.

If you have not created a script to start EDQ's Managed WebLogic server, then follow these steps:

- Ensure that the WebLogic Node Manager and the WebLogic Admin server are both running. You can stop and start the WebLogic Node Manager and the WebLogic Admin server using .cmd Windows files or Unix/Linux shell scripts that you can find in the **bin** folder of your new EDQ environment's edq_domain (An example of the path to this directory on a Windows machine is:

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\edq_domain\bin). The relevant files will be called: **startNodeManager.cmd / startNodeManager.sh** and **startWebLogic.cmd / startWebLogic.sh**.

- Open the WebLogic console in a web browser (the default address is usually **http://your-server-name:7001/console**).

- In the Domain Structure tree, on the left-side of the screen, select **Environments > Servers** to proceed to the Summary of Servers Screen.

- In the Summary of Servers screen, navigate to the **Control** tab.

- In the Control tab, select the checkbox for the EDQ Managed Server (by default, the EDQ Managed Server is called **edq_server1**). Then, from the **Shutdown** dropdown, select **When work completes**.

- Wait for the EDQ Managed Server to Shutdown, then ensure that its checkbox is selected, and click **Start**.

23. Open a Web Browser, and navigate to your new EDQ environment's Launchpad (unless you changed any of the default options when installing EDQ, its address will be <http://your-server-name:8001/edq>).

24. From the launchpad, open the Director user interface and log in.

Note that the user name and password for your new EDQ environment will not necessarily be the same as they were for your old EDQ environment. Unless you changed any of the default options when installing EDQ, the out-of-the-box user name for your new environment will be **weblogic**. You will have provided a password during the installation process.

25. Verify that all your projects have been successfully migrated.

26. Run each of your jobs or standalone processes to regenerate their results (the results will not have been migrated).

Note that you should run jobs that require a run profile or run label from the Server Console user interface rather than from the Director user interface.

Note also that, if any of your processes use match review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

In some cases a job or process may fail with one of the following error messages:

Cannot load data for replacements list reference data "X". Duplicate key "Y".

Cannot load data for replacements list supplied reference data. Duplicate key "{0}" .

Prior to release 11, EDQ was tolerant of reference data with duplicate lookup keys. EDQ 11 and higher is less tolerant of Reference Data with duplicate lookup keys. The errors above indicate that your migrated environment includes reference data that has duplicate lookup keys.

In this situation, Oracle recommends that you should deduplicate the reference data in question (on a case-insensitive basis). You can do this either manually if the volume of reference data is small, or by creating an EDQ process and using the Group and Merge processor if the volume of reference data is large.

*Alternatively - although it is not recommended - if previous behavior is required, you can add the following property to your new EDQ environment's **director.properties** file:*

```
replace.transform.allow.duplicate.key = true
```

*Note that the **director.properties** file is located in your new EDQ environment's local configuration directory (by default, this will be called **oedq.local.home**).*

*Enterprise Data Quality has a limit of 4000 characters when writing data to the Oracle Database. When EDQ runs snapshots, it truncates fields to ensure that they are within this limit. However, in very rare cases, EDQ processes can generate strings that are longer than the character limit. Unpatched instances of Oracle Watchlist Screening 11.1.1.7 can, for example, generate very long strings that contain delimited lists of watchlist ids matched by a single customer record. When processes produce strings that are longer than the limit, they will fail. This means that it is possible that processes that ran previously under Postgres will fail on Oracle. In such cases the EDQ log would contain an error such as **ORA-01461: can bind a LONG value only for insert into a LONG column**. If you experience this error, contact Oracle support.*

27. If you are using Case Management, then, from the Launchpad, open the Case Management user interface, log in and verify that all of your cases and alerts have been successfully migrated.

If Case Management appears to be missing cases and alerts (for example, if a filter that should return several cases and alerts in fact returns none), it is possible that the case management indexes need to be re-generated. Follow this procedure to re-generate Case Management indexes:

- *Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is **C:\Program Files\Java\jdk1.8.0_40\bin**).*
 - *Run the jconsole application (how you run this will depend upon your operating system – you may simply be able to double-click the application).*
-


-
- The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Local Process** radio button, and then highlight the top-most instance of the **weblogic.server** that appears in the list of local processes (this should be EDQ's Managed WebLogic Server). Input the username for your new EDQ environment (by default, this is **weblogic**), and password. Then click **Connect**.
 - Navigate to the **Mbeans** tab.
 - Within the tree, on the left-side of the Window, expand **edq** -> **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.
 - Ensure that **clear** is set to **true**.
 - Click **reindex**. An Info dialog box will report that the Method was successfully invoked.
-

28. Within Case Management, verify that you can successfully search for and find Cases and Alerts.

*If you cannot successfully search for and find Cases and Alerts, this is likely to indicate that your flags.xml file is incorrect. The flags.xml file is located in your new EDQ environment's casemanagement directory, which is itself within the local configuration directory (called **oedq.local.home** by default). flags.xml contains information about extended attributes. A fresh EDQ installation comes with a flags.xml file that includes a number of default extended attributes. You can add new extended attributes to flags.xml, but if you removed or changed the default extended attributes that came with the original file, then EDQ will not be able to properly index Cases and Alerts, and you will therefore not be able to successfully search for them. You should resolve this by putting the default extended attributes back into flags.xml, and reindexing cases and alerts. (How to reindex is explained in the note above this one.). One way to do this is to copy the default extended attributes from a fresh version of flags.xml. You will find a fresh version of flags.xml in your new EDQ environment's base configuration folder, which is called **oedq.home** by default.*

29. Out-of-the-box, your new EDQ environment's Launchpad may not display links to all of your EDQ user interfaces. To configure the launchpad, click **Log In**, enter your EDQ user name and password, and then select **Administration > Launchpad Configuration**. You can then select the user interfaces that you want to display on the Launchpad.
30. All of the user groups from your old EDQ environment will have been migrated to your new EDQ environment. However, the users from your old environment will no longer be active. This is because in your new EDQ environment, users are managed within your new WebLogic environment. You will need to:
- a. Use the WebLogic Server Administration Console to create a series of WebLogic user groups to mirror each of your old environment's user groups.

*To get to the part of the WebLogic Server Administration Console that you will require, in the Domain Structure select **Security Realms**. Then click your security realm (by default this is called **myrealm**) and navigate to the **Users and Groups** tab.*

- 
- b. Use the WebLogic Server Administration Console to recreate all of your old environment's users, assigning each of these users to one of the WebLogic user groups you created in the previous step.
 - c. Use the EDQ Launchpad (**Administration > External Groups**) to map each of the WebLogic user groups you created (which EDQ will see as **External Groups**) to the appropriate EDQ user group.

Upgrade Path 9:

From any Release of EDQ Using a Postgres / Tomcat Platform to EDQ 12.2.1 Using an Oracle / WebLogic Platform

Overview

Upgrading from a release of EDQ that uses the Postgres database and the Tomcat application server as its platform to EDQ 12.2.1 using the Oracle Database and Oracle WebLogic as its platform is essentially a two-part process:

- Part 1 – Install EDQ 12.2.1
- Part 2 – Migrate configuration data from your old EDQ environment to your new EDQ 12.2.1 environment. This entails using scripts provided with the EDQ 12.2.1 release to reshape and move data. It also entails manually copying some files.

Part 1: Install EDQ 12.2.1

1. Install Enterprise Data Quality 12.2.1 using the Oracle WebLogic Server as its Java Application Server and the Oracle Database as its structured query language (SQL) relational database management system (RDBMS); refer to <http://add-a-link-to-the-12.2.1-installation-doc> for more information about how to do this. During part 2, you will need to know the following information about your EDQ 12.2.1 environment:
 - a. The repository database's Host Name, Port Number and Database ID.
 - b. The schema owner and password for the config and results schemas.
 - c. The environment's domain location.

Part 2: Migrate the data from the old DIRECTOR schema to the new EDQCONFIG schema

2. Back up elements of your old EDQ environment.

During part 2, you will migrate configuration data from your old EDQ repository to your new EDQ repository, and you will also copy configuration files from your old environment to your new environment. You backup elements of your old environment (that is, the configuration data and the configuration files) so that you can easily restart this process should anything go seriously wrong.

- a. Before you back up your old EDQ environment you need to shut down EDQ's Tomcat application server and the EDQ repository (which is held in Postgresql databases). How you do this will vary by operating system.
 - i. In Windows there are usually three Windows services: one for the application server, and one for each of the Director and Results databases. You can stop these via the **Control Panel > Administrative Tools > Services**.

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the three Windows services will be called **Datanomic Application Server**, **Datanomic dnDirector Store** and **Datanomic dnResults Store**.*

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the three Windows services will be called **Oracle EDQ Application Server**, **Oracle EDQ Config Store** and **Oracle EDQ Results Store** respectively.

- ii. In Linux or Unix environments, it is likely that the application server and repository database(s) are started via init.d services (it's quite likely you will find them in **/etc/rc.d/init.d**). The service name and method of stopping them will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name stop** from a terminal window will often work.
- b. Take a copy of the folder that contains the Postgresql data, which is usually called **data**. The location and name of this folder may vary depending on your operating system and type of installation.
 - i. For Windows environments, see the note below.

If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the folder that contains the Postgresql data will be **C:\Program Files\Datanomic\dnDirector\data**.

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the folder that contains the Postgresql data will depend upon which version of Windows you have installed EDQ on. In Windows 7, for example, the default location is **C:\ProgramData\Oracle\Enterprise Data Quality\data** (note that you may have to change Windows Folder Options to view hidden files and folders before you can see this location).

- ii. In Linux or Unix environments, a common location is **/opt/PostgreSQL/version_number/data**. However, there is no single standard location, and you may need to ask your system administrator to tell you what the path is.
- c. Take a copy of your old EDQ environment's configuration folder, which is likely to be called either **config** or **oedq_local_home**.
 - i. For Windows environments, see the note below.

If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the configuration folder will be called **config**, and its default location is **C:\Program Files\Datanomic\dnDirector\config**.

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the configuration folder will be called **oedq_local_home**. Its location will depend upon the version of Windows you have installed on. On a Windows 7 machine, for example, the default location of **oedq_local_home** is:
C:\ProgramData\Oracle\Enterprise Data Quality\oedq_local_home.

- ii. In Linux or Unix environments, there is no default location or name for the EDQ configuration folder, and you may need to ask your system administrator to tell you what the path is. Having said that, by convention, the folder is often called either **config** or **oedq.local.home** or **oedq_local_home**.
- d. Restart your old EDQ environment's Repository (that is to say, the Results and Config databases), but **do not** restart its application server.
 - i. For Windows environments, start two of the three Windows services that you stopped above (do not restart the service for the application server though).
 - ii. In Linux or Unix environments, start the init.d service for the repository database, but do not start the init.d service for the application server. The service name and method of starting the repository database will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name start** from a terminal window will often work.

In case of a serious error during the upgrade you are about to perform (for example, if the migration script should fail to complete, and you are left with a schema that is stuck in between your old version of EDQ and your new version of EDQ), you can now fairly easily start the process again from scratch. How?

- Stop your old EDQ environment's application server and repository.
 - Delete the original versions of the folders you have backed up.
 - Copy the backups into the original locations.
 - Restart your old EDQ environment's repository.
- ...And then you can start the migration process again from scratch.*
-

3. Run and examine the premigration report. This will reveal any problems with the Case Management schema (which is, strictly speaking, not a schema at all, but just a set of tables within EDQ's config schema). If any problems are revealed, you should fix them before proceeding to step 3.
 - a. Open a command line tool:
 - i. In a Windows environment, open a Windows Command Prompt, running it as administrator (for example, right-click Command Prompt in the All Programs > Accessories group, and select **Run as administrator**).
 - ii. In a Unix or Linux environment, open a terminal window.
 - b. Change directory to your Java Development Kit's bin folder. You can do this by issuing the command **cd path_to_the_jdk_bin_folder** (for example, in a Windows environment you might enter a command similar to: **cd C:\Program Files\Java\jdk1.8.0_40\bin**).
 - c. Run the pre-migration report by issuing the following command:

```
java -jar path_to_migration.jar premigrate dbtype:dbid@host:port/user/password
```

Where:
 - **path_to_migration.jar** is the path to the **migration.jar** file that is delivered with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
 - **dbtype** is **pgsql** for PostgreSQL or **oracle** for Oracle.

- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to run the premigration report in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar  
premigrate pgsql:director@localhost:9000/director/director
```

- d. If the Premigration report reveals any problems, fix them, and run the report again before proceeding further. (If the premigration report returns no output, by the way, that's a good thing: it means no problems were found.)

Examples of Problems Revealed by the Premigration Report, With Suggested Fixes

Example One: Let's say that you see the following error in the premigration report:

```
Parameter x in Source y for Case Type z using Selection Function aa "  
" is using an attribute under input 'bb that is not mapped to case management.  
This input will not be added automatically mapped and must be manually  
configured."
```

In some earlier releases of EDQ it was possible for the value of a workflow parameter to be taken from a process attribute via a selection function even though the process attribute was not mapped to a case source attribute. For example, a workflow parameter called 'tel number' could be populated with the value of either 'cell-phone number' or 'landline number' via a First Non-empty Value selection function, even though neither 'cell-phone number' nor 'landline number' were mapped to an attribute in the case source. This is not permitted in later releases of EDQ. In order to fix this problem, you would need to create two new case source attributes and map 'cell-phone number' and 'landline number' to them. If you didn't do this, the migration would succeed, but in your upgraded system the 'tel number' parameter would not be populated, which may affect the correct workings of Case Management..

Example Two: Let's say that you see an error similar to this one in the premigration report:

Parameter X in Source Y for Case Type Z is defined differently across multiple match processes. Please make these configurations consistent.

In later releases of EDQ, a parameter within a case source can only be mapped to a single attribute, but this was not the case in older releases. This error message is reporting that the same case source is used in two (or more) match processes, and that the same parameter has been mapped to more than one attribute in these match processes. Before migrating the data, you should resolve this by either:

- Mapping the parameter to the same attribute in each of the match processes.
- Or creating a new parameter and mapping this to one of the attributes.

If you do not resolve this, the migration would succeed, but in your upgraded system the parameter would not be populated.

Example Three: Let's say that you see an error similar to this one in the premigration report:

No metadata for source XXX guid b491939a-e6fd-33d5-9752-241cbe26a378 (Code: 300,551)

Error executing script: java.lang.IllegalArgumentException

In older releases of EDQ, it was possible to import a Case Management configuration file (a .dxc file) that did not contain some historical structures, and this error indicates that this has been done in your old EDQ environment. If a Case Management configuration file that includes all historical case source structures in it, then importing this will solve the problem. If such a file is not available, contact Oracle Support before proceeding with your upgrade.

4. Upgrade the configuration schema.

- a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.

Note: If you are migrating from EDQ release 8.1 or below, then you will need to migrate the EDQ configuration schema in two steps:

- Step 1: migrate the configuration schema to release 9.0.
- Step 2: migrate the configuration schema to your target 12c release.

-
- b. **Only complete this step if you are migrating from EDQ release 8.1 or below** (see the note directly above):

- i. Acquire a distribution of EDQ 9.0. (you will not need to fully install EDQ 9.0, but you will need access to its migration scripts.) The easiest way to acquire EDQ 9.0 is to download and then unzip the Windows Quick-Start installer for EDQ 9.0.
- ii. Upgrade the configuration schema to release 9.0 by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with the EDQ 9.0 distribution. This should be in **~edq-9_0_x(XXX)\9.0.x.(XXX)** (e.g. `edq-9_0_8(1103)\9.0.8.(1103)`).
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to EDQ 9.0 in a Windows environment is:

```
java -jar C:\Users\Fred\Downloads\edq-9_0_8(1103)\9.0.8.(1103)\migration.jar migrate postgres:director@localhost:9000/director/director
```

- c. Upgrade the configuration schema to your EDQ 12c release by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ 12c environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to your new EDQ 12c release in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar migrate postgres:director@localhost:9000/director/director
```

5. Upgrade the Case Management schema.

- a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
- b. Upgrade the Case Management schema by issuing the following command:

```
java -jar path_to_migration.jar cm -db dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the Case Management schema in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar cm -db postgres:director@localhost:9000/director/director
```

6. Migrate the upgraded configuration and Case Management data from your old EDQ environment's repository to your new EDQ environment's repository.

- a. Shutdown your new EDQ environment's WebLogic Managed server (by default called `edq_server1`). One way to do this is from the WebLogic Server Administration Console: within the Domain Structure, expand **Environment** and select **Servers**. Then navigate to the **Control** tab, select the EDQ Managed Server (by default, this is called `edq_server1` and select **Shutdown > When work completes**).
- b. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.

- c. Start the Database Copy tool by issuing the following command:

```
java -jar path_to_rcumigration.jar
```

Where:

- `path_to_rcumigration.jar` is the path to the **rcumigration.jar** file that was installed with your new EDQ environment. This should be in `~ORACLE_HOME\edq\oracle.edq`

An example of a command to start the Database Copy tool in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\rcumigration.jar
```

The Database Copy Tool window will open at the Welcome screen.

- d. At the welcome screen, click **Next >** to proceed to the Configuration Directories screen.
- e. At the Configuration Directories screen:
- If upgrading from EDQ 9.0.x or below, leave the **Base Configuration directory location** field empty, and select the location of your old EDQ environment's configuration directory (usually called **config**) in the **Local Configuration directory location** field.
 - If upgrading from EDQ 11.x or above, select your old EDQ environment's base configuration directory location (this folder is usually called either **edq_home** or **edq.home**) in the **Base Configuration directory location** field; then select the location of your old EDQ environment's local configuration directory in the **Local Configuration directory location** field (this folder is usually called either **edq_local_home** or **edq.local.home**).

The reason for this step is so that the Database Copy tool can find the connection details for your old EDQ environment's repository, as this information is stored in the configuration directory.

- f. Click **Next >** to proceed to the Source Config Database screen.
- g. The Source Config Database screen should be populated with the connection details for the config schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Source Results Database screen.
- h. The Source Results Database screen should be populated with the connection details for the results schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Destination Config Database screen.
- i. In the Destination Config Database screen, enter the connection details for your new EDQ environment's config schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Destination Results Database screen.

- j. In the Destination Results Database screen, enter the connection details for your new EDQ environment's results schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Summary screen.

Your new EDQ environment's config and results schemas were created by the Repository Creation Utility (RCU) when you installed EDQ during part 1 above. You will have supplied the RCU with the information you need now (Datasource Username, Datasource Password, HOST NAME, Port Number and Database ID). The RCU will suggest default Schema Owners of DEV_EDQCONFIG and DEV_EDQRESULTS respectively for your new EDQ environment's config and results schemas. Unless you changed these defaults, these are the values that you should now enter into the Datasource username fields.

- k. In the Summary screen, select the **Skip Empty Check** checkbox (Your new EDQ environment the destination schemas will not be empty). Also select the **DATA IN DESTINATION DATABASES WILL BE DELETED** check box (in this case, we don't want to keep any of the default data that was inserted when our new EDQ environment was created).
- l. Click **Next >..** A dialog box informs you that you are about to commence the copying of data, and that all data in the destination databases will be deleted. Click **OK**.
- m. The Database Copy Tool now migrates the configuration (and some results) data from your old EDQ environment to your new EDQ environment, reporting its progress as it does so. When it has completed this operation, the **Finished** screen is displayed. This reports that your new database is ready to use, and provides the location of database copy tool's logfile. Click **Finish**.

In rare cases, the database copy may fail with an error similar to this one when you migrate your config schema from a Postgres database to the Oracle database:

```
Copying table: dn_xtree
Record skipped as value in Column 'node_name' in Table 'dn_xtree' is too
long:
??123456789A123456789A123456789A123456789A123456789A123456789A123456789A123456
```

The node_name mentioned in this error message is held in a field within EDQ's config schema. It is the name of one of the nodes that you can see within the Director user interface's Project Browser (for example, it may be the name of a project, or the name of a process, or the name of a set of staged data etc.). The length of fields in your old EDQ environment's Postgres config schema were defined in numbers of characters, some of which could be multi-byte characters. Field lengths in your new EDQ environment's Oracle database config schema are defined in numbers of bytes. The error is telling you that the node name used in your old EDQ environment is too long for your new EDQ environment. To fix this problem, you will need to open your old EDQ environment's Director user interface and, within the Project Browser, shorten the node name that has caused this error, removing any multi-byte characters if necessary. Bear in mind that you will have to revert to the backup you took during part 2, step 1, and start your old environment's application server before you can fix this problem. And once you have fixed the problem, you will then need to start part 2 of the upgrade process again.

7. Manually copy some of the configuration folders and files from your old EDQ environment to your new EDQ environment.

*The location of your old EDQ environment's local configuration folder will depend upon the release of EDQ that you are migrating from, which operating system it is installed on, and how it was installed. You should already know it, as you entered it during step 5d, above. Note that EDQ release 9.x and lower only had a single configuration directory, which was usually called **config** (this is the folder you should copy from); EDQ release 11.x and above has two configuration directories – one 'base' and one 'local'. If upgrading from EDQ release 11.x or above, copy from the 'local' configuration directory.*

*Your new EDQ environment's local configuration folder will have been created when you installed EDQ in part 1, above. By default, it will be called **oedq.local.home**, and it will be located somewhere under your new environment's domain location (For example:*

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\base_domain\config\fmwconfig\edq\oedq.local.home).

- a. Copy the entire contents of the following folders, including any sub-folders, from your old EDQ environment's local configuration folder (in EDQ releases prior to 11, there was only a single config folder, and you should copy from this) to your new EDQ environment's local configuration folder, merging them with any existing folders that have the same name:
 - i. **autorun**
 - ii. **av**
 - iii. **blueprints**
 - iv. **buckets**
 - v. **businessrules**
 - vi. **capscan**
 - vii. **commandarea**
 - viii. **dbconnectors**
 - ix. **edqp**
 - x. **extras**
 - xi. **housekeeping**
 - xii. **landingarea**
 - xiii. **localselection**
 - xiv. **localwidgets**
 - xv. **match**
 - xvi. **properties**
 - xvii. **qas**

- xviii. **resources**
- xix. **runprofiles**
- xx. **security**
- xxi. **webservices**

Note that, depending upon the EDQ version that you are migrating from, and how it has been configured, some of the folders listed above may not exist.

- b. Treat the **casemanagement** folder as follows:
 - i. Copy entire contents of the **casemanagement** folder, **except** for the indexes folder and the **flags.xml** file.
 - ii. If you **have not** made any changes to the **flags.xml** file, copy that file too.
 - iii. If you **have** made changes to the **flags.xml** file, manually copy any new extended attributes that you added, and paste them into the version of **flags.xml** that you will find in your new EDQ environment's **casemanagement** folder. However, ensure that you do not remove or modify any of the default extended attributes that you will find within your new EDQ environment's **casemanagement** folder.
- c. Treat the **localgadgets** folder as follows:
 - i. If you are migrating from EDQ release 8.1.x or above, copy entire contents of the **localgadgets** folder.
 - ii. If you are migrating from EDQ release 8.0.x or below, then copy the entire contents of the **localgadgets** folder except for any selection functions. You will need to create new versions of any tailored selection functions and place them in the **localselection** sub-folder of your new EDQ environment's local configuration folder. Existing custom selection functions will not work after upgrade.
- d. Treat the **notification** folder as follows:
 - i. If you are migrating from EDQ release 9.0.x or above, copy the entire contents of the **notification** folder.
 - ii. If you are migrating from EDQ release 8.1.x or below, copy the entire contents of the **notification** folder except for the **jobs** and **eventlog** sub-folders.

*The **jobs** and **eventlog** folders cannot be copied from EDQ release 8.1.x or lower because job template code was restructured in EDQ 9.0 and therefore modified versions of the job templates created prior to release 9.0 will not work. (If you do erroneously copy these folders, you will find that job notification e-mails will not work, and that the job notifications will not be visible in the event log.) If the contents of these folders were not modified in your old EDQ environment, then you need do nothing: the default files provided with your new EDQ environment will work out-of-the-box. But if you have carried out any modifications, then any changes you made will need to be re-applied to the new versions of the templates - copying over the old versions will not work.*

k. Treat the **triggers** folder as follows:

xxi. Check through each of the files in the triggers directory, and in any sub-directories.

If you find the following line in any file:

```
importClass( "com.datanomic.director.guihelper.users.server.UserCache" );
```

Change it to:

```
importClass( "com.datanomic.userauth.server.cache.UserCache" );
```

xxii. Once you have made any necessary changes, copy the entire contents of the **triggers** folder, including any sub-folders, from your old EDQ environment's configuration folder to your new EDQ environment's local configuration folder, merging it with any existing folder that has the same name.

l. Copy the following files from the root of your old EDQ environment's local configuration folder to the root of your new EDQ environment's local configuration folder:

xxiii. kfile

xxiv. override.properties

8. Restart your new EDQ environment's WebLogic Managed server (in other words, the EDQ application server that is called **edq_server1** unless you changed the default name when you installed EDQ). Exactly how you do this depends on your operating system and how your new EDQ environment has been configured (for example, whether you have created a script to start EDQ's Managed WebLogic server).

If you have created a script to start EDQ's Managed WebLogic server, then it is usually possible to restart it via that script.

If you have not created a script to start EDQ's Managed WebLogic server, then follow these steps:

- Ensure that the WebLogic Node Manager and the WebLogic Admin server are both running. You can stop and start the WebLogic Node Manager and the WebLogic Admin server using `cmd` Windows files or Unix/Linux shell scripts that you can find in the **bin** folder of your new EDQ environment's `edq_domain` (An example of the path to this directory on a Windows machine is:

C:\OracleMiddleware\Oracle_Home\user_projects\domains\edq_domain\bin). The relevant files will be called: **startNodeManager.cmd / startNodeManager.sh** and **startWebLogic.cmd / startWebLogic.sh**.

- Open the WebLogic console in a web browser (the default address is usually **http://your-server-name:7001/console**).

- In the Domain Structure tree, on the left-side of the screen, select **Environments > Servers** to proceed to the Summary of Servers Screen.

- In the Summary of Servers screen, navigate to the **Control** tab.

- In the Control tab, select the checkbox for the EDQ Managed Server (by default, the EDQ Managed Server is called **edq_server1**). Then, from the **Shutdown** dropdown, select **When work completes**.

- Wait for the EDQ Managed Server to Shutdown, then ensure that its checkbox is selected, and click **Start**.

9. Open a Web Browser, and navigate to your new EDQ environment's Launchpad (unless you changed any of the default options when installing EDQ, its address will be <http://your-server-name:8001/edq>).
10. From the launchpad, open the Director user interface and log in.

*Note that the user name and password for your new EDQ environment will not necessarily be the same as they were for your old EDQ environment. Unless you changed any of the default options when installing EDQ, the out-of-the-box user name for your new environment will be **weblogic**. You will have provided a password during the installation process.*

11. Verify that all your projects have been successfully migrated.
12. Run each of your jobs or standalone processes to regenerate their results (the results will not have been migrated).

Note that you should run jobs that require a run profile or run label from the Server Console user interface rather than from the Director user interface.

Note also that, if any of your processes use match review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

In some cases a job or process may fail with one of the following error messages:

Cannot load data for replacements list reference data "X". Duplicate key "Y".

Cannot load data for replacements list supplied reference data. Duplicate key "{0}".

Prior to release 11, EDQ was tolerant of reference data with duplicate lookup keys. EDQ 11 and higher is less tolerant of Reference Data with duplicate lookup keys. The errors above indicate that your migrated environment includes reference data that has duplicate lookup keys.

In this situation, Oracle recommends that you should deduplicate the reference data in question (on a case-insensitive basis). You can do this either manually if the volume of reference data is small, or by creating an EDQ process and using the Group and Merge processor if the volume of reference data is large.

*Alternatively - although it is not recommended - if previous behavior is required, you can add the following property to your new EDQ environment's **director.properties** file:*

```
replace.transform.allow.duplicate.key = true
```

*Note that the **director.properties** file is located in your new EDQ environment's local configuration directory (by default, this will be called **oedq.local.home**).*

Enterprise Data Quality has a limit of 4000 characters when writing data to the Oracle Database. When EDQ runs snapshots, it truncates fields to ensure that they are within this limit. However, in very rare cases, EDQ processes can generate strings that are longer than the character limit. Unpatched instances of Oracle Watchlist Screening 11.1.1.7 can, for example, generate very long strings that contain delimited lists of watchlist ids matched by a single customer record. When processes produce strings that are longer than the limit, they will fail. This means that it is possible that processes that ran previously under Postgres will fail on Oracle. In such cases the EDQ log would contain an error such as `ORA-01461: can bind a LONG value only for insert into a LONG column`. If you experience this error, contact Oracle support.

13. If you are using Case Management, then, from the Launchpad, open the Case Management user interface, log in and verify that all of your cases and alerts have been successfully migrated.
-

If Case Management appears to be missing cases and alerts (for example, if a filter that should return several cases and alerts in fact returns none), it is possible that the case management indexes need to be re-generated. Follow this procedure to re-generate Case Management indexes:

- Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is `C:\Program Files\Java\jdk1.8.0_40\bin`).
 - Run the `jconsole` application (how you run this will depend upon your operating system – you may simply be able to double-click the application).
 - The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Local Process** radio button, and then highlight the top-most instance of the **weblogic.server** that appears in the list of local processes (this should be EDQ's Managed WebLogic Server). Input the username for your new EDQ environment (by default, this is **weblogic**), and password. Then click **Connect**.
 - Navigate to the **Mbeans** tab.
 - Within the tree, on the left-side of the Window, expand **edq** -> **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.
 - Ensure that **clear** is set to **true**.
 - Click **reindex**. An Info dialog box will report that the Method was successfully invoked.
-

14. Within Case Management, verify that you can successfully search for and find Cases and Alerts.

*If you cannot successfully search for and find Cases and Alerts, this is likely to indicate that your flags.xml file is incorrect. The flags.xml file is located in your new EDQ environment's casemanagement directory, which is itself within the local configuration directory (called **oedq.local.home** by default). flags.xml contains information about extended attributes. A fresh EDQ installation comes with a flags.xml file that includes a number of default extended attributes. You can add new extended attributes to flags.xml, but if you removed or changed the default extended attributes that came with the original file, then EDQ will not be able to properly index Cases and Alerts, and you will therefore not be able to successfully search for them. You should resolve this by putting the default extended attributes back into flags.xml, and reindexing cases and alerts. (How to reindex is explained in the note above this one.). One way to do this is to copy the default extended attributes from a fresh version of flags.xml. You will find a fresh version of flags.xml in your new EDQ environment's base configuration folder, which is called **oedq.home** by default.*

15. Out-of-the-box, your new EDQ environment's Launchpad may not display links to all of your EDQ user interfaces. To configure the launchpad, click **Log In**, enter your EDQ user name and password, and then select **Administration > Launchpad Configuration**. You can then select the user interfaces that you want to display on the Launchpad.
16. All of the user groups from your old EDQ environment will have been migrated to your new EDQ environment. However, the users from your old environment will no longer be active. This is because in your new EDQ environment, users are managed within WebLogic rather than within EDQ itself. You will need to:
 - a. Use the WebLogic Server Administration Console to create a series of WebLogic user groups to mirror each of your EDQ user groups.

*To get to the part of the WebLogic Server Administration Console that you will require, in the Domain Structure select **Security Realms**. Then click your security realm (by default this is called **myrealm**) and navigate to the **Users and Groups** tab.*

- b. Use the WebLogic Server Administration Console to recreate all of your old EDQ users, assigning each of these users to one of the WebLogic user groups you created in the previous step.
 - c. Use the EDQ Launchpad (**Administration > External Groups**) to map each of the WebLogic user groups you created (which EDQ will see as **External Groups**) to the appropriate EDQ user group.

Upgrade Path 10:

From any Release of EDQ Using an Oracle / Tomcat Platform to EDQ 12.2.1 Using an Oracle / WebLogic Platform

Overview

Upgrading from a release of EDQ that uses the Oracle database and the Tomcat application server as its platform to EDQ 12.2.1 using the Oracle Database and Oracle WebLogic as its platform is essentially a two-part process:

- Part 1 – Install EDQ 12.2.1
- Part 2 – Migrate configuration data from your old EDQ environment to your new EDQ 12.2.1 environment. This entails using scripts provided with the EDQ 12.2.1 release to reshape and move data. It also entails manually copying some files.

Part 1: Install EDQ 12.2.1

2. Install Enterprise Data Quality 12.2.1 using the Oracle WebLogic Server as its Java Application Server and the Oracle Database as its structured query language (SQL) relational database management system (RDBMS); refer to <http://add-a-link-to-the-12.2.1-installation-doc> for more information about how to do this. During part 2, you will need to know the following information about your EDQ 12.2.1 environment:
 - a. The repository database's Host Name, Port Number and Database ID.
 - b. The schema owner and password for the config and results schemas.
 - c. The environment's domain location.

Part 2: Migrate the data from the old DIRECTOR schema to the new EDQCONFIG schema

16. Back up elements of your old EDQ environment.

During part 2, you will migrate configuration data from your old EDQ repository to your new EDQ repository, and you will also copy configuration files from your old environment to your new environment. You backup elements of your old environment (that is, the configuration data and the configuration files) so that you can easily restart this process should anything go seriously wrong.

- a. Before you back up your old EDQ environment you need to shut down EDQ's Tomcat application server. How you do this will vary by operating system.
 - i. In Windows there is usually a Windows service for the application server. You can stop this via the **Control Panel > Administrative Tools > Services**.

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the Windows service will be called **Datanomic Application Server**.*

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the Windows service will be called **Oracle EDQ Application Server**.

- ii. In Linux or Unix environments, it is likely that the application is started via an init.d service (it's quite likely you will find them in **/etc/rc.d/init.d**). The service name and method of stopping it will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name stop** from a terminal window will often work.
- b. Take a copy of your old EDQ environment's configuration folder, which is likely to be called either **config** or **oedq_local_home**.
 - i. For Windows environments, see the note below.

If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the configuration folder will be called **config**, and its default location is **C:\Program Files\Datanomic\dnDirector\config**.

If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the configuration folder will be called **oedq_local_home**. Its location will depend upon the version of Windows you have installed on. On a Windows 7 machine, for example, the default location of **oedq_local_home** is:
C:\ProgramData\Oracle\Enterprise Data Quality\oedq_local_home.

- ii. In Linux or Unix environments, there is no default location or name for the EDQ configuration folder, and you may need to ask your system administrator to tell you what the path is. Having said that, by convention, the folder is often called either **config** or **oedq.local.home** or **oedq_local_home**.
- c. Backup your old EDQ environment's database repository using the appropriate Oracle tools.
- d. **Do not** restart your old EDQ environment's application server.

In case of a serious error during the upgrade you are about to perform (for example, if the migration script should fail to complete, and you are left with a schema that is stuck in between your old version of EDQ and your new version of EDQ), you can now fairly easily start the process again from scratch. How?

- Stop your old EDQ environment's application server.
 - Delete the original versions of the folder you have backed up.
 - Copy the backup into its original location.
 - Restore the database repository from the backup.
- ...And then you can start the migration process again from scratch.
-

17. Run and examine the premigration report. This will reveal any problems with the Case Management schema (which is, strictly speaking, not a schema at all, but just a set of tables within EDQ's config schema). If any problems are revealed, you should fix them before proceeding to step 3.

- a. Open a command line tool:
 - i. In a Windows environment, open a Windows Command Prompt, running it as administrator (for example, right-click Command Prompt in the All Programs > Accessories group, and select **Run as administrator**).
 - ii. In a Unix or Linux environment, open a terminal window.
- b. Change directory to your Java Development Kit's bin folder. You can do this by issuing the command `cd path_to_the_jdk_bin_folder` (for example, in a Windows environment you might enter a command similar to: `cd C:\Program Files\Java\jdk1.8.0_40\bin`).
- c. Run the pre-migration report by issuing the following command:

```
java -jar path_to_migration.jar premigrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to run the premigration report in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar  
premigrate oracle:mydatabase@localhost:1521/director/director
```

- d. If the Premigration report reveals any problems, fix them, and run the report again before proceeding further. (If the premigration report returns no output, by the way, that's a good thing: it means no problems were found.)

Examples of Problems Revealed by the Premigration Report, With Suggested Fixes

Example One: Let's say that you see the following error in the premigration report:

```
Parameter x in Source y for Case Type z using Selection Function aa "  
" is using an attribute under input 'bb that is not mapped to case management.  
This input will not be added automatically mapped and must be manually  
configured."
```

In some earlier releases of EDQ it was possible for the value of a workflow parameter to be taken from a process attribute via a selection function even though the process attribute was not mapped to a case source attribute. For example, a workflow parameter called 'tel number' could be populated with the value of either 'cell-phone number' or 'landline number' via a First Non-empty Value selection function, even though neither 'cell-phone number' nor 'landline number' were mapped to an attribute in the case source. This is not permitted in later releases of EDQ. In order to fix this problem, you would need to create two new case source attributes and map 'cell-phone number' and 'landline number' to them. If you didn't do this, the migration would succeed, but in your upgraded system the 'tel number' parameter would not be populated, which may affect the correct workings of Case Management..

Example Two: Let's say that you see an error similar to this one in the premigration report:

```
Parameter X in Source Y for Case Type Z is defined differently across multiple  
match processes. Please make these configurations consistent.
```

In later releases of EDQ, a parameter within a case source can only be mapped to a single attribute, but this was not the case in older releases. This error message is reporting that the same case source is used in two (or more) match processes, and that the same parameter has been mapped to more than one attribute in these match processes. Before migrating the data, you should resolve this by either:

- Mapping the parameter to the same attribute in each of the match processes.
- Or creating a new parameter and mapping this to one of the attributes.

If you do not resolve this, the migration would succeed, but in your upgraded system the parameter would not be populated.

Example Three: Let's say that you see an error similar to this one in the premigration report:

```
No metadata for source XXX guid b491939a-e6fd-33d5-9752-241cbe26a378 (Code:
300,551)
Error executing script: java.lang.IllegalArgumentException
```

In older releases of EDQ, it was possible to import a Case Management configuration file (a .dxc file) that did not contain some historical structures, and this error indicates that this has been done in your old EDQ environment. If a Case Management configuration file that includes all historical case source structures in it, then importing this will solve the problem. If such a file is not available, contact Oracle Support before proceeding with your upgrade.

18. Upgrade the configuration schema.

- a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.

Note: If you are migrating from EDQ release 8.1 or below, then you will need to migrate the EDQ configuration schema in two steps:

- Step 1: migrate the configuration schema to release 9.0.
 - Step 2: migrate the configuration schema to your target 12c release.
-

- b. **Only complete this step if you are migrating from EDQ release 8.1 or below** (see the note directly above):

- i. Acquire a distribution of EDQ 9.0. (you will not need to fully install EDQ 9.0, but you will need access to its migration scripts.) The easiest way to acquire EDQ 9.0 is to download and then unzip the Windows Quick-Start installer for EDQ 9.0.
- ii. Upgrade the configuration schema to release 9.0 by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with the EDQ 9.0 distribution. This should be in `~edq-9_0_x(XXX)\9.0.x.(XXX)` (e.g. `edq-9_0_8(1103)\9.0.8.(1103)`).
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.

- password is the password of the database user.

An example of a command to migrate the configuration schema to EDQ 9.0 in a Windows environment is:

```
java -jar C:\Users\Fred\Downloads\edq-9_0_8_(1103)\9.0.8.(1103)\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

- c. Upgrade the configuration schema to your EDQ 12c release by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- path_to_migration.jar is the path to the **migration.jar** file that was installed with your new EDQ 12c environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- dbtype is postgresql for PostgreSQL or oracle for Oracle.
- dbid is the database name for PostgreSQL or the SID for Oracle.
- host is the name of the database host machine.
- port is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- user is the user ID that EDQ uses to connect to the Director database.
- password is the password of the database user.

An example of a command to migrate the configuration schema to your new EDQ 12c release in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

19. Upgrade the Case Management schema.

- Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
- Upgrade the Case Management schema by issuing the following command:

```
java -jar path_to_migration.jar cm -db dbtype:dbid@host:port/user/password
```

Where:

- path_to_migration.jar is the path to the **migration.jar** file that was installed with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- dbtype is postgresql for PostgreSQL or oracle for Oracle.

- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the Case Management schema in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar cm -db
oracle:mydatabase@localhost:1521/director/director
```

20. Migrate the upgraded configuration and Case Management data from your old EDQ environment's repository to your new EDQ environment's repository.
 - a. Shutdown your new EDQ environment's WebLogic Managed server (by default called `edq_server1`). One way to do this is from the WebLogic Server Administration Console: within the Domain Structure, expand **Environment** and select **Servers**. Then navigate to the **Control** tab, select the EDQ Managed Server (by default, this is called `edq_server1` and select **Shutdown > When work completes**).
 - b. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
 - c. Start the Database Copy tool by issuing the following command:

```
java -jar path_to_rcumigration.jar
```

Where:

- `path_to_rcumigration.jar` is the path to the **rcumigration.jar** file that was installed with your new EDQ environment. This should be in `~ORACLE_HOME\edq\oracle.edq`

An example of a command to start the Database Copy tool in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\rcumigration.jar
```

The Database Copy Tool window will open at the Welcome screen.

- d. At the welcome screen, click **Next >** to proceed to the Configuration Directories screen.
- e. At the Configuration Directories screen:
 - i. If upgrading from EDQ 9.0.x or below, leave the **Base Configuration directory location** field empty, and select the location of your old EDQ environment's

configuration directory (usually called **config**) in the **Local Configuration directory location** field.

- ii. If upgrading from EDQ 11.x or above, select your old EDQ environment's base configuration directory location (this folder is usually called either **edq_home** or **edq.home**) in the **Base Configuration directory location** field; then select the location of your old EDQ environment's local configuration directory in the **Local Configuration directory location** field (this folder is usually called either **edq_local_home** or **edq.local.home**).

The reason for this step is so that the Database Copy tool can find the connection details for your old EDQ environment's repository, as this information is stored in the configuration directory.

- f. Click **Next >** to proceed to the Source Config Database screen.
- g. The Source Config Database screen should be populated with the connection details for the config schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Source Results Database screen.
- h. The Source Results Database screen should be populated with the connection details for the results schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Destination Config Database screen.
- i. In the Destination Config Database screen, enter the connection details for your new EDQ environment's config schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Destination Results Database screen.
- j. In the Destination Results Database screen, enter the connection details for your new EDQ environment's results schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Summary screen.

Your new EDQ environment's config and results schemas were created by the Repository Creation Utility (RCU) when you installed EDQ during part 1 above. You will have supplied the RCU with the information you need now (Datasource Username, Datasource Password, HOST NAME, Port Number and Database ID). The RCU will suggest default Schema Owners of DEV_EDQCONFIG and DEV_EDQRESULTS respectively for your new EDQ environment's config and results schemas. Unless you changed these defaults, these are the values that you should now enter into the Datasource username fields.

- k. In the Summary screen, select the **Skip Empty Check** checkbox (Your new EDQ environment the destination schemas will not be empty). Also select the **DATA IN DESTINATION DATABASES WILL BE DELETED** check box (in this case, we don't want to keep any of the default data that was inserted when our new EDQ environment was created).
- l. Click **Next >**. A dialog box informs you that you are about to commence the copying of data, and that all data in the destination databases will be deleted. Click **OK**.

- m. The Database Copy Tool now migrates the configuration (and some results) data from your old EDQ environment to your new EDQ environment, reporting its progress as it does so. When it has completed this operation, the **Finished** screen is displayed. This reports that your new database is ready to use, and provides the location of database copy tool's logfile. Click **Finish**.

- 21. Manually copy some of the configuration folders and files from your old EDQ environment to your new EDQ environment.

*The location of your old EDQ environment's local configuration folder will depend upon the release of EDQ that you are migrating from, which operating system it is installed on, and how it was installed. You should already know it, as you entered it during step 5d, above. Note that EDQ release 9.x and lower only had a single configuration directory, which was usually called **config** (this is the folder you should copy from); EDQ release 11.x and above has two configuration directories – one 'base' and one 'local'. If upgrading from EDQ release 11.x or above, copy from the 'local' configuration directory.*

*Your new EDQ environment's local configuration folder will have been created when you installed EDQ in part 1, above. By default, it will be called **oedq.local.home**, and it will be located somewhere under the domain location that was created in part 1. (For example:*

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\base_domain\config\fmwconfig\edq\oedq.local.home).

- a. Copy the entire contents of the following folders, including any sub-folders, from your old EDQ environment's local configuration folder (in EDQ releases prior to 11, there was only a single config folder, and you should copy from this) to your new EDQ environment's local configuration folder, merging them with any existing folders that have the same name:
 - i. **autorun**
 - ii. **av**
 - iii. **blueprints**
 - iv. **buckets**
 - v. **businessrules**
 - vi. **capscan**
 - vii. **commandarea**
 - viii. **dbconnectors**
 - ix. **edqp**
 - x. **extras**
 - xi. **housekeeping**
 - xii. **landingarea**
 - xiii. **localselection**
 - xiv. **localwidgets**

- xv. **match**
- xvi. **properties**
- xvii. **qas**
- xviii. **resources**
- xix. **runprofiles**
- xx. **security**
- xxi. **webservices**

Note that, depending upon the EDQ version that you are migrating from, and how it has been configured, some of the folders listed above may not exist.

- b. Treat the **casemanagement** folder as follows:
 - i. Copy entire contents of the **casemanagement** folder, **except** for the indexes folder and the **flags.xml** file.
 - ii. If you **have not** made any changes to the **flags.xml** file, copy that file too.
 - iii. If you **have** made changes to the **flags.xml** file, manually copy any new extended attributes that you added, and paste them into the version of **flags.xml** that you will find in your new EDQ environment's **casemanagement** folder. However, ensure that you do not remove or modify any of the default extended attributes that you will find within your new EDQ environment's **casemanagement** folder.
- c. Treat the **localgadgets** folder as follows:
 - i. If you are migrating from EDQ release 8.1.x or above, copy entire contents of the **localgadgets** folder.
 - ii. If you are migrating from EDQ release 8.0.x or below, then copy the entire contents of the **localgadgets** folder except for any selection functions. You will need to create new versions of any tailored selection functions and place them in the **localselection** sub-folder of your new EDQ environment's local configuration folder. Existing custom selection functions will not work after upgrade.
- d. Treat the **notification** folder as follows:
 - i. If you are migrating from EDQ release 9.0.x or above, copy the entire contents of the **notification** folder.
 - ii. If you are migrating from EDQ release 8.1.x or below, copy the entire contents of the **notification** folder except for the **jobs** and **eventlog** sub-folders.

The **jobs** and **eventlog** folders cannot be copied from EDQ release 8.1.x or lower because job template code was restructured in EDQ 9.0 and therefore modified versions of the job templates created prior to release 9.0 will not work. (If you do erroneously copy these folders, you will find that job notification e-mails will not work, and that the job notifications will not be visible in the event log.) If the contents of these folders were not modified in your old EDQ environment, then you need do nothing: the default files provided with your new EDQ environment will work out-of-the-box. But if you have carried out any modifications, then any changes you made will need to be re-applied to the new versions of the templates - copying over the old versions will not work.

m. Treat the **triggers** folder as follows:

xxv. Check through each of the files in the triggers directory, and in any sub-directories.

If you find the following line in any file:

```
importClass("com.datanomic.director.guihelper.users.server.UserCache");
```

Change it to:

```
importClass("com.datanomic.userauth.server.cache.UserCache");
```

xxvi. Once you have made any necessary changes, copy the entire contents of the **triggers** folder, including any sub-folders, from your old EDQ environment's configuration folder to your new EDQ environment's local configuration folder, merging it with any existing folder that has the same name.

n. Copy the following files from the root of your old EDQ environment's local configuration folder to the root of your new EDQ environment's local configuration folder:

xxvii. kfile

xxviii. override.properties

22. Restart your new EDQ environment's WebLogic Managed server (in other words, the EDQ application server that is called **edq_server1** unless you changed the default name when you installed EDQ). Exactly how you do this depends on your operating system and how your new EDQ environment has been configured (for example, whether you have created a script to start EDQ's Managed WebLogic server).

If you have created a script to start EDQ's Managed WebLogic server, then it is usually possible to restart it via that script.

If you have not created a script to start EDQ's Managed WebLogic server, then follow these steps:

- Ensure that the WebLogic Node Manager and the WebLogic Admin server are both running. You can stop and start the WebLogic Node Manager and the WebLogic Admin server using `cmd` Windows files or Unix/Linux shell scripts that you can find in the `bin` folder of your new EDQ environment's `edq_domain` (An example of the path to this directory on a Windows machine is:

`C:\Oracle\Middleware\Oracle_Home\user_projects\domains\edq_domain\bin`). The relevant files will be called: `startNodeManager.cmd` / `startNodeManager.sh` and `startWebLogic.cmd` / `startWebLogic.sh`.

- Open the WebLogic console in a web browser (the default address is usually `http://your-server-name:7001/console`).

- In the Domain Structure tree, on the left-side of the screen, select `Environments > Servers` to proceed to the Summary of Servers Screen.

- In the Summary of Servers screen, navigate to the **Control** tab.

- In the Control tab, select the checkbox for the EDQ Managed Server (by default, the EDQ Managed Server is called `edq_server1`). Then, from the **Shutdown** dropdown, select **When work completes**.

- Wait for the EDQ Managed Server to Shutdown, then ensure that its checkbox is selected, and click **Start**.

23. Open a Web Browser, and navigate to your new EDQ environment's Launchpad (unless you changed any of the default options when installing EDQ, its address will be `http://your-server-name:8001/edq`).

24. From the launchpad, open the Director user interface and log in.

Note that the user name and password for your new EDQ environment will not necessarily be the same as they were for your old EDQ environment. Unless you changed any of the default options when installing EDQ, the out-of-the-box user name for your new environment will be **weblogic**. You will have provided a password during the installation process.

25. Verify that all your projects have been successfully migrated.

26. Run each of your jobs or standalone processes to regenerate their results (the results will not have been migrated).

Note that you should run jobs that require a run profile or run label from the Server Console user interface rather than from the Director user interface.

Note also that, if any of your processes use match review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

In some cases a job or process may fail with one of the following error messages:

`Cannot load data for replacements list reference data "X". Duplicate key "Y".`

Cannot load data for replacements list supplied reference data. Duplicate key "{0}" .

Prior to release 11, EDQ was tolerant of reference data with duplicate lookup keys. EDQ 11 and higher is less tolerant of Reference Data with duplicate lookup keys. The errors above indicate that your migrated environment includes reference data that has duplicate lookup keys.

In this situation, Oracle recommends that you should deduplicate the reference data in question (on a case-insensitive basis). You can do this either manually if the volume of reference data is small, or by creating an EDQ process and using the Group and Merge processor if the volume of reference data is large.

*Alternatively - although it is not recommended - if previous behavior is required, you can add the following property to your new EDQ environment's **director.properties** file:*

```
replace.transform.allow.duplicate.key = true
```

*Note that the **director.properties** file is located in your new EDQ environment's local configuration directory (by default, this will be called **oedq.local.home**).*

*Enterprise Data Quality has a limit of 4000 characters when writing data to the Oracle Database. When EDQ runs snapshots, it truncates fields to ensure that they are within this limit. However, in very rare cases, EDQ processes can generate strings that are longer than the character limit. Unpatched instances of Oracle Watchlist Screening 11.1.1.7 can, for example, generate very long strings that contain delimited lists of watchlist ids matched by a single customer record. When processes produce strings that are longer than the limit, they will fail. This means that it is possible that processes that ran previously under Postgres will fail on Oracle. In such cases the EDQ log would contain an error such as **ORA-01461: can bind a LONG value only for insert into a LONG column**. If you experience this error, contact Oracle support.*

27. If you are using Case Management, then, from the Launchpad, open the Case Management user interface, log in and verify that all of your cases and alerts have been successfully migrated.

If Case Management appears to be missing cases and alerts (for example, if a filter that should return several cases and alerts in fact returns none), it is possible that the case management indexes need to be re-generated. Follow this procedure to re-generate Case Management indexes:

- *Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is **C:\Program Files\Java\jdk1.8.0_40\bin**).*
 - *Run the jconsole application (how you run this will depend upon your operating system – you may simply be able to double-click the application).*
-


-
- The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Local Process** radio button, and then highlight the top-most instance of the **weblogic.server** that appears in the list of local processes (this should be EDQ's Managed WebLogic Server). Input the username for your new EDQ environment (by default, this is **weblogic**), and password. Then click **Connect**.
 - Navigate to the **Mbeans** tab.
 - Within the tree, on the left-side of the Window, expand **edq** -> **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.
 - Ensure that **clear** is set to **true**.
 - Click **reindex**. An Info dialog box will report that the Method was successfully invoked.
-

28. Within Case Management, verify that you can successfully search for and find Cases and Alerts.

*If you cannot successfully search for and find Cases and Alerts, this is likely to indicate that your flags.xml file is incorrect. The flags.xml file is located in your new EDQ environment's casemanagement directory, which is itself within the local configuration directory (called **oedq.local.home** by default). flags.xml contains information about extended attributes. A fresh EDQ installation comes with a flags.xml file that includes a number of default extended attributes. You can add new extended attributes to flags.xml, but if you removed or changed the default extended attributes that came with the original file, then EDQ will not be able to properly index Cases and Alerts, and you will therefore not be able to successfully search for them. You should resolve this by putting the default extended attributes back into flags.xml, and reindexing cases and alerts. (How to reindex is explained in the note above this one.). One way to do this is to copy the default extended attributes from a fresh version of flags.xml. You will find a fresh version of flags.xml in your new EDQ environment's base configuration folder, which is called **oedq.home** by default.*

29. Out-of-the-box, your new EDQ environment's Launchpad may not display links to all of your EDQ user interfaces. To configure the launchpad, click **Log In**, enter your EDQ user name and password, and then select **Administration > Launchpad Configuration**. You can then select the user interfaces that you want to display on the Launchpad.
30. All of the user groups from your old EDQ environment will have been migrated to your new EDQ environment. However, the users from your old environment will no longer be active. This is because in your new EDQ environment, users are managed within WebLogic rather than within EDQ itself. You will need to:
- a. Use the WebLogic Server Administration Console to create a series of WebLogic user groups to mirror each of your EDQ user groups.

*To get to the part of the WebLogic Server Administration Console that you will require, in the Domain Structure select **Security Realms**. Then click your security realm (by default this is called **myrealm**) and navigate to the **Users and Groups** tab.*

- 
- b. Use the WebLogic Server Administration Console to recreate all of your old EDQ users, assigning each of these users to one of the WebLogic user groups you created in the previous step.
 - c. Use the EDQ Launchpad (**Administration > External Groups**) to map each of the WebLogic user groups you created (which EDQ will see as **External Groups**) to the appropriate EDQ user group.

Upgrade Path 11:

From any Release of EDQ on a Postgres / Tomcat Platform to EDQ 12.2.1 Using an Oracle / Tomcat Platform

Overview

Upgrading from an installation of EDQ that uses the Postgres database and the Tomcat application server as its platform to EDQ 12.2.1 using the Oracle Database and the Tomcat application server is not recommended, but is possible. It is essentially a two-part process:

- Part 1 – Perform a Partial Installation of EDQ 12.2.1
- Part 2 – Upgrade Your Old EDQ Environment.

Note that the recommended path is to upgrade to EDQ 12.2.1 using the Oracle Database and the WebLogic application server as its platform

Part 1: Perform a Partial Installation of EDQ 12.2.1

Note that when installing EDQ it is often possible to complete the same task in different ways. For example, when you need to manipulate settings in the Oracle Database, you can usually do this through the Enterprise Manager user interface, or by issuing a SQL statement at the command line. In the instructions below, an example of one possible method of completing a task is often given. However, so long as the task is completed, you should feel free to use the approach that you are most comfortable with.

1. Ensure that your environment satisfies all EDQ 12.2.1 System Requirements, as outlined in section 1.4 of the Installing and Configuring Oracle Enterprise Data Quality guide, and upgrade your environment if necessary.
2. Download EDQ 12.2.1, as described in section 1.5 of the Installing and Configuring Oracle Enterprise Data Quality guide.
3. If your environment's current Java Development Kit does not support EDQ 12.2.1, install a Java Development Kit to support EDQ 12.2.1, as described in section 2.2 of the Installing and Configuring Oracle Enterprise Data Quality guide.
4. If your environment's current Tomcat application server does not support EDQ 12.2.1, install a release of the Tomcat Application server that does support EDQ 12.2.1.
5. Install Enterprise Data Quality 12.2.1 assets. This step entails installing the Oracle Fusion Middleware EDQ distribution. This is usually downloaded as a zip file that contains a single file with a name similar to **fmw_12.2.1.x.x_edq.jar**. Running this jar file invokes the Oracle Universal Installer.
 - s. Open a command line tool and navigate to your Java Development Kit's bin folder.
 - t. Open the Oracle Universal Installer to install Enterprise Data Quality assets by issuing the following command:

```
java -jar EDQ_installation_jar_file.jar
```

Where:

- `EDQ_installation_jar_file.jar` is the path to the Enterprise Data Quality Installation file.

An example of a command to Open the Oracle Universal Installer to install Enterprise Data Quality in a Windows environment is:

```
java -jar C:\Users\peter\Downloads\fmw_12.2.1_edq.jar
```

After some time, the Oracle Enterprise Data Quality 12c Installation window will open at the Welcome screen.

- At the Welcome screen click **Next >**.
- At the Auto Updates screen, make a choice about whether or not to auto-update (and if so how), and click **Next >**.
- At the Installation Location screen, select a directory where you would like to install EDQ, and then click **Next >**. For the purpose of this document, we will refer to this directory as **the EDQ installation folder**.

Note that, in order to install EDQ for the Tomcat application server, the EDQ installation folder must not be located within any existing Fusion Middleware home folder (if your system has one).

- At the Installation Type screen, select **Installation for Other Platforms**, and click **Next >**.
 - At the Prerequisite Checks screen, wait until all of the prerequisite are passed (if any fail, address the issues that caused them to fail, and try this step again until all prerequisite checks are passed), and then click **Next >**.
 - At the Installation Summary screen, click **Install**.
 - At the Installation Progress screen, wait until the installation has completed, and then click **Finish**.
You should find that your EDQ installation folder has been populated with a number of sub-folders.
- Install a release of the Oracle Database that will support EDQ 12.2.1.
 - EDQ requires an Oracle database to house its repository. Create this database, ensuring that it uses the AL32UTF8 (Unicode) character set, as described in section 2.4, Installing a Database to Support EDQ, of the Installing and Configuring Oracle Enterprise Data Quality guide.
 - Configure Your Oracle Database so That it is ready for Enterprise Data Quality
 - Connect to your Oracle Database.
 - Identify your database's processes and session limits. You can do this, for example, by issuing a sql statement similar to this one:

```
select resource_name, limit_value from v$resource_limit where
resource_name in ('processes','sessions');
```

- c. If the LIMIT_VALUE of either processes or sessions is less than 500, then increase them to 500. You can do this, for example, by issuing sql statements similar to one or both of the sql statements below:

```
alter system set processes = 500 scope = spfile;
alter system set sessions = 500 scope = spfile;
```

- d. If you had to increase either of the LIMIT_VALUES, you must restart the database for the new values to take effect. You can do this, for example, by issuing a sql statements similar to these:

```
shutdown immediate;
startup;
```

- e. Check the size and configuration of the Oracle database's USERS and UNDO tablespaces. You can do this, for example, with a sql statement similar to the one displayed below:

```
select
FILE_NAME, TABLESPACE_NAME, BYTES/1024/1024, MAXBYTES/1024/1024, AUTO
EXTENSIBLE from dba_data_files where TABLESPACE_NAME like 'USER%'
or TABLESPACE_NAME like 'UNDO%';
```

You can find the recommended sizes for the user and undo tablespaces in the **Installing the Required External Software Components** section of the **Oracle® Fusion Middleware Installing and Configuring Oracle Enterprise Data Quality** guide. The guide is currently located at: <https://PLEASE INSERT A LINK TO THE 12.2.1 VERSION OF THIS GUIDE>

- f. You may find that performance is improved by setting autoextend to off, and creating tablespace data files that have an appropriate permanent size. Some example sql statements are given below (the actual statements you use will depend upon the size of your existing tablespaces, where the available space on your environment is located, and so on). In the examples, we alter the size of a datafile and turn autoextend off:

```
ALTER DATABASE DATAFILE
'/u02/app/oracle/oradata/ORCL/users01.dbf' RESIZE 2048M;
```

```
ALTER DATABASE DATAFILE
'/u02/app/oracle/oradata/ORCL/users01.dbf' AUTOEXTEND OFF;
```

Before resizing tablespace datafiles, you must ensure that there is sufficient free storage in your environment. Also note that, as well as resizing the users tablespace (as illustrated in the example above), you may need to resize the undo tablespace.

- g. Ensure that your Oracle Database has an appropriately sized System Global Area (SGA) and Program Global Area (PGA). An example of how to do this using Enterprise Manager is set out below:

You can find the recommended sizes for the SGA and the PGA in the **Installing the Required External Software Components** section of the **Oracle® Fusion Middleware Installing and Configuring Oracle Enterprise Data Quality** guide. The guide is currently located at https://PLEASE_INSERT_A_LINK_TO_THE_12.2.1_VERSION_OF_THIS_GUIDE.

- i. Using a web browser navigate to your environment's Enterprise Manager console (the URL is likely to be similar to http://your_server_name:5500/em).
 - ii. Login to Enterprise Manager using your credentials, ensuring that you select the as sysdba checkbox.
 - iii. In the Configuration dropdown, select Initialization Parameters.
 - iv. In the Initialization Parameters screen, you can find a series of parameters relating to the SGA and the PGA.
 - v. If you need to change the parameters, you can do so by selecting the appropriate line and then clicking Set... Please consult your Database Administrator before making any necessary changes.
9. Create two Oracle database accounts named **edqconfig** and **edqresults** or other names of your choosing. When selecting names, consider that one schema will be the configuration schema and the other will be the results schema.
10. Grant the following privileges to each of these accounts:
 - **CREATE SESSION** (normally granted by CONNECT)
 - **CREATE INDEXTYPE** (normally granted by RESOURCE)
 - **CREATE SEQUENCE** (normally granted by RESOURCE)
 - **CREATE TABLE** (normally granted by RESOURCE)
 - **CREATE TYPE** (normally granted by RESOURCE)
 - **CREATE INDEX**
 - **DROP TABLE**
 - **DROP INDEX**
 - **UNLIMITED TABLESPACE** (recommended to allow the auto-extension of tablespace)
11. Create your new EDQ environment's configuration directories.
 - a. Create two empty directories to contain your new EDQ environment's configuration files. For the purposes of this document we will assume that you call these **oedq.home** and **oedq.local.home** respectively, though they do not necessarily have to be given those names.

For example:

On Linux and UNIX operating systems:

```
/opt/edqconfig/oedq.home
```

```
/opt/edqconfig/oedq.local.home
```

On the Windows operating system:

```
C:\edqconfig\oedq.home
```

```
C:\edqconfig\oedq.local.home
```

The first directory (**oedq.home**) is the base configuration directory and will contain the configuration files that should not be changed post-installation. The second directory (**oedq.local.home**) is the local configuration directory and will contain any custom settings that you create.

- b. Ensure that your application server user has read and write access to the two directories you create.
12. Run the EDQ Configapp to populate the configuration directories and the EDQ repository.
- i. Make certain that your Oracle database is running and open.
 - j. Open a command line tool and navigate to your Java Development Kit's bin folder.
 - k. Start the Configapp by issuing the following command:

```
java -jar path_to_configapp.jar
```

Where:

- `path_to_configapp.jar` is the path to the Configapp.

Note that the configapp is located in your EDQ installation folder's **edq/oracle.edq** directory.

An example of a command to start the Configapp in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq.jar
```

The Configapp will open.

- i. Click **Begin**.
- j. At the Configuration directory screen, leave **Create a new configuration directory** selected, and select the **oedq.home** directory that you in the previous step as the Base Configuration directory location, and select the **oedq.local.home** directory that you created above as the Local Configuration directory location. Then click **Next >**.
- k. At the Functional Pack screen, select the functional packs you want and are licensed to use, and then click **Next >** (ensure that you scroll right to the bottom of this screen to see all of the functional packs).

All Functional Packs are needed if you want to use the full capabilities of the EDQ Customer Data Services Pack on your EDQ server, such as for integration with Siebel Customer Relationship Management or Universal Customer Master.

All Functional Packs are needed if you want to install Oracle Watchlist Screening on your EDQ server.

- I. At the Configure Config repository database screen, leave the **Selection Type** set to **Properties**. Enter the user name and password for your new EDQ environment's configuration schema (which you created in a previous step). Select the option for the type of database (Oracle). Enter the host name of the system that contains the database, its port number, and the database unique id. (For Oracle the unique id is the SID not the service name. If you want to use a pluggable database, which requires a service name, you must use a raw JDBC URL or JNDI source.). Ensure that you configured the connection to the database correctly by clicking **Test**. Then click **Next >** to continue.
- m. At the Configure Results repository database screen, leave the **Selection Type** set to **Properties**. Enter the user name and password for your new EDQ environment's Results schema (which you created in a previous step). Select the option for the type of database (Oracle). Enter the host name of the system that contains the database, its port number, and the database unique id. (For Oracle the unique id is the SID not the service name. If you want to use a pluggable database, which requires a service name, you must use a raw JDBC URL or JNDI source.). Ensure that you configured the connection to the database correctly by clicking **Test**. Then click **Next >** to continue.
- ii. At the Selected options screen, Review the summary of your selections. Use **Back** to modify any of your selections, or click **Finish** to complete the configuration and continue.

In order for EDQ to function efficiently and in order for it to be able to process large volumes of data, sufficient storage space must be allocated to the tablespaces for the following Oracle database schemas:

- *EDQCONFIG*
 - *EDQRESULTS*
-

For production purposes, Oracle recommends the following minimum tablespace configuration, which in some cases may need to be extended:

- *30 GB total for the EDQCONFIG, held in a single file with autoextend turned off.*
 - *120 GB total for the EDQRESULTS, made up of four 30GB files with autoextend turned off.*
-

Part 2: Upgrade your Old EDQ Environment

1. Back up your old EDQ environment before you make any changes to it.

- i. Before you back up your old EDQ environment you need to shut down EDQ's Tomcat application server and the EDQ repository (which is held in Postgresql databases). How you do this will vary by operating system.
 - i. In Windows there are usually three Windows services: one for the application server, and one for each of the Director and Results databases. You can stop these via the **Control Panel > Administrative Tools > Services**.

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the three Windows services will be called **Datanomic Application Server**, **Datanomic dnDirector Store** and **Datanomic dnResults Store**.*

*If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the three Windows services will be called **Oracle EDQ Application Server**, **Oracle EDQ Config Store** and **Oracle EDQ Results Store** respectively.*

- ii. In Linux or Unix environments, it is likely that the application server and repository database(s) are started via init.d services (it's quite likely you will find them in **/etc/rc.d/init.d**). The service name and method of stopping them will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name stop** from a terminal window will often work.
- j. Take a copy of the folder that contains the Postgresql data, which is usually called **data**. The location and name of this folder may vary depending on your operating system and type of installation.
 - iii. For Windows environments, see the note below.

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the folder that contains the Postgresql data will be **C:\Program Files\Datanomic\dnDirector\data**.*

*If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the folder that contains the Postgresql data will depend upon which version of Windows you have installed EDQ on. In Windows 7, for example, the default location is **C:\ProgramData\Oracle\Enterprise Data Quality\data** (note that you may have to change Windows Folder Options to view hidden files and folders before you can see this location).*

- iv. In Linux or Unix environments, a common location is **/opt/PostgreSQL/version_number/data**. However, there is no single standard location, and you may need to ask your system administrator to tell you what the path is.
- k. Take a copy of your old EDQ environment's configuration folder, which is likely to be called either **config** or **oedq_local_home**.
 - v. For Windows environments, see the note below.

If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the configuration folder will be called **config**, and its default location is **C:\Program Files\Datanomic\dnDirector\config**.


If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the configuration folder will be called **oedq_local_home**. Its location will depend upon the version of Windows you have installed on. On a Windows 7 machine, for example, the default location of **oedq_local_home** is:
C:\ProgramData\Oracle\Enterprise Data Quality\oedq_local_home.

- vi. In Linux or Unix environments, there is no default location or name for the EDQ configuration folder, and you may need to ask your system administrator to tell you what the path is. Having said that, by convention, the folder is often called either **config** or **oedq.local.home** or **oedq_local_home**.
- I. Restart your old EDQ environment's Respository (that is to say, the Results and Config databases), but **do not** restart its application server.
 - vii. For Windows environments, start two of the three Windows services that you stopped above (do not restart the service for the application server though).
 - viii. In Linux or Unix environments, start the init.d service for the repository database, but do not start the init.d service for the application server. The service name and method of starting the repository database will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name start** from a terminal window will often work.

In case of a serious error during the upgrade you are about to perform (for example, if the migration script should fail to complete, and you are left with a schema that is stuck in between your old version of EDQ and your new version of EDQ), you can now fairly easily start the process again from scratch. How?

- Stop your old EDQ environment's application server and repository.
 - Delete the original versions of the folders you have backed up.
 - Copy the backups into the original locations.
 - Restart your old EDQ environment's repository.
- ...And then you can start the migration process again from scratch.
-

2. Run and examine the premigration report. This will reveal any problems with the Case Management schema (which is, strictly speaking, not a schema at all, but just a set of tables within EDQ's config schema). If any problems are revealed, you should fix them before proceeding to step 3.
 - d. Open a command line tool:
 - i. In a Windows environment, open a Windows Command Prompt, running it as administrator (for example, right-click Command Prompt in the All Programs > Accessories group, and select **Run as administrator**).
 - ii. In a Unix or Linux environment, open a terminal window.

- 
- e. Change directory to your Java Development Kit's bin folder. You can do this by issuing the command `cd path_to_the_jdk_bin_folder` (for example, in a Windows environment you might enter a command similar to: `cd C:\Program Files\Java\jdk1.8.0_40\bin`).
- f. Run the pre-migration report by issuing the following command:

```
java -jar path_to_migration.jar premigrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the migration.jar file that is delivered with your new EDQ environment. This should be in your EDQ installation folder's **edq\oracle.edq** sub-folder.
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to run the premigration report in a Windows environment is:

```
java -jar C:\EDQ_Home\edq\oracle.edq\migration.jar premigrate  
pgsql:director@localhost:9000/director/director
```

- g. If the Premigration report reveals any problems, fix them, and run the report again before proceeding further. (If the premigration report returns no output, by the way, that's a good thing: it means no problems were found.)

Examples of Problems Revealed by the Premigration Report, With Suggested Fixes

Example One: Let's say that you see the following error in the premigration report:

```
Parameter x in Source y for Case Type z using Selection Function aa "  
" is using an attribute under input 'bb that is not mapped to case management.  
This input will not be added automatically mapped and must be manually  
configured."
```

In some earlier releases of EDQ it was possible for the value of a workflow parameter to be taken from a process attribute via a selection function even though the process attribute was not mapped to a case source attribute. For example, a workflow parameter called 'tel number' could be populated with the value of either 'cell-phone number' or 'landline number' via a First Non-empty Value selection function, even though neither 'cell-phone number' nor 'landline number' were mapped to an attribute in the case source. This is not permitted in later releases of EDQ. In order to fix this problem, you would need to create two new case source attributes and map 'cell-phone number' and 'landline number' to them. If you didn't do this, the migration would succeed, but in your upgraded system the 'tel number' parameter would not be populated, which may affect the correct workings of Case Management..

Example Two: Let's say that you see an error similar to this one in the premigration report:

```
Parameter X in Source Y for Case Type Z is defined differently across multiple  
match processes. Please make these configurations consistent.
```

In later releases of EDQ, a parameter within a case source can only be mapped to a single attribute, but this was not the case in older releases. This error message is reporting that the same case source is used in two (or more) match processes, and that the same parameter has been mapped to more than one attribute in these match processes. Before migrating the data, you should resolve this by either:

- Mapping the parameter to the same attribute in each of the match processes.
- Or creating a new parameter and mapping this to one of the attributes.

If you do not resolve this, the migration would succeed, but in your upgraded system the parameter would not be populated.

Example Three: Let's say that you see an error similar to this one in the premigration report:

```
No metadata for source XXX guid b491939a-e6fd-33d5-9752-241cbe26a378 (Code:
300,551)
Error executing script: java.lang.IllegalArgumentException
```

In older releases of EDQ, it was possible to import a Case Management configuration file (a .dxc file) that did not contain some historical structures, and this error indicates that this has been done in your old EDQ environment. If a Case Management configuration file that includes all historical case source structures in it, then importing this will solve the problem. If such a file is not available, contact Oracle Support before proceeding with your upgrade.

3. Upgrade the configuration schema.
 - a. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
-

Note: If you are migrating from EDQ release 8.1 or below, then you will need to migrate the EDQ configuration schema in two steps:

- Step 1: migrate the configuration schema to release 9.0.
 - Step 2: migrate the configuration schema to your target 12c release.
-

- b. **Only complete this step if you are migrating from EDQ release 8.1 or below** (see the note directly above):

- i. Acquire a distribution of EDQ 9.0. (you will not need to fully install EDQ 9.0, but you will need access to its migration scripts.) The easiest way to acquire EDQ 9.0 is to download and then unzip the Windows Quick-Start installer for EDQ 9.0.
 - ii. Upgrade the configuration schema to release 9.0 by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with the EDQ 9.0 distribution. This should be in `~edq-9_0_x(XXX)\9.0.x.(XXX)` (e.g. `edq-9_0_8(1103)\9.0.8.(1103)`).
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.

- password is the password of the database user.

An example of a command to migrate the configuration schema to EDQ 9.0 in a Windows environment is:

```
java -jar C:\Users\Fred\Downloads\edq-9_0_8_(1103)\9.0.8.(1103)\migration.jar migrate postgres:director@localhost:9000/director/director
```

- c. Upgrade the configuration schema to your EDQ 12c release by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- path_to_migration.jar is the path to the migration.jar file that is delivered with your new EDQ environment. This should be in your EDQ installation folder's **edq\oracle.edq** sub-folder.
- dbtype is postgresql for PostgreSQL or oracle for Oracle.
- dbid is the database name for PostgreSQL or the SID for Oracle.
- host is the name of the database host machine.
- port is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- user is the user ID that EDQ uses to connect to the Director database.
- password is the password of the database user.

An example of a command to migrate the configuration schema to your new EDQ 12c release in a Windows environment is:

```
java -jar C:\EDQ_Home\edq\oracle.edq\migration.jar migrate postgres:director@localhost:9000/director/director
```

4. Upgrade the Case Management schema.

- Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
- Upgrade the Case Management schema by issuing the following command:

```
java -jar path_to_migration.jar cm -db dbtype:dbid@host:port/user/password
```

Where:

- path_to_migration. is the path to the migration.jar file that is delivered with your new EDQ environment. This should be in your EDQ installation folder's **edq\oracle.edq** sub-folder.
- dbtype is postgresql for PostgreSQL or oracle for Oracle.

- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the Case Management schema in a Windows environment is:

```
java -jar C:\EDQ_Home\edq\oracle.edq\migration.jar cm -db
postgres:director@localhost:9000/director/director
```

5. Migrate the upgraded configuration and Case Management data from your old EDQ environment's repository to your new EDQ environment's repository.
 - i. Within your command line tool, navigate to your Java Development Kit's bin folder.
 - j. Start the Database Copy tool by issuing the following command:

```
java -jar path_to_rcumigration.jar
```

Where:

- `path_to_rcumigration.jar` is the path to the **rcumigration.jar** file that is delivered with your new EDQ environment. This should be in your EDQ installation folder's **edq\oracle.edq** sub-folder.

An example of a command to start the Database Copy tool in a Windows environment is:

```
java -jar C:\EDQ_Home\edq\oracle.edq\rcumigration.jar
```

The Database Copy Tool window will open at the Welcome screen.

- k. At the welcome screen, click **Next >** to proceed to the Configuration Directories screen.
- l. At the Configuration Directories screen:
 - iii. If upgrading from EDQ 9.0.x or below, leave the **Base Configuration directory location** field empty, and select the location of your old EDQ environment's configuration directory (usually called **config**) in the **Local Configuration directory location** field.
 - iv. If upgrading from EDQ 11.x or above, select your old EDQ environment's base configuration directory location (this folder is usually called either **edq_home** or **edq.home**) in the **Base Configuration directory location** field; then select the location of your old EDQ environment's local configuration directory in the **Local**

Configuration directory location field (this folder is usually called either **edq_local_home** or **edq.local.home**).

The reason for this step is so that the Database Copy tool can find the connection details for your old EDQ environment's repository, as this information is stored in the configuration directory.

- m. Click **Next >** to proceed to the Source Config Database screen.
- n. The Source Config Database screen should be populated with the connection details for the config schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Source Results Database screen.
- o. The Source Results Database screen should be populated with the connection details for the results schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Destination Config Database screen.
- p. In the Destination Config Database screen, enter the connection details for your new EDQ environment's config schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Destination Results Database screen.
- q. In the Destination Results Database screen, enter the connection details for your new EDQ environment's results schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Summary screen.
- r. In the Summary screen, select the **Skip Empty Check** checkbox. Also select the **DATA IN DESTINATION DATABASES WILL BE DELETED** check box.
- s. Click **Next >**. A dialog box informs you that you are about to commence the copying of data, and that all data in the destination databases will be deleted. Click **OK**.
- t. The Database Copy Tool now migrates the configuration (and some results) data from your old EDQ environment to your new EDQ environment, reporting its progress as it does so. When it has completed this operation, the **Finished** screen is displayed. This reports that your new database is ready to use, and provides the location of database copy tool's logfile. Click **Finish**.

In rare cases, the database copy may fail with an error similar to this one when you migrate your config schema from a Postgres database to the Oracle database:

Copying table: dn_xtree

Record skipped as value in Column 'node_name' in Table 'dn_xtree' is too long:

??123456789A123456789A123456789A123456789A123456789A123456789A123456789A123456

The `node_name` mentioned in this error message is held in a field within EDQ's config schema. It is the name of one of the nodes that you can see within the Director user interface's Project Browser (for example, it may be the name of a project, or the name of a process, or the name of a set of staged data etc.). The length of fields in your old EDQ environment's Postgres config schema were defined in numbers of characters, some of which could be multi-byte characters. Field lengths in your new EDQ environment's Oracle database config schema are defined in numbers of bytes. The error is telling you that the node name used in your old EDQ environment is too long for your new EDQ environment. To fix this problem, you will need to open your old EDQ environment's Director user interface and, within the Project Browser, shorten the node name that has caused this error, removing any multi-byte characters if necessary. Bear in mind that you will have to revert to the backup you took during part 2, step 1, and start your old environment's application server before you can fix this problem. And once you have fixed the problem, you will then need to start part 2 of the upgrade process again.

6. Configure the Tomcat Application Server.
 - a. Create the necessary Oracle Application Development Framework (ADF) shared library:
 - i. Create a **lib.adf** ADF shared directory in your Tomcat directory. If, for example, Tomcat is installed at C:\Program Files\Apache Software Foundation\Tomcat 8.0, create a folder called **lib.adf** within that folder.
 - ii. Ensure that the operating system user who will run the EDQ application server has full permissions to the lib.adf folder.
 - iii. Unzip the **adf-essentials.zip** file, delivered with the EDQ product, into the **lib.adf** directory. The JAR files in this zip file must be in the **lib.adf** directory, not the **adf-essentials** directory that it unzips to by default. You will find the **adf-essentials.zip** file in your EDQ installation folder's **edq\adf-essentials** sub-folder.
 - iv. Configure the Tomcat `common.loader` property (typically located in the `catalina.properties` file within Tomcat's conf directory) to load Oracle ADF from their respective shared library paths. For example:

```
common.loader=${catalina.base}/lib,${catalina.base}/lib/*.jar,${catalina.home}/lib,${catalina.home}/lib/*.jar,${catalina.home}/lib.adf/*.jar
```

7. Manually copy some of the configuration folders and files from your old EDQ environment to your new EDQ environment.

The location of your old EDQ environment's local configuration folder will depend upon the release of EDQ that you are migrating from, which operating system it is installed on, and how it was installed. Note that EDQ release 9.x and lower only had a single configuration directory, which was usually called **config** (this is the folder you should copy from); EDQ release 11.x and above has two configuration directories – one 'base' and one 'local'. If upgrading from EDQ release 11.x or above, copy from the 'local' configuration directory.

Your new EDQ environment's local configuration folder will have been created when you installed EDQ in part 1, above. By default, it will be called **oedq.local.home**, and it will be located somewhere under your new environment's domain location (For example:

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\base_domain\config\fmwconfig\edq\oedq.local.home).

- h. Copy the entire contents of the following folders, including any sub-folders, from your old EDQ environment's local configuration folder (in EDQ releases prior to 11, there was only a single config folder, and you should copy from this) to your new EDQ environment's local configuration folder that you created earlier in this process (in the example used in this document, this was called, oedq.local.home) merging them with any existing folders that have the same name:
- i. **autorun**
 - ii. **av**
 - iii. **blueprints**
 - iv. **buckets**
 - v. **businessrules**
 - vi. **capscan**
 - vii. **commandarea**
 - viii. **dbconnectors**
 - ix. **edqp**
 - x. **extras**
 - xi. **housekeeping**
 - xii. **landingarea**
 - xiii. **localselection**
 - xiv. **localwidgets**
 - xv. **match**
 - xvi. **properties**
 - xvii. **qas**
 - xviii. **resources**
 - xix. **runprofiles**
 - xx. **security**
 - xxi. **webservices**

Note that, depending upon the EDQ version that you are migrating from, and how it has been configured, some of the folders listed above may not exist.

- i. Treat the **casemanagement** folder as follows:
 - i. Copy entire contents of the **casemanagement** folder, **except** for the indexes folder and the **flags.xml** file.
 - ii. If you **have not** made any changes to the **flags.xml** file, copy that file too.
 - iii. If you **have** made changes to the **flags.xml** file, manually copy any new extended attributes that you added, and paste them into the version of **flags.xml** that you will find in your new EDQ environment's **casemanagement** folder. However, ensure that you do not remove or modify any of the default extended attributes that you will find within your new EDQ environment's **casemanagement** folder.
- j. Treat the **localgadgets** folder as follows:
 - i. If you are migrating from EDQ release 8.1.x or above, copy entire contents of the **localgadgets** folder.
 - ii. If you are migrating from EDQ release 8.0.x or below, then copy the entire contents of the **localgadgets** folder except for any selection functions. You will need to create new versions of any tailored selection functions and place them in the **localselection** sub-folder of your new EDQ environment's local configuration folder. Existing custom selection functions will not work after upgrade.
- k. Treat the **notification** folder as follows:
 - i. If you are migrating from EDQ release 9.0.x or above, copy the entire contents of the **notification** folder.
 - ii. If you are migrating from EDQ release 8.1.x or below, copy the entire contents of the **notification** folder except for the **jobs** and **eventlog** sub-folders.

*The **jobs** and **eventlog** folders cannot be copied from EDQ release 8.1.x or lower because job template code was restructured in EDQ 9.0 and therefore modified versions of the job templates created prior to release 9.0 will not work. (If you do erroneously copy these folders, you will find that job notification e-mails will not work, and that the job notifications will not be visible in the event log.) If the contents of these folders were not modified in your old EDQ environment, then you need do nothing: the default files provided with your new EDQ environment will work out-of-the-box. But if you have carried out any modifications, then any changes you made will need to be re-applied to the new versions of the templates - copying over the old versions will not work.*

- o. Treat the **triggers** folder as follows:
 - xxix. Check through each of the files in the triggers directory, and in any sub-directories. If you find the following line in any file:

```
importClass("com.datanomic.director.guihelper.users.server.UserCache");
```

Change it to:

```
importClass("com.datanomic.userauth.server.cache.UserCache");
```

- xxx. Once you have made any necessary changes, copy the entire contents of the **triggers** folder, including any sub-folders, from your old EDQ environment's configuration folder to your new EDQ environment's local configuration folder, merging it with any existing folder that has the same name.
- p. Copy the following files from the root of your old EDQ environment's local configuration folder to the root of your new EDQ environment's local configuration folder:

xxxi. kfile

xxxii. override.properties

8. Deploy the EDQ application on the Tomcat application server.

- a. You must manually deploy the EDQ application on Tomcat:
 - i. Stop the Tomcat application server.
 - ii. Deploy the **edq.war** file on your application server. You will find the **edq.war** file in your EDQ installation folder's **edq\oracle.edq** sub-folder.

For more information, see the Tomcat Web Application Deployment web page at <http://tomcat.apache.org/tomcat-8.0-doc/deployer-howto.html>

- b. Link EDQ to the base and local configuration directories that you created in one of the following:
 - i. On any of the EDQ-supported platforms, you can use a Java property named - **Dedq.config.path** that specifies the paths to those directories. The syntax is as follows, where **path_to_base_config** is the path to the base configuration directory and **path_to_local_config** is the path to the local configuration directory.

Linux and UNIX (colon separates the paths):

```
-Dedq.config.path=path_to_base_config:path_to_local_config -  
Doracle.mds.cache=simple
```

Windows (semi-colon separates the paths):

```
-Dedq.config.path=path_to_base_config;path_to_local_config -  
Doracle.mds.cache=simple
```

- ii. In Windows environments only, you can append an environment variable named **EDQ_CONFIG_PATH** to your **JAVA_OPTS** environment variable, as shown in the following, where **path_to_base_config** is the path to the base configuration directory and **path_to_local_config** is the path to the local configuration directory.

```
EDQ_CONFIG_PATH="path_to_base_config;path_to_local_config"
```

For more information about setting Java parameters, see the Apache Tomcat Documentation website at <http://tomcat.apache.org/>

- c. Restart your application server service and ensure that that the EDQ application has successfully deployed.

9. Open a Web Browser, and navigate to your new EDQ environment's Launchpad (unless you changed any of the default options when installing Tomcat, its address will be port numbers when installing Tomcat its URL will be <http://your-server-name:8080/edq>).
10. From the launchpad, open the Director user interface and log in.
11. Verify that all your projects have been successfully migrated.
12. Run each of your jobs or standalone processes to regenerate their results (the results will not have been migrated).

Note that you should run jobs that require a run profile or run label from the Server Console user interface rather than from the Director user interface.

Note also that, if any of your processes use match review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

In some cases a job or process may fail with one of the following error messages:

```
Cannot load data for replacements list reference data "X". Duplicate key "Y".
```

```
Cannot load data for replacements list supplied reference data. Duplicate key "{0}" .
```

Prior to release 11, EDQ was tolerant of reference data with duplicate lookup keys. EDQ 11 and higher is less tolerant of Reference Data with duplicate lookup keys. The errors above indicate that your migrated environment includes reference data that has duplicate lookup keys.

In this situation, Oracle recommends that you should deduplicate the reference data in question (on a case-insensitive basis). You can do this either manually if the volume of reference data is small, or by creating an EDQ process and using the Group and Merge processor if the volume of reference data is large.

*Alternatively - although it is not recommended - if previous behavior is required, you can add the following property to your new EDQ environment's **director.properties** file:*

```
replace.transform.allow.duplicate.key = true
```

*Note that the **director.properties** file is located in your new EDQ environment's local configuration directory (by default, this will be called **oedq.local.home**).*


Enterprise Data Quality has a limit of 4000 characters when writing data to the Oracle Database. When EDQ runs snapshots, it truncates fields to ensure that they are within this limit. However, in very rare cases, EDQ processes can generate strings that are longer than the character limit. Unpatched instances of Oracle Watchlist Screening 11.1.1.7 can, for example, generate very long strings that contain delimited lists of watchlist ids matched by a single customer record. When processes produce strings that are longer than the limit, they will fail. This means that it is possible that processes that ran previously under Postgres will fail on Oracle. In such cases the EDQ log would contain an error such as `ORA-01461: can bind a LONG value only for insert into a LONG column`. If you experience this error, contact Oracle support.

13. If you are using Case Management, then, from the Launchpad, open the Case Management user interface, log in and verify that all of your cases and alerts have been successfully migrated.

If Case Management appears to be missing cases and alerts (for example, if a filter that should return several cases and alerts in fact returns none), it is possible that the case management indexes need to be re-generated. Follow this procedure to re-generate Case Management indexes:

- Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is `C:\Program Files\Java\jdk1.8.0_40\bin`).
 - Run the `jconsole` application (how you run this will depend upon your operating system – you may simply be able to double-click the application).
 - The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Remote Process** radio button. Enter a connection string for remote access to your Tomcat application server (`hostname:JMX_PORT_NO`), and enter your EDQ environment administration user's username (for example, `dnadmin`) and password. Then click **Connect**.
 - Navigate to the **Mbeans** tab.
 - Within the tree, on the left-side of the Window, expand `edq` - > **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.
 - Ensure that **clear** is set to **true**.
 - Click **reindex**. An Info dialog box will report that the Method was successfully invoked.
-

14. Within Case Management, verify that you can successfully search for and find Cases and Alerts.



*If you cannot successfully search for and find Cases and Alerts, this is likely to indicate that your flags.xml file is incorrect. The flags.xml file is located in your new EDQ environment's casemanagement directory, which is itself within the local configuration directory (called **oedq.local.home** by default). flags.xml contains information about extended attributes. A fresh EDQ installation comes with a flags.xml file that includes a number of default extended attributes. You can add new extended attributes to flags.xml, but if you removed or changed the default extended attributes that came with the original file, then EDQ will not be able to properly index Cases and Alerts, and you will therefore not be able to successfully search for them. You should resolve this by putting the default extended attributes back into flags.xml, and reindexing cases and alerts. (How to reindex is explained in the note above this one.). One way to do this is to copy the default extended attributes from a fresh version of flags.xml. You will find a fresh version of flags.xml in your new EDQ environment's base configuration folder, which is called **oedq.home** by default.*

Upgrade Path 12:

From any Release of EDQ on an Oracle / Tomcat Platform to EDQ 12.2.1 Using an Oracle / Tomcat Platform

Overview

Upgrading from an installation of EDQ that uses the Oracle database and the Tomcat application server as its platform to EDQ 12.2.1 using the Oracle Database and the Tomcat application server is not recommended, but is possible. It is essentially a two-part process:

- Part 1 – Perform a Partial Installation of EDQ 12.2.1
- Part 2 – Upgrade Your Old EDQ Environment.

Note that the recommended path is to upgrade to EDQ 12.2.1 using the Oracle Database and the WebLogic application server as its platform

Part 1: Perform a Partial Installation of EDQ 12.2.1

Note that when installing EDQ it is often possible to complete the same task in different ways. For example, when you need to manipulate settings in the Oracle Database, you can usually do this through the Enterprise Manager user interface, or by issuing a SQL statement at the command line. In the instructions below, an example of one possible method of completing a task is often given. However, so long as the task is completed, you should feel free to use the approach that you are most comfortable with.

17. Ensure that your environment satisfies all EDQ 12.2.1 System Requirements, as outlined in section 1.4 of the Installing and Configuring Oracle Enterprise Data Quality guide, and upgrade your environment if necessary.
18. Download EDQ 12.2.1, as described in section 1.5 of the Installing and Configuring Oracle Enterprise Data Quality guide.
19. If your environment's current Java Development Kit does not support EDQ 12.2.1, install a Java Development Kit to support EDQ 12.2.1, as described in section 2.2 of the Installing and Configuring Oracle Enterprise Data Quality guide.
20. Install Enterprise Data Quality 12.2.1 assets. This step entails installing the Oracle Fusion Middleware EDQ distribution. This is usually downloaded as a zip file that contains a single file with a name similar to **fmw_12.2.1.x.x_edq.jar**. Running this jar file invokes the Oracle Universal Installer.
 - bb. Open a command line tool and navigate to your Java Development Kit's bin folder.
 - cc. Open the Oracle Universal Installer to install Enterprise Data Quality assets by issuing the following command:

```
java -jar EDQ_installation_jar_file.jar  
Where:
```


- `EDQ_installation_jar_file.jar` is the path to the Enterprise Data Quality Installation file.

An example of a command to Open the Oracle Universal Installer to install Enterprise Data Quality in a Windows environment is:

```
java -jar C:\Users\peter\Downloads\fmw_12.2.1_edq.jar
```

After some time, the Oracle Enterprise Data Quality 12c Installation window will open at the Welcome screen.

- dd. At the Welcome screen click **Next >**.
- ee. At the Auto Updates screen, make a choice about whether or not to auto-update (and if so how), and click **Next >**.
- ff. At the Installation Location screen, select a directory where you would like to install EDQ, and then click **Next >**. For the purpose of this document, we will refer to this directory as the EDQ installation folder.

*Note that, in order to install EDQ for the Tomcat application server, the EDQ installation folder **must not be** located within any existing Fusion Middleware home folder (if your system has one).*

- gg. At the Installation Type screen, select **Installation for Other Platforms**, and click **Next >**.
 - hh. At the Prerequisite Checks screen, wait until all of the prerequisite are passed (if any fail, address the issues that caused them to fail, and try this step again until all prerequisite checks are passed), and then click **Next >**.
 - ii. At the Installation Summary screen, click **Install**.
 - jj. At the Installation Progress screen, wait until the installation has completed, and then click **Finish**.
You should find that your EDQ installation folder has been populated with a number of sub-folders.
21. Create two new Oracle database accounts named **edqconfig** and **edqresults** or other names of your choosing. When selecting names, consider that one schema will be the configuration schema and the other will be the results schema (the names you choose must obviously be different to those used in your old EDQ environment).
22. Grant the following privileges to each of these accounts:
- **CREATE SESSION** (normally granted by CONNECT)
 - **CREATE INDEXTYPE** (normally granted by RESOURCE)
 - **CREATE SEQUENCE** (normally granted by RESOURCE)
 - **CREATE TABLE** (normally granted by RESOURCE)

- **CREATE TYPE** (normally granted by RESOURCE)
- **CREATE INDEX**
- **DROP TABLE**
- **DROP INDEX**
- **UNLIMITED TABLESPACE** (recommended to allow the auto-extension of tablespace)

23. Create your new EDQ environment's configuration directories.

- a. Create two empty directories to contain your new EDQ environment's configuration files. For the purposes of this document we will assume that you call these **oedq.home** and **oedq.local.home** respectively, though they do not necessarily have to be given those names.

For example:

On Linux and UNIX operating systems:

```
/opt/edqconfig/oedq.home
```

```
/opt/edqconfig/oedq.local.home
```

On the Windows operating system:

```
C:\edqconfig\oedq.home
```

```
C:\edqconfig\oedq.local.home
```

The first directory (**oedq.home**) is the base configuration directory and will contain the configuration files that should not be changed post-installation. The second directory (**oedq.local.home**) is the local configuration directory and will contain any custom settings that you create.

- b. Ensure that your application server user has read and write access to the two directories you create.

24. Run the EDQ Configapp to populate the configuration directories and the EDQ repository.

- m. Make certain that your Oracle database is running and open.
- n. Open a command line tool and navigate to your Java Development Kit's bin folder.
- o. Start the Configapp by issuing the following command:

```
java -jar path_to_configapp.jar
```

Where:

- `path_to_configapp.jar` is the path to the Configapp.

Note that the configapp is located in your EDQ installation folder's **edq/oracle.edq** directory.

An example of a command to start the Configapp in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\configapp.jar
```

The Configapp will open.

- p. Click **Begin**.
- a. At the Configuration directory screen, leave **Create a new configuration directory** selected, and select the **oedq.home** directory that you in the previous step as the Base Configuration directory location, and select the **oedq.local.home** directory that you created above as the Local Configuration directory location. Then click **Next >**.
- b. At the Functional Pack screen, select the functional packs you want and are licensed to use, and then click **Next >** (ensure that you scroll right to the bottom of this screen to see all of the functional packs).

All Functional Packs are needed if you want to use the full capabilities of the EDQ Customer Data Services Pack on your EDQ server, such as for integration with Siebel Customer Relationship Management or Universal Customer Master.

All Functional Packs are needed if you want to install Oracle Watchlist Screening on your EDQ server.

- c. At the Configure Config repository database screen, leave the **Selection Type** set to **Properties**. Enter the user name and password for your new EDQ environment's configuration schema (which you created in a previous step). Select the option for the type of database (Oracle). Enter the host name of the system that contains the database, its port number, and the database unique id. (For Oracle the unique id is the SID not the service name. If you want to use a pluggable database, which requires a service name, you must use a raw JDBC URL or JNDI source.). Ensure that you configured the connection to the database correctly by clicking **Test**. Then click **Next >** to continue.
- d. At the Configure Results repository database screen, leave the **Selection Type** set to **Properties**. Enter the user name and password for your new EDQ environment's Results schema (which you created in a previous step). Select the option for the type of database (Oracle). Enter the host name of the system that contains the database, its port number, and the database unique id. (For Oracle the unique id is the SID not the service name. If you want to use a pluggable database, which requires a service name, you must use a raw JDBC URL or JNDI source.). Ensure that you configured the connection to the database correctly by clicking **Test**. Then click **Next >** to continue.
- iii. At the Selected options screen, Review the summary of your selections. Use **Back** to modify any of your selections, or click **Finish** to complete the configuration and continue.

In order for EDQ to function efficiently and in order for it to be able to process large volumes of data, sufficient storage space must be allocated to the tablespaces for the following Oracle database schemas:

- *EDQCONFIG*
 - *EDQRESULTS*
-

For production purposes, Oracle recommends the following minimum tablespace configuration, which in some cases may need to be extended:

- 30 GB total for the EDQCONFIG, held in a single file with autoextend turned off.
 - 120 GB total for the EDQRESULTS, made up of four 30GB files with autoextend turned off.
-

Part 2: Upgrade your Old EDQ Environment

17. Back up your old EDQ environment before you make any changes to it.
 - m. Before you back up your old EDQ environment you need to shut down EDQ's Tomcat application server. How you do this will vary by operating system.
 - iii. In Windows there is usually a Windows service for the application server. You can stop this via the **Control Panel > Administrative Tools > Services**.
-

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the Windows service will be called **Datanomic Application Server**.*

*If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the Windows service will be called **Oracle EDQ Application Server**.*

- iv. In Linux or Unix environments, it is likely that the application server is started via init.d services (it's quite likely you will find them in **/etc/rc.d/init.d**). The service name and method of stopping it will have been decided by the person who installed your old EDQ environment. However, issuing the command **service_name stop** from a terminal window will often work.
 - n. Take a copy of your old EDQ environment's configuration folder, which is likely to be called either **config** or **oedq_local_home**.
 - v. For Windows environments, see the note below.
-

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the configuration folder will be called **config**, and its default location is **C:\Program Files\Datanomic\dnDirector\config**.*

*If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the configuration folder will be called **oedq_local_home**. Its location will depend upon the version of Windows you have installed on. On a Windows 7 machine, for example, the default location of **oedq_local_home** is: **C:\ProgramData\Oracle\Enterprise Data Quality\oedq_local_home**.*

- vi. In Linux or Unix environments, there is no default location or name for the EDQ configuration folder, and you may need to ask your system administrator to tell you what the path is. Having said that, by convention, the folder is often called either **config** or **oedq.local.home** or **oedq_local_home**.
- o. Backup the two EDQ Repository schemas in your Oracle database. The names of these schemas will have been decided at installation time, and can vary. You can find their connection strings (which will include their names) in the **director.properties** file. The properties you need to look at are called **dataSource.username** and **resultsDataSource.username** respectively.

*If you your old version of EDQ is release 9.x or below, and you installed it on Windows using the **dnDirectorSetup.exe** installer and used the default settings, then the default location of the **director.properties** file is **C:\Program Files\Datanomic\dnDirector\config**.*

*If you your old version of EDQ is release 11.1.1.7 or above, and you installed it on Windows using the **edqSetup.exe** installer and used the default settings, then the **director.properties** file will be within a folder called **oedq_local_home**. Its location will depend upon the version of Windows you have installed on. On a Windows 7 machine, for example, the default location of **oedq_local_home** is:
C:\ProgramData\Oracle\Enterprise Data Quality\oedq_local_home.*

- p. **Do not** restart your old environment's application server.
18. If your environment's existing Oracle Database will not support EDQ 12.2.1, upgrade it to a release that will support EDQ 12.2.1.
19. Run and examine the premigration report. This will reveal any problems with the Case Management schema (which is, strictly speaking, not a schema at all, but just a set of tables within EDQ's config schema). If any problems are revealed, you should fix them before proceeding to step 3.
- d. Open a command line tool:
 - i. In a Windows environment, open a Windows Command Prompt, running it as administrator (for example, right-click Command Prompt in the All Programs > Accessories group, and select **Run as administrator**).
 - ii. In a Unix or Linux environment, open a terminal window.
 - e. Change directory to your Java Development Kit's bin folder. You can do this by issuing the command **cd path_to_the_jdk_bin_folder** (for example, in a Windows environment you might enter a command similar to: **cd C:\Program Files\Java\jdk1.8.0_40\bin**).
 - f. Run the pre-migration report by issuing the following command:

```
java -jar path_to_migration.jar premigrate dbtype:dbid@host:port/user/password
```

Where:
 - **path_to_migration.jar** is the path to the **migration.jar** file that is delivered with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
 - **dbtype** is **pgsql** for PostgreSQL or **oracle** for Oracle.

- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to run the premigration report in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar
premigrate oracle:mydatabase@localhost:1521/director/director
```

- g. If the Premigration report reveals any problems, fix them, and run the report again before proceeding further. (If the premigration report returns no output, by the way, that's a good thing: it means no problems were found.)

Examples of Problems Revealed by the Premigration Report, With Suggested Fixes

Example One: Let's say that you see the following error in the premigration report:

```
Parameter x in Source y for Case Type z using Selection Function aa "
" is using an attribute under input 'bb that is not mapped to case management.
This input will not be added automatically mapped and must be manually
configured."
```

In some earlier releases of EDQ it was possible for the value of a workflow parameter to be taken from a process attribute via a selection function even though the process attribute was not mapped to a case source attribute. For example, a workflow parameter called 'tel number' could be populated with the value of either 'cell-phone number' or 'landline number' via a First Non-empty Value selection function, even though neither 'cell-phone number' nor 'landline number' were mapped to an attribute in the case source. This is not permitted in later releases of EDQ. In order to fix this problem, you would need to create two new case source attributes and map 'cell-phone number' and 'landline number' to them. If you didn't do this, the migration would succeed, but in your upgraded system the 'tel number' parameter would not be populated, which may affect the correct workings of Case Management..

Example Two: Let's say that you see an error similar to this one in the premigration report:

Parameter X in Source Y for Case Type Z is defined differently across multiple match processes. Please make these configurations consistent.

In later releases of EDQ, a parameter within a case source can only be mapped to a single attribute, but this was not the case in older releases. This error message is reporting that the same case source is used in two (or more) match processes, and that the same parameter has been mapped to more than one attribute in these match processes. Before migrating the data, you should resolve this by either:

- Mapping the parameter to the same attribute in each of the match processes.
- Or creating a new parameter and mapping this to one of the attributes.

If you do not resolve this, the migration would succeed, but in your upgraded system the parameter would not be populated.

Example Three: Let's say that you see an error similar to this one in the premigration report:

No metadata for source XXX guid b491939a-e6fd-33d5-9752-241cbe26a378 (Code: 300,551)

Error executing script: java.lang.IllegalArgumentException

In older releases of EDQ, it was possible to import a Case Management configuration file (a .dxc file) that did not contain some historical structures, and this error indicates that this has been done in your old EDQ environment. If a Case Management configuration file that includes all historical case source structures in it, then importing this will solve the problem. If such a file is not available, contact Oracle Support before proceeding with your upgrade.

20. Upgrade the configuration schema.

- h. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.

Note: If you are migrating from EDQ release 8.1 or below, then you will need to migrate the EDQ configuration schema in two steps:

- Step 1: migrate the configuration schema to release 9.0.
- Step 2: migrate the configuration schema to your target 12c release.

-
- i. **Only complete this step if you are migrating from EDQ release 8.1 or below** (see the note directly above):

- i. Acquire a distribution of EDQ 9.0. (you will not need to fully install EDQ 9.0, but you will need access to its migration scripts.) The easiest way to acquire EDQ 9.0 is to download and then unzip the Windows Quick-Start installer for EDQ 9.0.
- ii. Upgrade the configuration schema to release 9.0 by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that is delivered with the EDQ 9.0 distribution. This should be in **~edq-9_0_x(XXX)\9.0.x.(XXX)** (e.g. `edq-9_0_8(1103)\9.0.8.(1103)`).
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to EDQ 9.0 in a Windows environment is:

```
java -jar C:\Users\Fred\Downloads\edq-9_0_8(1103)\9.0.8.(1103)\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

- j. Upgrade the configuration schema to your EDQ 12c release by issuing the following command:

```
java -jar path_to_migration.jar migrate dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ 12c environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the configuration schema to your new EDQ 12c release in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar migrate oracle:mydatabase@localhost:1521/director/director
```

21. Upgrade the Case Management schema.

- k. Within your command line tool, ensure that you are still within your Java Development Kit's bin folder.
- l. Upgrade the Case Management schema by issuing the following command:

```
java -jar path_to_migration.jar cm -db dbtype:dbid@host:port/user/password
```

Where:

- `path_to_migration.jar` is the path to the **migration.jar** file that was installed with your new EDQ environment. This should be in **~ORACLE_HOME\edq\oracle.edq**
- `dbtype` is `pgsql` for PostgreSQL or `oracle` for Oracle.
- `dbid` is the database name for PostgreSQL or the SID for Oracle.
- `host` is the name of the database host machine.
- `port` is optional and should be used if the database is running on a port number other than the default port number. If no value is specified, the default port for the database type is assumed.
- `user` is the user ID that EDQ uses to connect to the Director database.
- `password` is the password of the database user.

An example of a command to migrate the Case Management schema in a Windows environment is:

```
java -jar C:\Oracle\Middleware\Oracle_Home\edq\oracle.edq\migration.jar cm -db oracle:mydatabase@localhost:1521/director/director
```

22. Migrate the upgraded configuration and Case Management data from your old EDQ environment's repository to your new EDQ environment's repository.

- u. Within your command line tool, navigate to your Java Development Kit's bin folder.
- v. Start the Database Copy tool by issuing the following command:

```
java -jar path_to_rcumigration.jar
```

Where:

- `path_to_rcumigration.jar` is the path to the **rcumigration.jar** file that is delivered with your new EDQ environment. This should be in your EDQ installation folder's **edq\oracle.edq** sub-folder.

An example of a command to start the Database Copy tool in a Windows environment is:

```
java -jar C:\EDQ_Home\edq\oracle.edq\rcumigration.jar
```

The Database Copy Tool window will open at the Welcome screen.

- w. At the welcome screen, click **Next >** to proceed to the Configuration Directories screen.
- x. At the Configuration Directories screen:
- vii. If upgrading from EDQ 9.0.x or below, leave the **Base Configuration directory location** field empty, and select the location of your old EDQ environment's configuration directory (usually called **config**) in the **Local Configuration directory location** field.
- viii. If upgrading from EDQ 11.x or above, select your old EDQ environment's base configuration directory location (this folder is usually called either **edq_home** or **edq.home**) in the **Base Configuration directory location** field; then select the location of your old EDQ environment's local configuration directory in the **Local Configuration directory location** field (this folder is usually called either **edq_local_home** or **edq.local.home**).

The reason for this step is so that the Database Copy tool can find the connection details for your old EDQ environment's repository, as this information is stored in the configuration directory.


- y. Click **Next >** to proceed to the Source Config Database screen.
- z. The Source Config Database screen should be populated with the connection details for the config schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Source Results Database screen.
- aa. The Source Results Database screen should be populated with the connection details for the results schema from your old EDQ environment's repository. Click **Test** to ensure that you can successfully connect to this schema, and then click **Next >** to proceed to the Destination Config Database screen.
- bb. In the Destination Config Database screen, enter the connection details for your new EDQ environment's config schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Destination Results Database screen.
- cc. In the Destination Results Database screen, enter the connection details for your new EDQ environment's results schema and click **Test** to ensure that you can successfully connect. Then click **Next >** to proceed to the Summary screen.
- dd. In the Summary screen, select the **Skip Empty Check** checkbox. Also select the **DATA IN DESTINATION DATABASES WILL BE DELETED** check box.

- ee. Click **Next >**. A dialog box informs you that you are about to commence the copying of data, and that all data in the destination databases will be deleted. Click **OK**.
 - ff. The Database Copy Tool now migrates the configuration (and some results) data from your old EDQ environment to your new EDQ environment, reporting its progress as it does so. When it has completed this operation, the **Finished** screen is displayed. This reports that your new database is ready to use, and provides the location of database copy tool's logfile. Click **Finish**.
23. If your environment's current Tomcat application server does not support EDQ 12.2.1, install a release of the Tomcat Application server that does support EDQ 12.2.1.
24. Configure the Tomcat Application Server.
- a. Create the necessary Oracle Application Development Framework (ADF) shared library:
 - i. Create a **lib.adf** ADF shared directory in your Tomcat directory. If, for example, Tomcat is installed at C:\Program Files\Apache Software Foundation\Tomcat 8.0, create a folder called **lib.adf** within that folder.
 - ii. Ensure that the operating system user who will run the EDQ application server has full permissions to the lib.adf folder.
 - iii. Unzip the **adf-essentials.zip** file, delivered with the EDQ product, into the **lib.adf** directory. The JAR files in this zip file must be in the **lib.adf** directory, not the adf-essentials directory that it unzips to by default. You will find the **adf-essentials.zip** file in your EDQ installation folder's **edq\adf-essentials** sub-folder.
 - iv. Configure the Tomcat common.loader property (typically located in the catalina.properties file within Tomcat's conf directory) to load Oracle ADF from their respective shared library paths. For example:

```
common.loader=${catalina.base}/lib,${catalina.base}/lib/*.jar,${catalina.home}/lib,${catalina.home}/lib/*.jar,${catalina.home}/lib.adf/*.jar
```
25. Manually copy some of the configuration folders and files from your old EDQ environment to your new EDQ environment.

*The location of your old EDQ environment's local configuration folder will depend upon the release of EDQ that you are migrating from, which operating system it is installed on, and how it was installed. Note that EDQ release 9.x and lower only had a single configuration directory, which was usually called **config** (this is the folder you should copy from); EDQ release 11.x and above has two configuration directories – one 'base' and one 'local'. If upgrading from EDQ release 11.x or above, copy from the 'local' configuration directory.*

- I. Copy the entire contents of the following folders, including any sub-folders, from your old EDQ environment's local configuration folder (in EDQ releases prior to 11, there was only a single config folder, and you should copy from this) to your new EDQ environment's local configuration folder that you created earlier in this process (in the example used in this document, this was called, oedq.local.home) merging them with any existing folders that have the same name:

- 
- i. **autorun**
 - ii. **av**
 - iii. **blueprints**
 - iv. **buckets**
 - v. **businessrules**
 - vi. **capscan**
 - vii. **commandarea**
 - viii. **dbconnectors**
 - ix. **edqp**
 - x. **extras**
 - xi. **housekeeping**
 - xii. **landingarea**
 - xiii. **localselection**
 - xiv. **localwidgets**
 - xv. **match**
 - xvi. **properties**
 - xvii. **qas**
 - xviii. **resources**
 - xix. **runprofiles**
 - xx. **security**
 - xxi. **webservices**

Note that, depending upon the EDQ version that you are migrating from, and how it has been configured, some of the folders listed above may not exist.

- m. Treat the casemanagement folder as follows:
 - i. Copy entire contents of the **casemanagement** folder, **except** for the **indexes** folder and the **flags.xml** file.
 - ii. If you **have not** made any changes to the **flags.xml** file, copy that file too.
 - iii. If you **have** made changes to the **flags.xml** file, manually copy any new extended attributes that you added, and paste them into the version of **flags.xml** that you will find in your new EDQ environment's **casemanagement** folder. However, ensure that you do not remove or modify any of the default extended attributes that you will find within your new EDQ environment's **casemanagement** folder.

- n. Treat the **localgadgets** folder as follows:
 - i. If you are migrating from EDQ release 8.1.x or above, copy entire contents of the **localgadgets** folder.
 - ii. If you are migrating from EDQ release 8.0.x or below, then copy the entire contents of the **localgadgets** folder except for any selection functions. You will need to create new versions of any tailored selection functions and place them in the **localselection** sub-folder of your new EDQ environment's local configuration folder. Existing custom selection functions will not work after upgrade.
- o. Treat the **notification** folder as follows:
 - i. If you are migrating from EDQ release 9.0.x or above, copy the entire contents of the **notification** folder.
 - ii. If you are migrating from EDQ release 8.1.x or below, copy the entire contents of the **notification** folder except for the **jobs** and **eventlog** sub-folders.

*The **jobs** and **eventlog** folders cannot be copied from EDQ release 8.1.x or lower because job template code was restructured in EDQ 9.0 and therefore modified versions of the job templates created prior to release 9.0 will not work. (If you do erroneously copy these folders, you will find that job notification e-mails will not work, and that the job notifications will not be visible in the event log.) If the contents of these folders were not modified in your old EDQ environment, then you need do nothing: the default files provided with your new EDQ environment will work out-of-the-box. But if you have carried out any modifications, then any changes you made will need to be re-applied to the new versions of the templates - copying over the old versions will not work.*

- q. Treat the **triggers** folder as follows:
 - xxxiii. Check through each of the files in the triggers directory, and in any sub-directories. If you find the following line in any file:

```
importClass( "com.datanomic.director.guihelper.users.server.UserCache" );
```

Change it to:

```
importClass( "com.datanomic.userauth.server.cache.UserCache" );
```
 - xxxiv. Once you have made any necessary changes, copy the entire contents of the **triggers** folder, including any sub-folders, from your old EDQ environment's configuration folder to your new EDQ environment's local configuration folder, merging it with any existing folder that has the same name.
- r. Copy the following files from the root of your old EDQ environment's local configuration folder to the root of your new EDQ environment's local configuration folder:
 - xxxv. **kfile**
 - xxxvi. **override.properties**

26. Deploy the EDQ application on the Tomcat application server.

a. You must manually deploy the EDQ application on Tomcat:

- i. Stop the Tomcat application server.
- ii. Deploy the `edq.war` file on your application server. You will find the `edq.war` file in your EDQ installation folder's `edq\oracle.edq` sub-folder.

For more information, see the Tomcat Web Application Deployment web page at <http://tomcat.apache.org/tomcat-8.0-doc/deployer-howto.html>

b. Link EDQ to the base and local configuration directories that you created in one of the following:

- i. On any of the EDQ-supported platforms, you can use a Java property named `-Dedq.config.path` that specifies the paths to those directories. The syntax is as follows, where `path_to_base_config` is the path to the base configuration directory and `path_to_local_config` is the path to the local configuration directory.

Linux and UNIX (colon separates the paths):

```
-Dedq.config.path=path_to_base_config:path_to_local_config -  
Doracle.mds.cache=simple
```

Windows (semi-colon separates the paths):

```
-Dedq.config.path=path_to_base_config;path_to_local_config -  
Doracle.mds.cache=simple
```

- ii. In Windows environments only, you can append an environment variable named **EDQ_CONFIG_PATH** to your **JAVA_OPTS** environment variable, as shown in the following, where `path_to_base_config` is the path to the base configuration directory and `path_to_local_config` is the path to the local configuration directory.

```
EDQ_CONFIG_PATH="path_to_base_config;path_to_local_config"
```

For more information about setting Java parameters, see the Apache Tomcat Documentation website at <http://tomcat.apache.org/>

c. Restart your application server service and ensure that `edq.war` is successfully deployed.

27. Open a Web Browser, and navigate to your new EDQ environment's Launchpad (unless you changed any of the default options when installing Tomcat, its address will be port numbers when installing Tomcat its URL will be <http://your-server-name:8080/edq>).

28. From the launchpad, open the Director user interface and log in.

29. Verify that all your projects have been successfully migrated.

30. Run each of your jobs or standalone processes to regenerate their results (the results will not have been migrated).

Note that you should run jobs that require a run profile or run label from the Server Console user interface rather than from the Director user interface.

Note also that, if any of your processes use match review, you will need to run these (or jobs of which they are a part) with intelligent execution turned off before you can access Match Review. You can turn off Intelligent Execution in a job or process's Run Preferences.

In some cases a job or process may fail with one of the following error messages:

```
Cannot load data for replacements list reference data "X". Duplicate key "Y".
```

```
Cannot load data for replacements list supplied reference data. Duplicate key "{0}" .
```

Prior to release 11, EDQ was tolerant of reference data with duplicate lookup keys. EDQ 11 and higher is less tolerant of Reference Data with duplicate lookup keys. The errors above indicate that your migrated environment includes reference data that has duplicate lookup keys.

In this situation, Oracle recommends that you should deduplicate the reference data in question (on a case-insensitive basis). You can do this either manually if the volume of reference data is small, or by creating an EDQ process and using the Group and Merge processor if the volume of reference data is large.

Alternatively - although it is not recommended - if previous behavior is required, you can add the following property to your new EDQ environment's **director.properties** file:

```
replace.transform.allow.duplicate.key = true
```

Note that the **director.properties** file is located in your new EDQ environment's local configuration directory (by default, this will be called **oedq.local.home**).

31. If you are using Case Management, then, from the Launchpad, open the Case Management user interface, log in and verify that all of your cases and alerts have been successfully migrated.

If Case Management appears to be missing cases and alerts (for example, if a filter that should return several cases and alerts in fact returns none), it is possible that the case management indexes need to be re-generated. Follow this procedure to re-generate Case Management indexes:

- Navigate to your Java Development Kit's bin directory (the location of this will depend upon your operating system and whether or not you installed your JDK in the default location; an example path in a Windows environment is **C:\Program Files\Java\jdk1.8.0_40\bin**).
 - Run the `jconsole` application (how you run this will depend upon your operating system – you may simply be able to double-click the application).
-

-
- The New Connection dialog will appear within the Java Monitoring & Management Console window. Select the **Remote Process** radio button. Enter a connection string for remote access to your Tomcat application server (`hostname:JMX_PORT_NO`), and enter your EDQ environment administration user's username (for example, `dnadmin`) and password. Then click **Connect**.
 - Navigate to the **Mbeans** tab.
 - Within the tree, on the left-side of the Window, expand **edq** -> **CaseManagement** -> **Manager** -> **Operations**, and then select **reindex**.
 - Ensure that **clear** is set to **true**.
 - Click **reindex**. An Info dialog box will report that the Method was successfully invoked.
-

32. Within Case Management, verify that you can successfully search for and find Cases and Alerts.

*If you cannot successfully search for and find Cases and Alerts, this is likely to indicate that your `flags.xml` file is incorrect. The `flags.xml` file is located in your new EDQ environment's `casemanagement` directory, which is itself within the local configuration directory (called **oedq.local.home** by default). `flags.xml` contains information about extended attributes. A fresh EDQ installation comes with a `flags.xml` file that includes a number of default extended attributes. You can add new extended attributes to `flags.xml`, but if you removed or changed the default extended attributes that came with the original file, then EDQ will not be able to properly index Cases and Alerts, and you will therefore not be able to successfully search for them. You should resolve this by putting the default extended attributes back into `flags.xml`, and reindexing cases and alerts. (How to reindex is explained in the note above this one.). One way to do this is to copy the default extended attributes from a fresh version of `flags.xml`. You will find a fresh version of `flags.xml` in your new EDQ environment's base configuration folder, which is called **oedq.home** by default.*







Oracle Corporation, World Headquarters

500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries

Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

-  blogs.oracle.com/oracle
-  facebook.com/oracle
-  twitter.com/oracle
-  oracle.com

Hardware and Software, Engineered to Work Together

Copyright © 2015, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

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.0115

White Paper Title
January 2015
Author: [OPTIONAL]
Contributing Authors: [OPTIONAL]