Genesys Quality Management 8.1

Using Oracle with Genesys Quality Management

About Genesys

Genesys is the world's leading provider of customer service and contact center software - with more than 4,000 customers in 80 countries. Drawing on its more than 20 years of customer service innovation and experience, Genesys is uniquely positioned to help companies bring their people, insights and customer channels together to effectively drive today's customer conversation. Genesys software directs more than 100 million interactions every day, maximizing the value of customer engagement and differentiating the experience by driving personalization and multi-channel customer service - and extending customer service across the enterprise to optimize processes and the performance of customer-facing employees. Go to www.genesyslab.com for more information.

Each product has its own documentation for online viewing at the Genesys Technical Support website or on the Documentation Library DVD, which is available from Genesys upon request. For more information, contact your sales representative.

Notice

Although reasonable effort is made to ensure that the information in this document is complete and accurate at the time of release, Genesys Telecommunications Laboratories, Inc., cannot assume responsibility for any existing errors. Changes and/or corrections to the information contained in this document may be incorporated in future versions.

Your Responsibility for Your System's Security

You are responsible for the security of your system. Product administration to prevent unauthorized use is your responsibility. Your system administrator should read all documents provided with this product to fully understand the features available that reduce your risk of incurring charges for unlicensed use of Genesys products.

Trademarks

Genesys and the Genesys logo are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other company names and logos may be trademarks or registered trademarks of their respective holders. © 2012 Genesys Telecommunications Laboratories, Inc. All rights reserved.

The Crystal monospace font is used by permission of Software Renovation Corporation, www.SoftwareRenovation.com.

Technical Support from VARs

If you have purchased support from a value-added reseller (VAR), please contact the VAR for technical support.

Technical Support from Genesys

If you have purchased support directly from Genesys, please contact Genesys Technical Support. Before contacting technical support, please refer to the ***Genesys Care Program Guide f***or complete contact information and procedures.

Ordering and Licensing Information

Complete information on ordering and licensing Genesys products can be found in the ***Genesys Licensing Guide.***

Released by

Genesys Telecommunications Laboratories, Inc. www.genesyslab.com

Document Version: 81gqm_oracle_10-2012_v8.1.501.00

Table of Contents

Chapter

1 Introduction

This chapter provides an overview of this document, identifies the primary audience, introduces document conventions, and lists related reference information.

This chapter contains the following sections:

Document Purpose

This document covers the additional or differing processes in GQM installation and maintenance that are required when using an Oracle database instead of (or in addition to) the embedded PostgreSQL database provided by Genesys.

This document does not cover the installation, use and administration of Oracle databases – please refer to Oracle user documentation for this information.

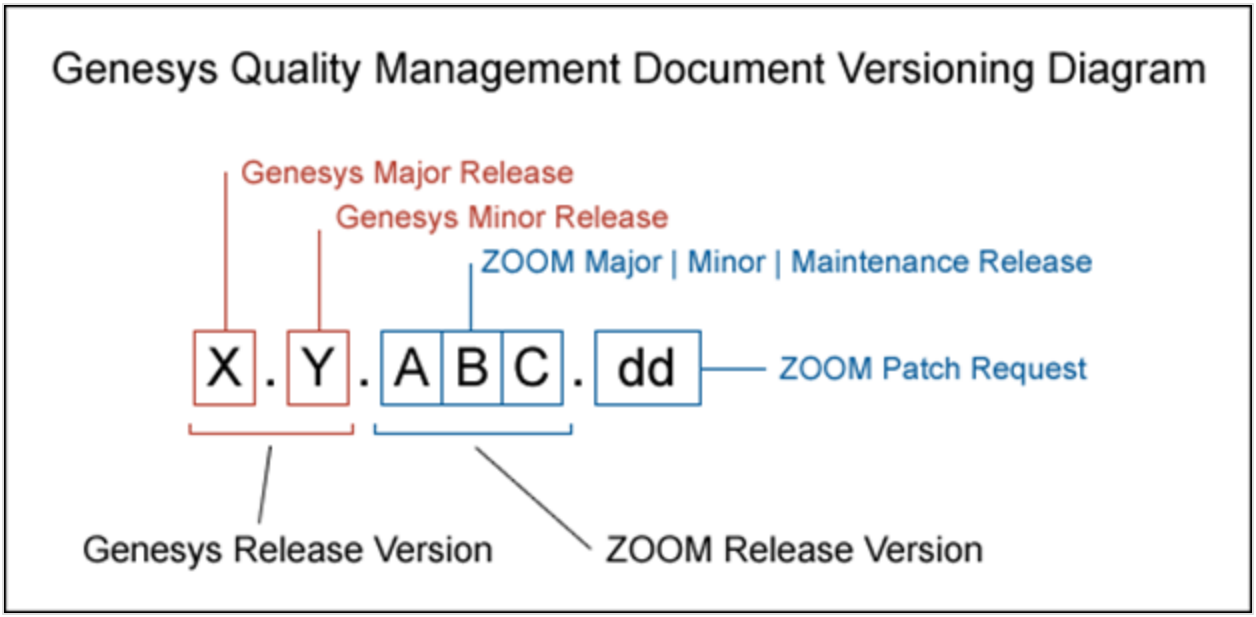
Audience

This document is intended for system engineers and administrators responsible for installation and upgrading of Genesys Quality Management.

Document Version

The Genesys Quality Management products are provided by a partnership between Genesys and ZOOM International. The Genesys Quality Management products use a versioning format that represents a combination/joining of the versions used by these two separate entities. Although the Genesys Quality Management products and documentation use this combined versioning format, in much of the software and logs you will see the ZOOM versioning alone. You need to be aware of this, for example, when communicating with Technical Support.

The version for this document is based on the structure shown in the following diagram:



Typographical Conventions

Names of functions and buttons are in bold. For example: **Upload**.

File names, file paths, command parameters and scripts launched from the command line are in `non-proportional font`.

Referred documents are in italics. For example: see the document *This is a Document* for more information.

```
Code is placed on a gray background and bordered
```

Hyperlinks are shown in blue and underlined: http://genesyslab.com/support/contact.

Expected Knowledge

Readers of this document are expected to have the following skills or knowledge:

- Basic knowledge of the Genesys Call Recording system features and functionality.
- Knowledge of RedHat Enterprise Linux installation and configuration.
- Administrative knowledge of Oracle database systems.

- Unix system administration skills.
- Network administration skills.

Chapter

2 Overview

Since version 8.0.48x of Genesys Quality Management, Oracle databases have been supported in addition to (or instead of) the embedded PostgreSQL database supplied as part of the GQM installation. Oracle databases are more suitable for GQM installations requiring high throughput and performance (such as for large numbers of call center agents and simultaneous calls), and will often be part of an enterprise database strategy, enabling more efficient corporate maintenance and backup procedures to be used.

An Oracle database can be used as the only configured database (storing all system and call data), or it can be used in addition to the embedded PostgreSQL database for specific data, such as call information. These database mappings can be modified after GQM installation, although a system restart will be required after each change.
A typical use case for mixed database deployments is a larger cluster scenario, where multiple smaller distributed recorder installations (using embedded PostgreSQL databases) provide call data to a central Oracle-powered Replay Server.

This Guide covers two main operations: deploying GQM 8.1.50x with Oracle database support, and migrating existing data between PostgreSQL and Oracle.

All Oracle-specific operations such as database installation, setup and maintenance are the responsibility of the customer; Genesys will not provide direct support for maintaining Oracle databases as we do for the embedded PostgreSQL database.

Supported Oracle Versions

GQM 8.1.50x supports Oracle database version 11g and above.

Chapter

3 GQM Installation

This section describes the additional steps that are required during GQM installation in order to use an Oracle database. For a detailed guide to installing GQM, please refer to the Implementation Guide.

This chapter contains the following sections:

Pre-install Tasks

Before beginning the GQM installation, ensure you complete the following tasks:

1. Set up access and credentials (**administrative database username & password** and optional tablespace for GQM) in a running Oracle database instance. The administrative username and password is needed during installation for the `create_schema.sh` script.

Important

The Oracle database instance used for GQM MUST have its `NLS_LANG` setting set to the following:
`AMERICAN_AMERICA.AL32UTF8`
This setting can be checked by running the following database query:

```
Select * from nls_database_parameters;
       PARAMETER                      VALUE
       ------------------------------ ----------------------------------------
       NLS_LANGUAGE                   AMERICAN
       NLS_TERRITORY                  AMERICA
       NLS_CURRENCY                   $
       NLS_ISO_CURRENCY               AMERICA
       NLS_NUMERIC_CHARACTERS         .,
       NLS_CHARACTERSET               AL32UTF8
       NLS_CALENDAR                   GREGORIAN
       NLS_DATE_FORMAT                DD-MON-RR
       NLS_DATE_LANGUAGE              AMERICAN
       NLS_SORT                       BINARY
       NLS_TIME_FORMAT                HH.MI.SSXFF AM
       NLS_TIMESTAMP_FORMAT           DD-MON-RR HH.MI.SSXFF AM
       NLS_TIME_TZ_FORMAT             HH.MI.SSXFF AM TZR
       NLS_TIMESTAMP_TZ_FORMAT        DD-MON-RR HH.MI.SSXFF AM TZR
       NLS_DUAL_CURRENCY              $
       NLS_COMP                       BINARY
       NLS_LENGTH_SEMANTICS           BYTE
       NLS_NCHAR_CONV_EXCP            FALSE
       NLS_NCHAR_CHARACTERSET         AL16UTF16
       NLS_RDBMS_VERSION              11.2.0.1.0
```

2. For any Oracle clients (for example Oracle SQL Developer) used with the GQM database schema, ensure that their host OS also has the `NLS_LANG` property set to `AL32UTF8`, which can be achieved as follows:

- On a **Unix-based host OS**, ensure the following system variable is defined:

```
NLS_LANG= AMERICAN_AMERICA.AL32UTF8
```

See Installation and Setup for an example of how to achieve this in RedHat Linux.

- On a **Windows-based host OS**, ensure the following registry key is set:
  `"NLS_LANG"=" AMERICAN_AMERICA.AL32UTF8"`
  This registry key is in the Oracle HOME registry branch, which can be found at the following locations for Oracle 11g:
  either:
  `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_OraClient11g_home1`
  or:
  `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\ORACLE\KEY_OraClient11g_home1`

Installation and Setup

This guide assumes a new installation of GQM 8.1.50x. Earlier versions of GQM must be upgraded to at least version 8.0.48x using the upgrade wizard or manual upgrade methods before the following steps can be attempted (see the Upgrade Guide).

A basic overview of installation and setup is included here – please refer to the Implementation Guide for details of the standard installation procedure.

Run Standard Installer and Setup

- Start the installer from the DVD / ISO and install the required Operating System (RHEL) as normal.
- After OS installation and a system restart, log in as root administrator and start GQM setup (`/opt/callrec/bin/callrec-setup`).

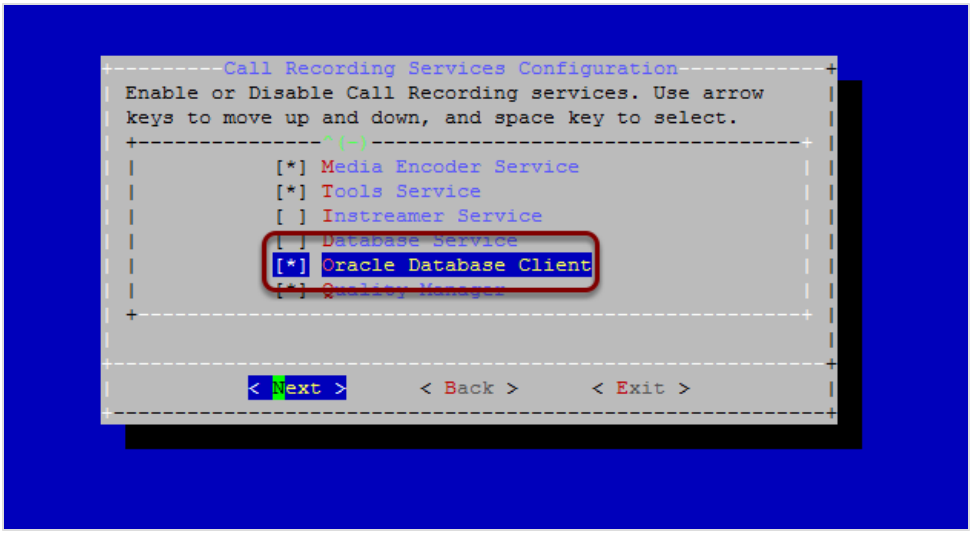


Figure 1: Selecting the Oracle Database Client

- On the services screen, select **Oracle Database Client** and unselect **Database Service** (the embedded PostgreSQL database).

Important:

If you wish to install and set up the embedded PostgreSQL database in addition to Oracle, you must run GQM setup twice; the first time selecting Database Service, and the second time selecting Oracle Database Client as described here.

After installation is complete, database pools, such as call data and Quality Manager data, can then be assigned to the different databases as appropriate – see Database Pool Mapping.

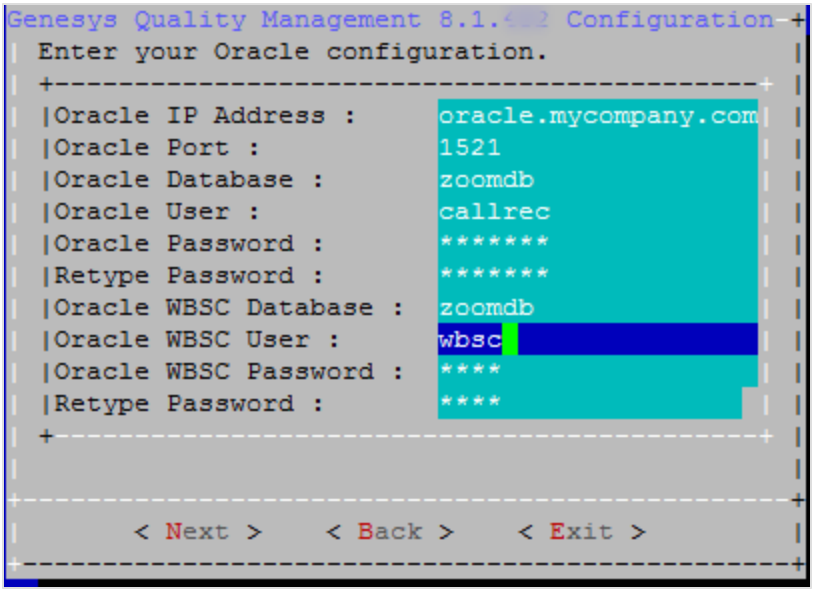


Figure 2: Oracle Database Configuration

- Enter the Oracle database credentials as follows:

1. **Oracle IP Address** (or hostname): for example `oracle.mycompany.com`
2. **Oracle Port**: default is `1521`
3. **Oracle Database** (or service name for Call Recording schema): for example `zoomdb`
4. **Oracle User** (Call Recording database user): for example `callrec`
5. **Oracle Password** (Call Recording user password): default is `callrec`
6. **Oracle WBSC Database** (or service name for Quality Manager schema): for example `zoomdb`
7. **Oracle WBSC User** (Quality Manager database user): for example `wbsc`

8. **Oracle WBSC Password** (Quality Manager user password): default is `wbsc`

Tip:

Within the Call Recording product, the term 'callrec' will often be seen, which is synonymous with this product.
Similarly, the terms 'scorecard' and 'wbsc' are synonymous with the Quality Manager product, and 'screenrec' is synonymous with the Screen Capture product.

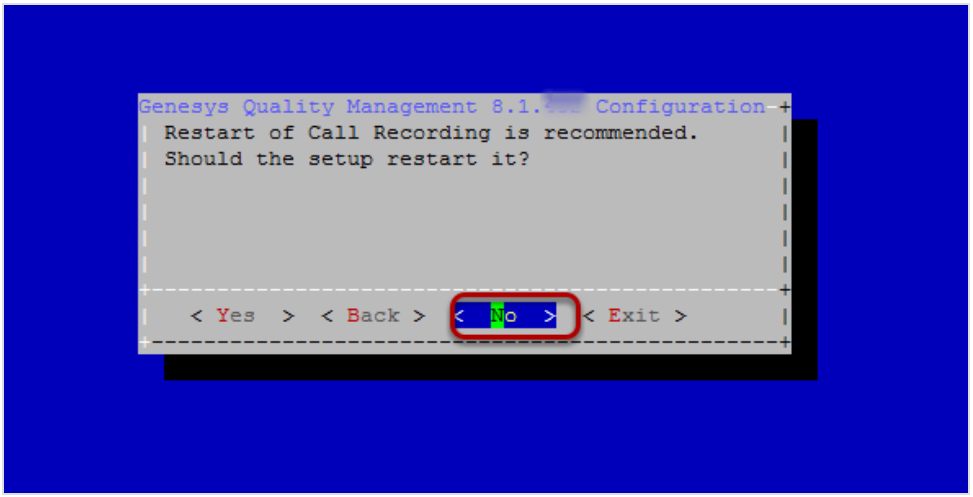


Figure 3: Do Not Restart Call Recording

- On the screen asking if you wish to restart Call Recording after completing Call Recording setup, make sure you select **No**.
- Complete the GQM setup.

Set System Variables

- After GQM setup is complete, **you must set the system variable NLS_LANG** to the following value for correct Oracle Client character set selection: `AMERICAN_AMERICA.AL32UTF8`

1. To check the current setting, ensure that you are logged in as the root user and run the following command:

```
env | grep NLS_LANG
[output:]
                        NLS_LANG=AMERICAN_AMERICA.AL32UTF8
```

2. If the command output is not the same as the above, run the following commands to set the NLS_LANG system variable:

```
echo >> ~/.bash_profile "NLS_LANG=\"AMERICAN_AMERICA.AL32UTF8\""
echo >> ~/.bash_profile "export NLS_LANG"
source ~/.bash_profile
```

Install the Database Schema

When configuring an Oracle connection for the first time, create the Call Recording and Quality Manager user schema (database tables, triggers, etc.) in the Oracle database. This is achieved in one operation by running a schema creation script in the `/opt/callrec/db_oracle_scripts/scripts` directory.

Tip:

If you wish to remove existing Call Recording and Quality Manager schema from the Oracle database, see Removing the Database Schema.

The script is available in two versions; the `create_schemas.sh` script is a Linux shell script, while the `create_schemas.bat` script is a Windows DOS script. In either case, the script must be run on a host server that has the Oracle 11g database client installed. This Oracle client software is automatically included as part of the GQM installation process, so the `create_schemas.sh`. Linux script can be run directly on the GQM server, as described here. An additional benefit of running the schema creation script on the GQM server, is that you also ensure that there is correct connectivity between the server and Oracle database.

The Linux version of the script must be run by the root user. The script's usage and parameters are as follows:

```
sh create_schemas.sh [system_user] [system_password] [database_name]
[callrec_schema_name] [wbsc_schema_name] [options]
```

The DOS version is similar to the Linux version, but `create_schemas.bat` is run without the `sh` command preceding it.

The following parameters are required:

- `system_user`: Username of database administrator account (see Pre-install Tasks).
- `system_password`: Password of database administrator account.
- `database_name`: Database name, in the form:
  `//hostname.domain.com:port/servicename`
  for example: `//oracle.mycompany.com:1521/zoomdb`.
- `callrec_schema_name`: Call Recording schema user entered as **Oracle User** earlier during GQM setup.
- `wbsc_schema_name`: Quality Manager schema user entered as **Oracle WBSC User** earlier during GQM setup.

You can also type: `sh create_schemas.sh` [without parameters] to view this parameter list.

The following options can also be specified (not required in a standard installation):

`--tbscallrec` value: name of tablespace used for Call Recording (default: `USERS`).

`--tbswbsc` value: name of tablespace used for Quality Manager (default: `USERS`).

`--temptbs` value: name of tablespace for temporary files (default: `TEMP`).

`--data Y [or] N`: create default data: user admin, roles, etc. (default: `Y`). This should normally be set to `Y` for new installations – the only case where the data is not required is when preparing a new database for migration of existing data.

`--create_admin Y [or] N`: create the user `callrec_wbsc_admin` with administrative rights for the Call Recording and Quality Manager schema (default: `N`).
This user has the following default credentials:
username: `callrec_wbsc_admin`
password: `adm`

See the following Linux example:

```
cd /opt/callrec/db_oracle_scripts/scripts
sh create_schemas.sh system sys //oracle.mycompany.com:1521/zoomdb callrec
wbsc --tbscallrec USERS --tbswbsc USERS --temptbs TEMP --data Y --create_
admin Y
```

Update Oracle Schema

After the `create_schema.sh` script is run, the new Call Recording and Quality Manager schema users have their passwords set to the **default** password values on the **Oracle configuration** screen in GQM setup. See the **Oracle User Password** and **Oracle WBSC User Password** properties in the chapter: Run Standard Installer and Setup for more details. If the default values were used, no updates are required.

- If different password values were used, reset the passwords for these Call Recording and Quality Manager schema users within Oracle.
  Please consult the Oracle documentation for how to reset database user passwords.

Start Call Recording

- After schema installation is complete, start Call Recording at the command line, ensuring that the Call Recording Core service starts (indicating correct database connection):

```
service callrec start
```

Note that some other services may not start since they are not fully configured or await license activation – see the Implementation Guide for more details.

Installation and basic setup are now complete. Configure Call Recording and Quality Manager via their respective web interfaces (see the Call Recording Administration Guide and Quality Manager Administration Guide).

Troubleshooting Database Parameters

If there are any issues in starting up, check the database parameters in `/opt/callrec/etc/core.xml`, and the error log at `/opt/callrec/logs/error.log`.

After completing GQM setup with the Oracle Database Client service activated, the core.xml file should contain database pool configuration entries similar to the following (here with the default entries used earlier):

```
    <Pool name="callrec"
poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
      <Url dbName="zoomdb" host="oracle.mycompany.com" port="1521"/>
      <Login password="callrec" userName="callrec"/>
      <Connections init="1" max="20" timeOut="5"/>
```

```
      <SpecificSetting>
        <Value name="sqlMapClass">
cz.zoom.callrec.core.callstorage.pojo.oracle.SqlMap</Value>
</SpecificSetting>
    </Pool>
    <Pool name="Maintenance"
poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
      <Url dbName="zoomdb" host="oracle.mycompany.com" port="1521"/>
      <Login password="callrec" userName="callrec"/>
      <Connections init="1" max="20" timeOut="5"/>
      <SpecificSetting>
        <Value name="sqlMapClass">
cz.zoom.callrec.tools.bean.oracle.SqlMap</Value>
      </SpecificSetting>
      </Pool>
    <Pool name="keymanager"
poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
      <Url dbName="zoomdb" host="oracle.mycompany.com" port="1521"/>
      <Login password="callrec" userName="callrec"/>
      <Connections init="1" max="20" timeOut="5"/>
      <SpecificSetting>
        <Value name="sqlMapClass">
cz.zoom.callrec.keyman.impl.pojo.oracle.SqlMap</Value>
      </SpecificSetting>
    </Pool>
    <Pool name="scorecard"
poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
      <Url dbName="zoomdb" host="oracle.mycompany.com" port="1521"/>
      <Login password="wbsc" userName="wbsc"/>
      <Connections init="1" max="20" timeOut="5"/>
      <SpecificSetting>
        <Value name="sqlMapClass">
cz.zoom.scorecard.business.data.xmlOracle.SqlMap</Value>
      </SpecificSetting>
    </Pool>
```

Modify the `dbName`, `host`, `password`, and `username` properties (for all occurrences) if required, then restart Call Recording:

```
service callrec restart
```

If there are issues with connections between Call Recording or Quality Manager to the Oracle database instance, please contact Genesys Support.

Chapter

4 Database Migration

This section covers the scripts and procedures necessary to migrate Call Recording and Quality Manager database data between the embedded PostgreSQL database and an external Oracle database; both PostgreSQL to Oracle and Oracle to PostgreSQL migration can be performed.

For customers with existing GQM deployments using PostgreSQL, the pattern of deployment and migration depends on the current version and products you have installed. See the Deployment and Migration Scenarios section.

This chapter contains the following sections:

Deployment and Migration Scenarios

The following scenarios illustrate the basic tasks necessary to accomplish a successful migration to an Oracle database for the given installed software versions. These scenarios use GQM 8.1.500 as the final target version; however the minimum target version is 8.0.48x, in order to leverage the Oracle database.

Call Recording Only (Existing version: 8.0.46x - 8.0.47x)

- Create new 8.1.500+ Installation with Oracle
  Upgrade existing Call Recording PostgreSQL database to latest minor version (using database scripts provided with official Call Recording ISO in `/opt/callrec/dbscripts/updates` directory).
- Perform Call Recording database migration of calls to 8.1.500+ (using `/opt/callrec/bin/dbmigration` script included with the 8.1.50x installation, with a correctly configured `/opt/callrec/etc/migration.xml` file for PostgreSQL to Oracle migration).

Call Recording (8.0.46x – 8.1.50x) + Quality Manager

- Create new 8.1.500+ Installation with Oracle
- Upgrade existing Call Recording PostgreSQL database to latest minor version (using database scripts provided with official Call Recording ISO in `/opt/callrec/dbscripts/updates` directory).
- Perform Call Recording database migration of calls to 8.1.500+ (using `/opt/callrec/bin/dbmigration` script included with the 8.1.50x installation).
- Upgrade existing Quality Manager PostgreSQL database to version 8.1.500+ (using `/opt/callrec/bin/scmigration2` script in the 8.1.50x installation, with a correctly configured `/opt/callrec/etc/migration.xml` file for PostgreSQL to PostgreSQL migration).
- Perform Quality Manager database migration from PostgreSQL to Oracle (using `/opt/callrec/bin/scmigration2` script in the 8.1.50x installation, with a correctly configured `/opt/callrec/etc/migration.xml` file for PostgreSQL to Oracle migration).

Migration Requirements

The following information specifies the product and database version requirements for Call Recording and Quality Manager database migration.

Important:

1. Quality Manager migration from PostgreSQL to Oracle requires a SOURCE installation of GQM 8.0.48x (or higher), due to schema incompatibilities with earlier database versions.
For Quality Manager 8.0.46x – 8.0.47x migration to Oracle, it is therefore necessary to FIRST upgrade the earlier GQM version to GQM 8.0.48x (or higher) before attempting data migration.
Please refer to the Upgrade Guide for the supported upgrade procedure.
2. Migrated Quality Manager evaluations will not be playable without separate (Call Recording) migration of the calls used in the evaluations.

Call Recording Database Migration

PostgreSQL to Oracle

Source database:

PostgreSQL database for an existing Call Recording 8.0.46x(or higher) installation (PostgreSQL 8.4 or higher is required for GQM (8.0.46x or higher) installations).

Target database:

Empty Oracle 11g (or higher) database.

Oracle to PostgreSQL

Source database:

Oracle: 11g (or higher) database for an existing GQM 8.0.48x (or higher) installation.

Target database:

Empty PostgreSQL 8.4 (or higher) database.

Quality Manager Database Migration

PostgreSQL to Oracle

Source database:

PostgreSQL database for an existing GQM 8.0.48x (or higher) installation.

Target database:

Empty Oracle 11g (or higher) database.

Oracle to PostgreSQL

Source database:

Oracle: 11g (or higher) database for an existing GQM 8.0.48x (or higher) installation.

Target database:

Empty PostgreSQL 8.4 (or higher) database.

Migration Overview

Before running the migration scripts, the target database must be empty; if any data does exist from an earlier migration, this is likely to be overwritten.

The following migration procedure is based on the migration of an existing GQM 8.1.50x installation with embedded PostgreSQL database to Oracle. A functional, empty Oracle database instance is assumed, with no pre-created Call Recording or Quality Manager schema.

The migration scripts create two separate Oracle schema for Call Recording and Quality Manager.

The entire migration process is performed at the command line, logged in as the root user with full permissions. A working knowledge of XML syntax is assumed.

Call Recording Migration

Edit the migration configuration XML file at `/opt/callrec/etc/migration.xml` as follows:

Source Database Pool

Within the `Database` node, create and insert a new database pool, representing the source ('from') database (in this case PostgreSQL), using the following code (with values for `host`, `port`, `dbName`, `username`, `password` updated appropriately):

```
<Pool name="callrec50xsource"
poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
  <Url host="localhost" port="5432" dbName="callrec"/>
  <Login userName="callrec" password="callrec"/>
  <Connections max="20" init="1" timeOut="5"/>
  <SpecificSetting>
    <Value name="sqlMapClass">
cz.zoom.callrec.tools.migration.db.version50.SqlMap</Value>
  </SpecificSetting>
</Pool>
```

Important:

Note that the `sqlMapClass` value must be correct, reflecting the correct version (version50 = Call Recording database version 8.1.500+) and database driver (PostgreSQL).

The pool names used can differ, as long as they are unique and correctly referenced later.

Target Database Pool

Create and insert a second new database pool below the first, representing the target ('to') database (in this case Oracle), using the following code (with values for `host`, `port`, `dbName`, `username`, `password` updated appropriately):

```
<Pool name="callrec50xtarget"
poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
  <Url host="oracle.mycompany.com" port="1521" dbName="zoomdb"/>
  <Login userName="callrec" password="callrec"/>
```

```
  <Connections max="20" init="1" timeOut="5"/>
  <SpecificSetting>
    <Value name="sqlMapClass">
cz.zoom.callrec.tools.migration.db.version50.oracle.SqlMap</Value>
  </SpecificSetting>
</Pool>
```

The `sqlMapClass` value must reflect the correct version and database driver. For Oracle, this value would be:
`cz.zoom.callrec.tools.migration.db.version50.oracle.SqlMap`

For PostgreSQL, this value would be the same as used for the earlier source pool, that is:
`cz.zoom.callrec.tools.migration.db.version50.SqlMap`

Source and Target Assignment

Finally, the new source and target database pools need to be correctly assigned for the migration operation. This is achieved by adding the following two nodes in the `SpecifiedConfiguration` section:

Export Node

Within the first `Group` node (with `name` value set as `export`) add the following `EqualGroup` node, ensuring the `dbPool` value reflects the source database pool name you defined earlier:

```
<EqualGroup name="export">
  <Value name="name">cr50xsource</Value>
  <Value name="dbPool">callrec50xsource</Value>
  <Value name="class">
cz.zoom.callrec.tools.migration.db.version50.ExportImpl</Value>
</EqualGroup>
```

The `class` value should again represent the correct version (8.1.500 here) and database driver (PostgreSQL here). The 8.1.500 Oracle class value would be:
`cz.zoom.callrec.tools.migration.db.version50.oracle.ExportImpl`

The name value used (`cr50xsource`) can be any permitted within XML syntax rules, and will be the export reference name used later when running the migration scripts.

Import Node

Similarly, within the second `Group` node (with name value set as imports) add the following `EqualGroup` node, ensuring the `dbPool` value reflects the target database pool name you defined earlier:

```
<EqualGroup name="import">
  <Value name="name">cr50xtarget</Value>
  <Value name="dbPool">callrec50xtarget</Value>
<Value name="class">
cz.zoom.callrec.tools.migration.db.version50.oracle.ImportImpl
</Value>
</EqualGroup>
```

Once again, ensure the correct `class`value is used (the class here representing database version 8.1.500 for the Oracle driver). The equivalent class value for the 8.1.500 PostgreSQL database driver would be:
`cz.zoom.callrec.tools.migration.db.version50.ImportImpl`

The name value used (`cr50xtarget`) can be any permitted within XML syntax rules, and will be the import reference name used later when running the migration scripts.

Run the Migration Script

After saving the migration.xml file, the Call Recording migration script can be run. This takes the following form:

```
/opt/callrec/bin/dbmigration [-config <config> | -configfile <configfile>]
[-countCRC] [-dryrun] [-export <name>] [-import <name>] [-limit <limit>] [-
logger <logger>] [-migrate <options>] [-nobind]
```

The parameters and options are as follows:

Parameter	Option(s)
`-config <config>`	URL to running configuration manager,for example `//localhost:30400/migration` Use this method OR `-configfile`
`-configfile <configfile>`	Use a configuration file, for example `/opt/callrec/etc/migration.xml` Use this method OR `-config`

Parameter	Option(s)
`-countCRC`	Check and count the CRC for each file WARNING: this will heavily impair migration performance
`-dryrun`	Test mode - don't modify files or database. Displays all operations that will be performed.
`-export <export>`	Specify the export database configuration group, for example `cr50xsource`
`-help`	display usage help
`-import <import>`	Specify the import database configuration group, for example `cr50xtarget`
`-limit <limit>`	limit number of calls processed at one time. Default value: `1000`
`-logger <logger>`	log4j properties file to define the logging properties (doesn't exist by default) for example `/opt/callrec/etc/migration.log4j.properties` Similar to all Call Recording `tool/script` log4j parameters (see similar `xxxx.log4j.properties` files in the `/opt/callrec/etc/` directory)
`-migrate <migrate>`	What to migrate – select from the following options: all – both OLD Quality Manager & Call Recording (note that this option will remove OLD Quality Manager if it exists in the target database) `callrec` – all Call Recording data `calls` – Call data `roles` – User roles
-nobind	Do not attempt to bind to the RMI registry. This option is only enabled in exceptional circumstances – normally it should be ignored. Default is to bind to RMI.

Table 1: Parameters and Options

Sample (minimal)

```
/opt/callrec/bin/dbmigration -migrate callrec -export cr50xsource -import
cr50xtarget
```

It is recommended to try a test run of the script using the `-dryrun` option (see the parameters above), before attempting a 'real' data migration.

After running the 'real' migration, use an Oracle database administration tool, such as Oracle SQL Developer or TOAD, to verify that the migration has taken place.

Quality Manager Migration

Quality Manager migration configuration is very similar to the earlier Call Recording method. Quality Manager can either be migrated from/to the same Oracle database (but different schema) as Call Recording, or from/to a completely different Oracle database. In this case, the former default scenario is used, which migrates Quality Manager from an embedded PostgreSQL database to the same Oracle database as Call Recording (but different schema).

Once again, edit the migration configuration XML file at `/opt/callrec/etc/migration.xml` as follows.

Source Database Pool

Within the `Database` node, create and insert a new database pool, representing the source ('from') Quality Manager database (in this case PostgreSQL), using the following code (with values for `host`, `port`, `dbName`, `username`, `password` updated appropriately):

```
<Pool name="scorecard50xsource"
poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
  <Url host="localhost" port="5432" dbName="callrec"/>
  <Login userName="wbsc" password="wbsc"/>
  <Connections max="20" init="1" timeOut="5"/>
  <SpecificSetting>
    <Value name="sqlMapClass">
cz.zoom.scorecard.business.data.SqlMap</Value>
  </SpecificSetting>
</Pool>
```

Note that the `sqlMapClass` value must be correct (and is different to that for the Call Recording version).

For PostgreSQL, this value would be:
`cz.zoom.scorecard.business.data.SqlMap`

For Oracle, this value would be:
`cz.zoom.scorecard.business.data.xmlOracle.SqlMap`

The pool names used can differ, as long as they are unique and correctly referenced later.

Target Database Pool

Create and insert a second new database pool below the first, representing the target ('to') Quality Manager database (in this case Oracle), using the following code (again with values for `host`, `port`, `dbName`, `username`, `password` updated appropriately):

```
<Pool name="scorecard50xtarget"
poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
  <Url host="oracle.mycompany.com" port="1521" dbName="zoomdb"/>
  <Login userName="wbsc" password="wbsc"/>
  <Connections max="20" init="1" timeOut="5"/>
  <SpecificSetting>
    <Value name="sqlMapClass">
cz.zoom.scorecard.business.data.xmlOracle.SqlMap</Value>
  </SpecificSetting>
</Pool>
```

The `sqlMapClass` value must be correct as in the sample.

For Oracle, this value would be the same as used for the earlier source pool:
`cz.zoom.scorecard.business.data.xmlOracle.SqlMap`

For PostgreSQL, this value is:
`cz.zoom.scorecard.business.data.SqlMap`

Source and Target Assignment

The new Quality Manager source and target database pools need to be correctly assigned for the migration operation. However, unlike the earlier Call Recording method, a complete new `SpecifiedConfiguration` node must be created within the `Configuration` node, which will then contain the export and import nodes.

For clarity, the whole new `SpecifiedConfiguration` node is shown below, which should be added after the first (Call Recording) `SpecifiedConfiguration` node (with name value migration), but still within the `Configuration` node.

```
<SpecifiedConfiguration name="scorecardMigration">
  <Group name="exports">
    <EqualGroup name="export" egName="sc50xsource">
      <Value name="dbPool">scorecard50xsource</Value>
      <Value name="class">cz.zoom.scorecard.migration.ExportImpl</Value>
    </EqualGroup>
```

```
  </Group>
  <Group name="imports">
    <EqualGroup name="import" egName="sc50xtarget">
      <Value name="dbPool">scorecard50xtarget</Value>
      <Value name="class">cz.zoom.scorecard.migration.ImportImpl</Value>
    </EqualGroup>
  </Group>
</SpecifiedConfiguration>
```

For the first `Group` node (with name value set as `exports`), ensure the `EqualGroup` node's `dbPool` value reflects the Quality Manager source database pool name defined earlier. Similarly, within the second `Group` node (with `name` value set as `imports`) ensure that the `EqualGroup` node's `dbPool` value reflects the Quality Manager target database pool name defined earlier.

Important:

The export and import Quality Manager `EqualGroup` configuration nodes are the same as for Call Recording, apart from two minor differences:
- The name property for `EqualGroup` nodes is here renamed to `egName`
- The class values do not change depending on database type and version

Run the Migration Script

After saving the changes made to the `migration.xml` file, the Quality Manager migration script can now be run. This takes the following form:

```
/opt/callrec/bin/scmigration2 [-config <config> | -configfile
<configfile>] [-export <name>] [-import <name>]
[-limit <limit>] [-logger <logger>] [-migrate <options>]
```

The parameters and options are as follows:

Parameter	Option(s)
-config <config>	URL to running configuration manager ,for example, //localhost:30400/migration Use this method OR -configfile
-configfile <configfile>	Use a configuration file, for example /opt/callrec/etc/migration.xml Use this method OR -config
-export <export>	Specify the export database configuration group, for example, sc50xsource
-help	display usage help
-import <import>	Specify the import database configuration group, for example sc50xtarget
-limit <limit>	limit number of evaluations processed at one time. Default value: 1000
-logger <logger>	log4j properties file to define the logging properties (doesn't exist by default) for example /opt/callrec/etc/scmigration2.log4j.properties Similar to all Call Recording tool/script log4j parameters (see similar xxxx.log4j.properties files in the /opt/callrec/etc/ directory)
-migrate <migrate>	What to migrate – select from the following options: all – all Quality Manager data (users, questionnaires, evaluation data) users – users only questforms – questionnaires only usersquestforms – users and questionnaires only **Important: Playing Evaluations** Migrated Quality Manager evaluations will not be playable without separate (Call Recording) migration of the calls used in the evaluations.

Table 2: Migration Options

Sample (minimal)

```
/opt/callrec/bin/scmigration2 -configurl //localhost:30400/migration
-export sc50xsource -import sc50xtarget -migrate all -limit 1000
```

After running the migration, use an Oracle database administration tool, such as Oracle SQL Developer or TOAD, to verify that the migration has taken place.

Chapter

5 Oracle Mapping and Maintenance

The majority of Oracle database maintenance tasks are beyond the scope of this document, and are the responsibility of the Oracle database administrator. However, the following procedures are specific to the GQM installation.

This chapter contains the following sections:

Database Pool Mapping

Database pools (such as those for call data, Quality Manager data, etc.) can be mapped to different database instances, if these are available to GQM (for example, several Oracle database instances, or both the embedded PostgreSQL database and an Oracle database instance, or other external PostgreSQL / Oracle databases, etc.).

Re-mapping database pools can be accomplished in the Call Recording Web GUI and directly in the XML configuration files. In both cases, Call Recording will need to be restarted.

Important:

Switching databases can lead to configuration data loss!
If database pools such as the main `callrec` pool are re-mapped on a configured system, any existing configuration data (such as recording rules, users and passwords) will need to be re-entered.

Call Recording Web GUI

After logging in as system administrator in the Call Recording Web GUI, navigate to the **Settings > Configuration > Call Recording Core > Database** tab.

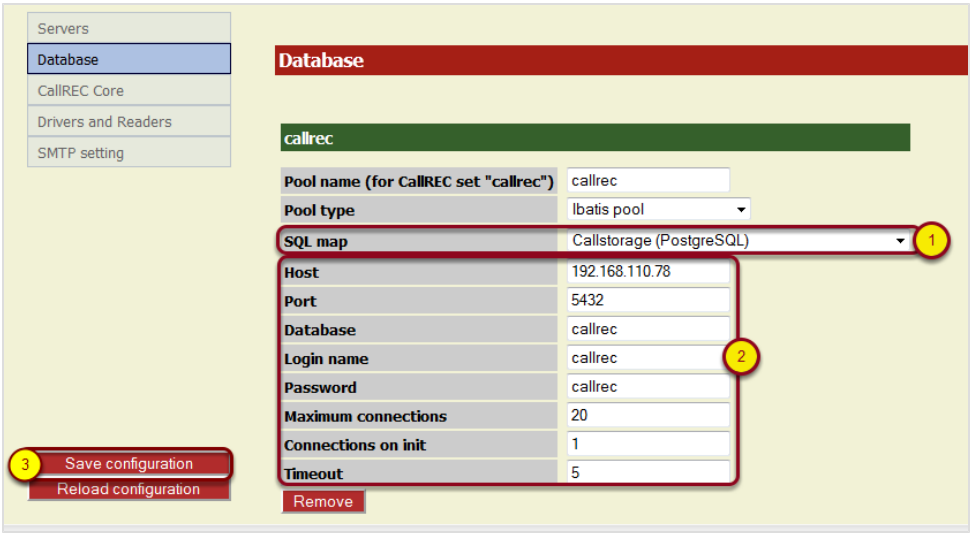


Figure 4: Database Pool Mapping in the Web GUI

1. For each database pool (for example `callrec`), select the appropriate database mapping from the dropdown **SQL map** field.
2. Update the connection details as required.
3. When complete, click **Save configuration**.

Restart Call Recording (see below for one method).

XML Configuration Files

- After logging in as the root user to the server on which the Configuration Service is running (this is the Call Recording server for single server installations), edit the `/opt/callrec/etc/core.xml` file, which contains the database pool configuration.

The following xml snippets show the main Call Recording call data pool xml for (default) Oracle and PostgreSQL mapping.

Oracle Mapping Sample:

```
<Pool name="callrec" poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
  <Url dbName="zoomdb" host="oracle.mycompany.com" port="1521"/>
  <Login password="callrec" userName="callrec"/>
  <Connections init="1" max="20" timeOut="5"/>
  <SpecificSetting>
    <Value name="sqlMapClass">
cz.zoom.callrec.core.callstorage.pojo.oracle.SqlMap</Value>
  </SpecificSetting>
</Pool>
```

PostgreSQL Mapping Sample:

```
<Pool name="callrec" poolType="cz.zoom.util.db.pool.ibatis.IbatisPool">
  <Url dbName="callrec" host="192.168.110.78" port="5432"/>
  <Login password="callrec" userName="callrec"/>
  <Connections init="1" max="20" timeOut="5"/>
  <SpecificSetting>
    <Value name="sqlMapClass">
cz.zoom.callrec.core.callstorage.pojo.SqlMap</Value>
  </SpecificSetting>
</Pool>
```

- Edit the database pool mapping (ensuring that you assign the correct `sqlMapClass` – see samples above) and save the file.

- Restart the Call Recording service by typing the command: `service callrec restart`.

Removing the Database Schema

If an attempt at installing the Call Recording and Quality Manager database schema was only partially successful, or they are no longer required in the Oracle database, remove the schema using the `drop_schemas` script (in the same scripts directory as the `create_schemas` script: `/opt/callrec/db_oracle_scripts/scripts`).

The removal script is available in two versions; the `drop_schemas.sh` script is a Linux shell script, and the `drop_schemas.bat` script is a Windows DOS script.

The script's usage and parameters are as follows (for the Linux version):

```
sh drop_schemas.sh [system_user] [system_password] [database_name]
[callrec_schema_name] [wbsc_schema_name] [options]
```

The DOS version is similar to the Linux version, but `drop_schemas.bat` is run without the `sh` command preceding it.

The following parameters are required:

- `system_user`: Username of database administrator account (see Pre-install Tasks).
- `system_password`: Password of database administrator account.
- `database_name`: Database name, in the form: `//hostname.domain.com:port/servicename` for example: `//oracle.mycompany.com:1521/zoomdb`.
- `callrec_schema_name`: Call Recording schema user; by default this is the **Oracle User** value on the Oracle parameters screen in GQM Setup.
- `wbsc_schema_name`: Quality Manager schema user; by default this is the **WBSC User** value on the Oracle parameters screen in GQM Setup.

You can also type: `sh create_schema.sh` [without parameters] to view this parameter list.

The following options can also be specified (not required in a standard installation):

`--drop_admin Y [or] N`: delete the user `callrec_wbsc_admin`. This user is created by the `create_schemas` script when the `create_admin Y` option is specified. The user has administrative rights for the Call Recording

and Quality Manager schema. See the topic Install the Database Schema for more information.

See the following Linux example:

```
cd /opt/callrec/db_oracle_scripts/scripts
sh drop_schemas.sh system sys //oracle.mycompany.com:1521/zoomdb
callrec wbsc --drop_admin Y
```

Chapter

6 Additional Reference

For Additional information about Oracle, please refer to the official Oracle user documentation at:

http://www.oracle.com/technetwork/database/enterprise-edition/documentation/index.html

Chapter

7 Request Technical Support

Technical Support from VARs

If you have purchased support from a value-added reseller (VAR), contact the VAR for technical support.

Technical Support from Genesys

If you have purchased support directly from Genesys, please contact http://genesyslab.com/support/contact Genesys Technical Support.