



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Framework Migration Guide

Updating the Database Schema

12/18/2025

Contents

- 1 Updating the Database Schema
 - 1.1 Updating the Configuration Database Schema
 - 1.2 Updating the Centralized Log Database Schema
 - 1.3 Converting the Configuration Database from Single Tenant Format

Updating the Database Schema

Management Framework Configuration Server can operate with database versions from 8.1.1 and up without the need for migration. If you are using an older version of Configuration Server, or if you want to take advantage of the latest features that require the latest database schema, you must upgrade the database schema. If you already have a compatible database schema version, you can convert to a newer version either when upgrading Configuration Server or at any time later.

If you only need to pick up the latest definitions (such as new application types, media type definitions, and so on) and you are already using Configuration Service 8.1.1 or later, you do not need to update the database schema. You only have to update the database locale to get the required definitions. See [Upgrading Configuration Definitions](#) for detailed instructions.

When upgrading the database schema, you stay with the same type of database (single-tenant or multi-tenant) and the same deployment mode (single-language or multi-language) as the original database that is being updated. Configuration Database, version 8.0 or later, can be upgraded to the latest available schema of a single-language database.

Management Framework Message Server can operate with a Centralized Log Database version 7.6 and up without needing to be migrated. Optionally, you can upgrade to the latest available version of the Centralized Log Database to increase efficiency and/or use new features.

This Migration Guide assumes that you have deployed separate databases for each product, and that you have not altered the database schema for those databases. In addition, note the following:

- Database conversion requires additional time, compares to upgrading the software, so plan accordingly.
- Genesys strongly suggests that you make a backup copy of any database being migrated, to enable you to rollback changes quickly in case any errors occur during or after the migration.

Warning

If you have changed the original database schema structure, or you use the same database to keep multiple schemas for other products, you are responsible for handling any data entered into the modified structure. Upgrade scripts may not work properly in this case. If you encounter any issues while migrating databases for which the schema was altered, or if the database is shared with other products, you might want to involve Genesys Professional Services to investigate the impact before moving forward.

Updating the Configuration Database Schema

Install the Upgrade Scripts package on the host where DBMS client software is installed and can access the target Configuration Database. You can upgrade the database schema using the database upgrade scripts directly, or, in some cases, by using the Configuration Conversion Wizard (CCW).

Using the Upgrade Scripts directly

Use the following procedure to upgrade the locale.

1. Downtime is necessary when applying the upgrade scripts. If Configuration Server is configured in HA mode, refer to [Minimize Downtime](#) while the primary Configuration Server and its backup are stopped for the upgrade. Otherwise, be prepared to stop and start the lone Configuration Server, using any hints from [Minimize Downtime](#) to minimize the downtime.
2. Use the command line (see the table below) and/or the vendor-provided User Interface to connect to the Configuration Database and execute the SQL scripts.

DBMS	Command Lines
DB2	Connect to DB2: <pre>db2 connect to %DBID% USER %USER% USING %PASSWORD% Run the script: db2 -f upg_cfg_%VER%_db2.sql</pre>
MSSQL	<pre>sqlcmd -S %HOST%\%PORT% -U %USER% -P %PASSWORD% -d %DBID% -i upg_cfg_%VER%_mssql.sql</pre>
Oracle	Connect to Oracle : <pre>sqlplus connect %USER%/%PASSWORD%@%SID% as default sqlplus>@upg_cfg_%VER%_ora.sql</pre>
Postgre	<pre>set PGPASSWORD=%PASSWORD% psql -h %HOST% -d %DBID% -U %USER% -p %PORT% -a -w -f upg_cfg_%VER%_postgre.sql</pre>
Where: %HOST% is the database server host %PORT% is the database server port %USER% is the database user name %PASSWORD% is the database user password %DBID% is the database name %VER% is the upgrade version VVV_to_TTT	

3. Use the script file in the ConfigDBMScript IP **upgrade** folder that matches the DBMS type. It is a single- or multi-language file (depending on the Configuration Server option **multi-languages**), and matches the current and target Configuration Database version.
4. Apply the upgrade script to upgrade the Configuration Database from version VVV to a next version TTT. If the current and target versions are separated by one or more intermediate versions, you must upgrade the database across those intermediate versions to get to the required target version. Apply the appropriate upgrade scripts to apply sequentially.
5. Restart Configuration Server (and its backup, if in HA mode).

Minimize Downtime

You can minimize downtime when upgrading your database by either using a backup Configuration Server, or by ensuring that the single instance of Configuration Server is running but is not connected

to the Configuration Database while you are running the upgrade scripts.

If you have a backup Configuration Server, use the following procedure. If you have only a single Configuration Server, follow steps in the procedure for a primary server. Downtime might be longer in this case, because the single server will have to be restarted and become fully operational before any client can connect.

1. Stop the backup Configuration Server.
2. Modify the configuration file of the backup Configuration Server to include the **upgrade-mode=1** option to enable side-by-side startup without contacting the configured peer server.
3. In Solution Control Server (SCS), set **disable-switchover=true** in the **[general]** section so that SCS will not automatically perform the switchover.
4. Disconnect the primary Configuration Server from the database (set **force-offline=true** in the Configuration Database section), or shut down all DB Servers that the primary server is configured to use.
5. Apply the upgrade script to the database.
6. Start the backup server and let it initialize in primary mode.
7. Stop the original primary server that is running in read-only mode. Clients will fail over to the backup server currently running in primary mode.
8. When the upgrade script is applied, reverse the previous steps, as follows:
 - a. In SCS, set **disable-switchover=false**, or remove it altogether, to restore automatic switchovers.
 - b. In the configuration file of both Configuration Servers, remove the **upgrade-mode=1** option to re-establish communication between the two servers at startup.
 - c. Restart the backup server normally.

Using CCW

If your current Configuration Database is using the 8.1.1, or later schema, you do not have to migrate your database to get new data types and enumerators. The new version of Configuration Server 8.5 will run against your current Configuration Database. Future versions of Configuration Server will include extensions to the list of application types you can utilize without migrating your system. However, if you want to update the database schema, you can use CCW. Refer to [Configuration Conversion Wizard \(CCW\)](#) for detailed information.

Tip

After upgrading the database schema to 8.5 (single language mode) and before the first start of Configuration Server, make sure that the Configuration Server configuration file contains the configuration option **langid** set to 1033 (or another appropriate value if you are applying Language Packs).

Updating the Centralized Log Database Schema

If you have upgraded your Message Servers to 8.1 or later, you might also want to update the schema of your Log Databases, as follows:

Warning

Updating a large Log Database with a large number of records can take a significantly long time.

1. For each Message Server configured on the Log Database to be updated, set the configuration option **db_storage** to false.
2. Execute the script **upgrade_7Xto80_<DBMS type>.sql** for the type of DBMS that you are using.
3. After the upgrade script has completed running, set the configuration option **db_storage** to true for those Message Servers configured on the updated database, as necessary.

Converting the Configuration Database from Single Tenant Format

Important

Use the scripts described in this section only if you are using Configuration Server 8.5.101.11 or later. In earlier Configuration Server releases, use CCW to perform the conversion, as described in [Single-tenant Configuration Database to Multi-tenant Configuration Database Format](#).

If you are using a Configuration Database in single tenant format, the Configuration Database Maintenance Scripts include a set of scripts that you can run against your single-tenant database to create your database in multi-tenant format. Each script is specific to the type of DBMS you are using, as listed in the table below.

You must stop Configuration Server before converting the database. The conversion itself must be performed off-line. Genesys also recommends that you perform the scripts with the help of Genesys Professional Services.

Warning

Do not use these scripts with any other database; they are designed to work only with

the Configuration Database. Using them with another database will not harm the target database, but might harm the Configuration Database. For more information, refer to this [Product Advisory](#).

DBMS	Script Name
DB2 See note below	upg_cfg_850ST_to_850MT_db2.sql
Microsoft SQL	upg_cfg_850ST_to_850MT_mssql.sql
Oracle	upg_cfg_850ST_to_850MT_ora.sql
PostgreSQL	upg_cfg_850ST_to_850MT_postgre.sql
Note: Genesys recommends using the DB2 Command-Line Processor to run Genesys SQL scripts.	