



DBMaker

32-bit DB transfer to 64-bit DB Operation Guide

Version: 01.00

Document No: 54/DBM546-T09232024-01-TR32

Author: Production Team

Syscom Computer Engineering CO.

Print Date: September 23, 2024

Table of Content

- 1. Introduction 1-1**
- 2. Transfer 32-bit DB to 64-bit DB 2-1**
 - 2.1. Steps in Windows System 2-1**
 - 2.2. Steps in Linux System..... 2-3**

1. Introduction

DBMaker 32-bit DB can only be used in 32-bit system, to transfer 32-bit DB to 64-bit DB. Users have to execute UNLOAD DB command to unload database schema and data, create a 64-bit database and load schema and data.

Please make sure there's no schedule or user operating database before executing UNLOAD DB command to prevent data changes that cannot be reflected in the exported data and keep database's integrity. Depends on the install environment, please notice the following situation: If 32-bit and 64-bit database are on the same machine but in different directory, need to set the path of database. The following chapters will teach users how to transfer database in Windows and Linux system.

2. Transfer 32-bit DB to 64-bit DB

2.1. Steps in Windows System

1. Use dmsqlc or dmsql32 connect to the 32-bit database with DBA/SYSDBA privilege or user SYSADM. Take DBMaker 5.4.x as example, the default directory of dmsqlc.exe and dmsql32.exe is C:\DBMaker\5.4\bin\

The CONNECT syntax is as follow:

```
CONNECT TO database_name user_name password;
```

The following example shows users how to use dmsqlc connect to DBSAMPLE5 as user SYSADM:

```
C:\DBMaker\5.4\bin>dmsqlc
dmSQL> connect to DBSAMPLE5 sysadm;
USE db #1 connected to db:<DBSAMPLE5> by user:<SYSADM>
dmSQL>
```

2. Execute UNLOAD DB command, file_name is the name of exported file, the export directory is local directory, full path is also available

The UNLOAD DB syntax is as follow:

```
UNLOAD DB TO file_name;
```

```
UNLOAD DB TO 'full_path\file_name';
```

Continue from step1, execute UNLOAD DB, the file name is DBSAMPLE5_32bit and will be exported to C:\DBMaker\unload\

```
C:\DBMaker\5.4\bin>dmsqlc
dmSQL> connect to DBSAMPLE5 sysadm;
USE db #1 connected to db:<DBSAMPLE5> by user:<SYSADM>
dmSQL> unload db to 'C:\DBMaker\unload\DSAMPLE5_32bit';
dmSQL>
```

3. After step2 is completed, there will be at least 2 files generated in C:\DBMaker\unload\: .s0 file stores database schema; .b0 file stores database data



After UNLOAD DB is completed, execute TERMINATE DB to terminate database, then execute QUIT to close dmsqlc.

```
C:\DBMaker\5.4\bin>dmsqlc
dmSQL> connect to DBSAMPLE5 sysadm;
USE db #1 connected to db:<DBSAMPLE5> by user:<SYSADM>
dmSQL> unload db to 'C:\DBMaker\unload\DSAMPLE5_32bit';
dmSQL> terminate db;
USE db #1 terminated(CURRENT)
dmSQL> quit;
C:\DBMaker\5.4\bin>
```

4. Backup or delete the 32-bit database (DB_DBDir directory). If database has file objects, they will also be unloaded to the directory set in the UNLOAD DB command. If users have created stored procedure in the database, stored procedures' definition will be exported to .b0. User defined function will not be unloaded, users have to manually move the original file and create it in the new database.
5. Install DBMaker 64-bit on a 64-bit platform.
6. Copy 32-bit database's dmconfig.ini to 64-bit. Take DBMaker 5.4.x as example, the default directory of dmconfig.ini will be C:\DBMaker\5.4\. If users use bundle version, the default directory of dmconfig.ini will be in the extracted file.

Notice the DB_DBDir and DB_FoDir in dmconfig.ini, the directory must exist in the new machine, create the directory if it doesn't exist.

7. Copy every unloaded file (.s0, .s1, .s2, .b0, .b1, .b2, file object, user defined function file) to the 64-bit machine, file objects in the DB_FoDir path.
8. Use dmsqlc or dmsql32 execute CREATE DB command, then execute DISCONNECT , The following example shows how to create database with CREATE DB.

```
C:\DBMaker\5.4\bin>dmsqlc
dmSQL> create db DBSAMPLE5;
USE db #1 connected to db:<DBSAMPLE5> by user:<SYSADM>

dmSQL> disconnect;
USE db #1 disconnected from db:<DBSAMPLE5> by user:<SYSADM>

dmSQL>
```

9. Use dmserver.exe to start database dbsample5



10. Connect to the database and execute LOAD DB, the LOAD DB syntax is as follow:

LOAD DB FROM 'full_path\file_name';

```
dmSQL> connect to DBSAMPLE5 sysadm;
USE db #1 connected to db:<DBSAMPLE5> by user:<SYSADM>

dmSQL> load db from 'C:\DBMaker\unload\DBSAMPLE5_32bit';

dmSQL>
```

The LOAD command will generate a log file DBSAMPLE5_32bit.log, WARNING and ERROR during the LOAD command will be written into this file.

11. If it is determined that the WARNING will not affect the operation of the database, check whether there are any problems with the data, and then users can use the new database normally.

If ERROR causes LOAD DB failure, check DBSAMPLE5_32bit.log. Delete the new database and go back to step 5 if ERROR is unsolvable.

2.2. Steps in Linux System

1. Use dmsqlc connect to the 32-bit database with DBA/SYSDBA privilege or user SYSADM. Take DBMaker 5.4.x as example, the default directory of dmsqlc.exe is /home/5.4/bin/

The CONNECT syntax is as follow:

```
CONNECT TO database_name user_name password;
```

The following example shows users how to use dmsqlc connect to DBSAMPLE5 as user SYSADM:

```
dbmaker@ubuntu:~/5.4/bin$ ./dmsqlc
dmSQL> connect to dbsample5 sysadm;
USE db #1 connected to db:<dbsample5> by user:<SYSADM>
dmSQL>
```

2. Execute UNLOAD DB command, file_name is the name of exported file, the export directory is local directory, full path is also available

```
UNLOAD DB TO file_name;
```

```
UNLOAD DB TO 'full_path/file_name';
```

Continue from step1, execute UNLOAD DB, the file name is dbsample5_32bit and will be exported to /home/dbmaker/unload/

```
dbmaker@ubuntu:~/5.4/bin$ ./dmsqlc
dmSQL> connect to dbsample5 sysadm;
USE db #1 connected to db:<dbsample5> by user:<SYSADM>
dmSQL> unload db to '/home/dbmaker/unload/dbsample5_32bit';
```

3. After step2 is completed, there will be at least 2 files generated in C:\DBMaker\unload: .s0 file stores database schema; .b0 file stores database data



```
dbsampl  dbsampl
e5_32bit. e5_32bit.
  b0      s0
```

After UNLOAD DB is completed, execute TERMINATE DB to terminate database, then execute QUIT to close dmsqlc.

```
dbmaker@ubuntu:~/5.4/bin$ ./dmsqlc
dmSQL> connect to dbsample5 sysadm;
USE db #1 connected to db:<dbsample5> by user:<SYSADM>
dmSQL> unload db to '/home/dbmaker/unload/dbsample5_32bit';
dmSQL> terminate db;
USE db #1 terminated(CURRENT)
dmSQL> quit;
```

4. Backup or delete the 32-bit database (DB_DBDir directory). If database has file objects, they will also be unloaded to the directory set in the UNLOAD DB command. If users have created stored procedure in the database, stored procedures' definition will be exported to .b0. User defined function will not be unloaded, users have to manually move the original file and create it in the new database.
5. Install DBMaker 64-bit on a 64-bit platform.

6. Copy 32-bit database's dmconfig.ini to 64-bit. Take DBMaker 5.4.x as example, the default directory of dmconfig.ini will be /home/dbmaker/5.4/. If users use bundle version, the default directory of dmconfig.ini will be in the extracted file.

Notice the DB_DBDir and DB_FoDir in dmconfig.ini, the directory must exist in the new machine, create the directory if it doesn't exist.

7. Copy every unloaded file (.s0, .s1, .s2, .b0, .b1, .b2, file object, user defined function file) to the 64-bit machine, file objects in the DB_FoDir path.
8. Use dmsqls execute CREATE DB command, then execute DISCONNECT and QUIT to close dmsqls. The following example shows how to create database with CREATE DB.

```
dbmaker@ubuntu:~/5.4/bin$ ./dmsqls
dmSQL> create db dbsample5;
USE db #1 connected to db:<dbsample5> by user:<SYSADM>

dmSQL> disconnect;
USE db #1 disconnected from db:<dbsample5> by user:<SYSADM>

dmSQL> quit;
dbmaker@ubuntu:~/5.4/bin$
```

NOTE : dmsqls can only use to CREATE DB, other operations require dmsqlc

9. Use dmserver to start database dbsample5

```
dbmaker@ubuntu:~/5.4/bin$ ./dmserver dbsample5
DBMaker 5.4.5 (#31093, 20230504)
Copyright(C) 1995-2023 CASEMaker Inc. All rights reserved.
SQL Server bound to port 3000
Enable SMP: 8 CPUs
The database has started successfully.
Database Server is running in the background mode.
Process ID = 4746
```

10. Connect to the database and execute LOAD DB, the LOAD DB syntax is as follow:

```
LOAD DB FROM 'full_path/file_name';
dbmaker@ubuntu:~/5.4/bin$ ./dmsqlc
dmSQL> connect to dbsample5 sysadm;
USE db #1 connected to db:<dbsample5> by user:<SYSADM>

dmSQL> load db from '/home/dbmaker/unload/dbsample5_32bit';

dmSQL>
```

The LOAD command will generate a log file DBSAMPLE5_32bit.log, WARNING and ERROR during the LOAD command will be written into this file.

12. If it is determined that the WARNING will not affect the operation of the database, check whether there are any problems with the data, and then users can use the new database normally.

If ERROR causes LOAD DB failure, check DBSAMPLE5_32bit.log. Delete the new database and go back to step 5 if ERROR is unsolvable.