GENESYS

Framework Database Connectivity Guide

FAQs and Troubleshooting

5/5/2025

FAQs and Troubleshooting

Contents

This section contains answers to frequently asked questions about your database setup as well as some information about troubleshooting common problems. Look through this information before contacting Genesys Customer Care with your questions and problems.

Frequently Asked Questions

Question

If I change the port of my MS SQL server, how do I connect to my database?

Answer:
If you change the port of the MS SQL database, then you need to make the following changes in both the Client Network Utility and the Server Network Utility programs in MS SQL:

- On the **General** tab on the Client Network Utility, ensure that TCP/IP is enabled and click **Properties**. Enter the new port number and repeat this procedure on the Server Network Utility.

- Save the changes and your DB Client.

Question

What is the character limit for the stored procedure parameter?

Answer:
The maximum number of characters for processes are as follows:

- Parameter name: 64 bytes

- Number of parameters: 255

- Input/Output parameter limits: 2 KB

- The SQL statement passed from router to DB Server: 10240 bytes

Question

What causes the error Unresolved symbol: `getrlimit64(code)` in the DB Client's log?

Answer:
Generally, this error message is caused either by a mismatch between the DB Client and the OS version (for instance, running a 32-bit DB Client on a 64-bit OS), or by using an outdated DB Client patch.

Question

What should I do if the DB Client cannot connect to Oracle database and logs the error **ORA-12154** or **TNS-12154**?

Answer:

This error indicates that the Listener may be incorrectly configured in the **tnsnames.ora** file. Carefully check to ensure that the Listener is entered correctly in **tnsnames.ora**.

Question

What should I do if the DB Client cannot connect to an Oracle database and logs the error **ORA-12560**?

Answer:
To correct this problem, check that the **DBMS Name** parameter specified in the DAP Application object configured in the Configuration Layer is correct-it must match the name of the Listener configured in the **tnsnames.ora** file. Note that the **DBMS Name** parameter, although mandatory for all other database types, is not required for Oracle and thus can be left blank.

Question

What should I do if the DB Client cannot connect to Oracle database and records the error **ORA-12203** or **TNS-12203**?

Answer:
To correct this problem, verify that the option **dbserver** in the DB Client's configuration matches the setting of the **ORACLE_SID** environment variable on the computer on which the DB Client is running. It might also be helpful to analyze the **sqlnet.log** log file in the DB Client installation folder. This file is created by the Oracle Client in case there is a need to troubleshoot a network configuration problem.

Question

How do I make a symbolic link for dbclient if I decided to switch between DBMS?

Answer:

Currently, when installing DBServer, it will create symbolic links for only the DBMS selected during installation. If you use more than one DBMS or decide to switch the DBMS, then you must make dbclient symbolic links manually.

The following are the commands to create a symbolic link for a different dbclient.

64-bit

```
sudo ln -s dbclient_postgre_64 dbclient_postgre
sudo ln -s dbclient_oracle_64 dbclient_oracle
sudo ln -s dbclient_msql_64 dbclient_msql
sudo ln -s dbclient_db2_64 dbclient_db2
```

32-bit

```
sudo ln -s dbclient_postgre_32 dbclient_postgre
sudo ln -s dbclient_oracle_32 dbclient_oracle
sudo ln -s dbclient_db2_32 dbclient_db2
```

Troubleshooting

This section discusses the solutions to several problems commonly encountered with DB Server.

> ## Important
> Set the value of the **verbose** option to 3 when you are troubleshooting a DB Client. This setting provides a detailed log that may indicate the source of a particular problem. After you resolve the problem, reset the verbose option to its previous level to increase performance.

Increase the Processes Allowed Limit in Windows

By default, Microsoft Windows limits each local system account to approximately 120 processes that are running as-or are started by-services. When the number of running processes reaches this limit, any attempt to start an additional process will fail. Because Windows counts DBClient processes, 120 can be too low a limit. One example of when this may happen is during Outbound deployment with a high number of calling lists.

To change the number of processes allowed, follow the procedure described in the Microsoft Knowledge Base article located here, which describes changing the registry value:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager\SubSystems\
Windows
```

In this procedure, each additional 512 kilobytes that you specify in the third **SharedSection** parameter will allow approximately 120 additional processes/services to be started. The default **SharedSection** setting is highlighted in the default registry value data:

```
%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
SharedSection=1024,3072,512
Windows=On
SubSystemType=Windows
ServerDll=basesrv,1
ServerDll=winsrv:UserServerDllInitialization,3
ServerDll=winsrv:ConServerDllInitialization,2
ProfileControl=Off
MaxRequestThreads=16
```

Estimate the number of processes/services and apply it to the above calculation. For example:

- 65 services are running under the local system account.

- You plan to start DB Server as a service under the local system account.

- You plan to use 60 connections from Genesys DB Client applications to the DBMS.

These estimates require 125 services, which exceeds the default limit of 120.

In this example, Genesys recommends that you set the third **SharedSection** parameter to 1024, which increases the limit to 240. Use the Registry Editor to change the setting, as follows:

SharedSection=1024,3072,1024

> ## Warning
>
> Within the procedure, Microsoft presents a disclaimer about the Registry Editor: If you use Registry Editor incorrectly, you may cause serious problems that may require you to reinstall your operating system. Microsoft cannot guarantee that you can solve problems that result from using Registry Editor incorrectly. Use Registry Editor at your own risk.

DB Client Cannot Open the SQL Server Database

- Make sure the SQL Server client is properly installed and configured on the machine on which the DB Client is running (see the *Framework Deployment Guide* for more information).

- When DB Server tries to open the database, the following message appears in the log:

```
DBClient 96 - Server: username = 'user', password = '***********', database = 'test', DB Server = 'cti'
```

  Try to open the database manually (with the SQL Server client) using the values from the log message. If you can't do this, the problem lies with your SQL Server client configuration. In this case, contact your Database Administrator.

Request to Execute SQL Statement Fails

Check the log file for error messages. You may need to contact your Database Administrator for information on database-specific error codes. Check the log for the text of the SQL command that fails. Try to execute the same command manually with the SQL Server client, using the same username, password, and so on as specified in the Configuration Layer (configuration file).