

Using SMO to copy a database and data

Asked 16 years, 1 month ago Modified 9 years, 2 months ago Viewed 28k times



23

I am trying to make a copy of a database to a new database on the same server. The server is my local computer running SQL 2008 Express under Windows XP. Doing this should be quite easy using the SMO.Transfer class and it almost works!



My code is as follows (somewhat simplified):



```
Server server = new Server("server");
Database sourceDatabase = server.Databases["source database"];

Database newDatabase = new Database(server, "new name");
newDatabase.Create();

Transfer transfer = new Transfer(sourceDatabase);
transfer.CopyAllObjects = true;
transfer.Options.WithDependencies = true;
transfer.DestinationDatabase = newDatabase.Name;
transfer.CopySchema = true;
transfer.CopyData = true;
StringCollection transferScript = transfer.ScriptTransfer();

using (SqlConnection conn = new SqlConnection(connectionString))
{
    conn.Open();
    using (SqlCommand switchDatabase = new SqlCommand("USE " + newDatabase.Name,
conn))
    {
        switchDatabase.ExecuteNonQuery();

        foreach (string scriptLine in transferScript)
        {
            using (SqlCommand scriptCmd = new SqlCommand(scriptLine, conn,
transaction))
            {
                int res = scriptCmd.ExecuteNonQuery();
            }
        }
    }
}
```

What I do here is to first create a new database, then generate a copy script using the `Transfer` class and finally running the script in the new database.

This works fine for copying the structure, but the `CopyData` option doesn't work!

Are there any undocumented limits to the `CopyData` option? The documentation only says that the option specifies whether data is copied.

I tried using the `TransferData()` method to copy the database without using a script but then I get an exception that says "Failed to connect to server" with an inner exception that says "A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not accessible. Verify that the instance name is correct and that SQL Server is configured to allow remote connections. (provider: Named Pipes Provider, error: 40 - Could not open a connection to SQL Server)"

I also tried to enable Named Pipes on the server, but that doesn't help.

Edit: I found a solution that works by making a backup and then restoring it to a new database. It's quite clumsy though, and slower than it should be, so I'm still looking for a better solution.

c# .net sql-server-2008 smo

Share

Improve this question

Follow

edited Sep 28, 2015 at 6:42



Irshad

3,131 ● 5 ● 32 ● 54

asked Nov 6, 2008 at 11:42



Rune Grimstad

36.2k ● 10 ● 65 ● 77

4 Answers

Sorted by: Highest score (default) ◆



20

Well, after contacting Microsoft Support I got it working properly, but it is slow and more or less useless. Doing a backup and then a restore is much faster and I will be using it as long as the new copy should live on the same server as the original.



The working code is as follows:



```
ServerConnection conn = new ServerConnection("rune\\sql2008");
Server server = new Server(conn);

Database newdb = new Database(server, "new database");
newdb.Create();

Transfer transfer = new Transfer(server.Databases["source database"]);
transfer.CopyAllObjects = true;
transfer.CopyAllUsers = true;
transfer.Options.WithDependencies = true;
transfer.DestinationDatabase = newdb.Name;
transfer.DestinationServer = server.Name;
transfer.DestinationLoginSecure = true;
transfer.CopySchema = true;
transfer.CopyData = true;
transfer.Options.ContinueScriptingOnError = true;
transfer.TransferData();
```

The trick was to set the DestinationDatabase property. This must be set even if the target is that same as the source. In addition I had to connect to the server as a named instance instead of using the other connection options.

Share Improve this answer Follow

answered Nov 26, 2008 at 14:28



Rune Grimstad

36.2k ● 10 ● 65 ● 77

- 2 A backup and restore will always be faster because it's not subject to locking overhead, transactions, etc. Backups are designed to be ridiculously fast for this sort of thing. Sorry I didn't get to you sooner! :-D – Brent Ozar Dec 1, 2008 at 13:13

But running a backup might interfere with the current backup sequence and log transfer/truncation. Can you use SMO to create a COPY_ONLY backup as described at msdn.microsoft.com/en-us/library/ms191495.aspx – flipdoubt Dec 8, 2010 at 15:37

also, backup and restores with C# SMO loses all users and logins, doesn't it? – DevDave May 31, 2013 at 9:33

Its worth noting this only works if you are logged in as an "sa" user otherwise when you create the data your user gets assigned as dbo and then it tries to create a new user in the target database of your current user and you get an exception. Even when you specifically tell it not to copy logins, usernames or anything it wants to create your current user for some reason. – ScottGuymer Apr 23, 2015 at 14:16 ✎

I used the above code but i am getting exception in "Object Reference Not set To Instance Of an Object" while i set the property transfer.copydata=true..And also without this property mean, Schema getting copied fine no issue..How to solve this – Dharani Dharan Feb 19, 2016 at 6:36



3



Try setting [SetDefaultInitFields](#) to true on the **Server** object.

I had the same issue with the SMO database object running slowly. I guess this is because sql server doesn't like to retrieve entire objects and collections at once, instead lazy loading everything, causing a round-trip for each field, which for an entire database is pretty inefficient.

Share Improve this answer Follow

answered Jun 23, 2009 at 6:07



Rob

3,066 ● 4 ● 31 ● 32



3

I had a go at getting this working and have come up with an answer that doesn't use the Transfer class. Here is the Method i used:



```
public bool CreateScript(string oldDatabase, string newDatabase)
{
    SqlConnection conn = new SqlConnection("Data Source=.;Initial Catalog="
+ newDatabase + ";User Id=sa;Password=sa;");
    try
    {
        Server sv = new Server();
        Database db = sv.Databases[oldDatabase];

        Database newDatabase = new Database(sv, newDatabase);
        newDatabase.Create();

        ScriptingOptions options = new ScriptingOptions();
        StringBuilder sb = new StringBuilder();
        options.ScriptData = true;
        options.ScriptDrops = false;
        options.ScriptSchema = true;
        options.EnforceScriptingOptions = true;
        options.Indexes = true;
        options.IncludeHeaders = true;
        options.WithDependencies = true;

        TableCollection tables = db.Tables;

        conn.Open();
        foreach (Table mytable in tables)
        {
            foreach (string line in
db.Tables[mytable.Name].EnumScript(options))
            {
                sb.Append(line + "\r\n");
            }
            string[] splitter = new string[] { "\r\nGO\r\n" };
            string[] commandTexts = sb.ToString().Split(splitter,
StringSplitOptions.RemoveEmptyEntries);
            foreach (string command in commandTexts)
            {
                SqlCommand comm = new SqlCommand(command, conn);
                comm.ExecuteNonQuery();
            }
            return true;
        }
        catch (Exception e)
        {
            System.Diagnostics.Debug.WriteLine("PROGRAM FAILED: " + e.Message);
            return false;
        }
        finally
        {
            conn.Close();
        }
    }
}
```

Share Improve this answer Follow

answered Oct 18, 2011 at 13:25

 emy
31 ● 1

I get error on `sb.append(line+"\n\n")` , "Object Reference Not set To Instance Of an Object"
– [francesco totti](#) Sep 13, 2013 at 14:41

@francescototti: no you didn't. There's no way you'd get that error unless you made modifications to the code that resulted in `sb` being null. – [siride](#) Feb 9, 2015 at 22:34



Here is my solution:

1

1. I have a Database named is Olddbatabase

2. I backup it to E:\databackup\Old.bak



3. If you want to create a Duplicate Database from Olddbatabase in the same server with name NewDatabase



3.1 You can use command in query tool : `EXEC OldDatabase.dbo.sp_helpfile;` to determinat path of OldDatabase is stored in case you want to save NewDatabase in the same folder.

or You can save NewDatabase in new Path which you want

1. use this command in Query tool

```
RESTORE DATABASE NewDatabase FROM DISK = 'E:\databackup\Old.bak'  
WITH MOVE 'OldDatabase' TO 'E:\New path (or the same  
path)\NewDatabase_Data.mdf', MOVE 'OldDatabase_log' TO 'E:\New path (or  
the same path)\NewDatabase_Log.ldf';
```

Note: you can Use these command above in c# with : Create a Store procedure in sql which include Above commands. And you can call the store procedure in C #

[Share](#) [Improve this answer](#) [Follow](#)

answered Dec 8, 2013 at 14:22



[emcuapharaon](#)

11 ● 1
