MSSQLTips.com
brought to you by Edgewood Solutions

Your daily source for SQL Server Tips

Rebuilding the SQL Server master database - Part 1

Written By: Tim Ford -- 6/23/2008

Problem

I recently found myself in a situation where I had to rebuild the master database on one of the nodes of our development SQL 2005 cluster. I had to do so because there was a server collation requirement by one of the databases we were looking to host on the server. The collation change would not cause issues with any of the current databases on the instance so I could safely change the server collation for the instance. The process itself is quite simple, the proof is in the preparation however. The chances things will go wrong are low, but you still need to be prepared for the possibility that your quick rebuild is anything but. The need to prepare is also critical because once you rebuild the master database you'll notice that there is a little issue involving NO LOGINS! Yikes! Yes, the rebuild gives you a new master database fresh from the DVD. So, how do you prepare for a master rebuild and what do you do afterwards? Hopefully this checklist will give you an idea.

Solution

As mentioned above, the below steps are the process for getting ready for a rebuild of the master database. In each of these steps I will show you scripts that can be run to collect the data needed to quickly rebuild your server after you have restored the master database. In future tips we will walk though the actual rebuild process, but for now we will focus on having the necessary data on hand to do the rebuild.

Getting Ready

The biggest thing here is having the necessary data on hand and ready to apply once the master database has been rebuilt. Without the data below you will need to do a lot of manual work to get your server up and running again. So in each of these steps we look at what data you will need and how you can script it out from your server prior to doing a rebuild.

Prior to running these change the query output to Text (Ctrl-T), so you can easily save the output.

Task	Steps

Script "Detach of All User Databases"	Before rebuilding master you'll detach all of the user database on the instance. This script will provide output for doing so.
	
	``--CODE BEGIN
	DECLARE @Command_Detach VARCHAR(1000)
	SELECT @Command_Detach = '
	IF N''[?]'' NOT IN(''model'' ,''master'', ''msdb'', ''tempdb'')
	BEGIN
	SELECT ''EXEC usp_KillDBConnections ''''?''''
	EXEC sp_detach_db ''''?'''',''''true''''''
	END'
	
	EXEC sp_MSforeachdb @Command_Detach
	--CODE END```
	
	Save the output of this query as **0.sql**

Script "Attach of All User Databases"	After rebuilding the master database you'll need to attach all the user databases you previously detached. You'll also notice that when you attach the databases the "dbo" alias will default to the login the script is executed as. Therefore you'll need to not only script out the attach process, but also an ownership change.
	
	Script Attach of all user databases as follows, saving the results as **1.sql**
	
	``--CODE BEGIN
	DECLARE @Command_Attach VARCHAR(1000)
	SELECT @Command_Attach = '
	IF N''[?]'' NOT IN(''model'' ,''master'', ''msdb'', ''tempdb'')
	BEGIN
	SELECT ''EXEC sp_attach_db ''''?'''',
	'''''' + RTRIM(filename) + '''''','''
	FROM [?]..sysfiles
	WHERE fileid = (SELECT MIN(fileid) FROM [?]..sysfiles)
	UNION ALL
	SELECT '''''''' + RTRIM(filename) + ''''''''
	FROM [?]..sysfiles```

| | ```
WHERE fileid > (SELECT MIN(fileid) FROM [?]..sysfiles) AND
fileid < (SELECT MAX(fileid) FROM [?]..sysfiles)
UNION ALL
SELECT '''''''' + RTRIM(filename) + ''''''''
FROM [?]..sysfiles
WHERE fileid = (SELECT MAX(fileid) FROM [?]..sysfiles)
END'

EXEC sp_msforeachdb @Command_Attach
--CODE BEGIN
``` |
| Script "All Logins" | The master database is the logical repository for all system objects in a SQL instance. (The physical repository being the resource database in SQL 2005.) When the master database is rebuilt any changes made to it and data placed into it since installation, will be lost. These items will need to be replaced after the rebuild process. The most-critical of these item types are the instance's logins. The following scripts will provide you with the means to restore all of your logins after the rebuild process:

http://support.microsoft.com/kb/918992 contains scripts for the creation of two stored procedures: sp_hexadecimal and sp_help_revlogin. Download the scripts, run them in the master database to create the stored procedures, then run the following script that utilizes those stored procedures to script out all the logins for every database on your SQL instance:

```
--CODE BEGIN
EXEC dbo.sp_help_revlogin
--CODE END
```

Make sure you change the output for the query to either text or to file so you'll be able to run this script after the rebuild.

For the sake of this tip, name the file **2.sql.** |
| Script "DBO Aliasing for User Databases" | Save the results of the following script that will re-establish "dbo" aliasing as **3.sql**

```
--CODE BEGIN
DECLARE @Command_dbo VARCHAR(1000)
SELECT @Command_dbo =
'
use [?]
DECLARE @Owners TABLE (dbid int, dbname varchar(100), loginname varchar(100))
INSERT INTO @Owners (dbid, dbname, loginname)
SELECT DB_ID(), DB_NAME(), SL.[name]
FROM sysusers SU LEFT JOIN master..syslogins SL ON SU.sid = SL.sid
WHERE SU.[name] = ''dbo''

SELECT ''EXEC [?].dbo.sp_changedbowner N'''''' + loginname + '''''', '' + ''false''
FROM @Owners WHERE dbid = DB_ID()
'
EXEC sys.sp_MSforeachdb @Command_dbo
--CODE END
``` |
| Script "All Securables" | Any securables granted at the instance level is necessary because that information is also stored within the master database. Database securables will be retained in the databases themselves. I found the code located here, created by Pedro Lopes perfect for this task at hand. My suggestion is to limit the results to only the system databases, as the user databases will be reattached with their securables intact.

Save the output of this script as **4.sql**. |
| Script "Custom Objects in master and msdb Databases" | Quite often we DBAs find ourselves creating objects such as the sp_hexadecimal and sp_help_revlogin stored procedures we created earlier in the system databases. Be sure that if you wish to recreate these objects after rebuild that you follow the same process outlined above for each object (script as create to file.) |
| Script "Remaining Objects" | For each of the following object types you wish to recreate after the rebuild of the master database (System Objects, SQL Agent Jobs, Operators) perform the following steps:

• Select object in Object Explorer
• Right click and select Script as CREATE
• Select To File
• Take note of the file name for restoration process after rebuild |

Without the above information ready to be applied after the master database has been rebuilt you would need to go through a lot of manual work to get your server ready for users. Although this was an exercise in how to

pull the data together, these steps should be built into your disaster recovery steps so you always have the necessary data on in hand in case of the need to restore the master database.

<u>Next Steps</u>

- At this point you're prepared to rebuild the master database. The next tip in this series will outline that process and the final tip will walk you through the restoration of databases and objects post-rebuild.
- sp_MSforeachdb is thoroughly explained in this <u>tip</u>.
- For information concerning the system databases review this <u>tip</u>.