

## Tech Note 1052

# Resolving "DATABASENAME Transaction Log is Full" Issue

---

All Tech Notes, Tech Alerts and KBCD documents and software are provided "as is" without warranty of any kind. See the [Terms of Use](#) for more information.

Topic#: 002896

Created: July 2014

## Introduction

Wonderware products such as Application Server, Historian Server, Corporate Energy Management (CEM), Information Server, MES, etc. use Microsoft SQL Server databases. Each of these products uses a data file (.MDF file) and a Transaction Log file (LDF file).

In some cases, the Transaction Log file size becomes very large, on the scale of giga-bytes. This *Tech Note* describes a step-by-step procedure to solve the issue.

You might see the following error in the ArchestrA logger because the Transaction Log is full:

**The transaction log for database 'DatabaseName' is full. To find out why space in the log cannot be reused, see the log\_reuse\_wait\_desc column in sys.databases.**

**Note:** The following procedure should be common for all databases. The Historian Runtime database is the example provided in this *Tech Note*.

## Application Versions

- SQL Server 2008 or 2012
- System Platform 2012R2 P01 or
- Wonderware System Platform 2014 and later

## Abbreviations

- **SSMS:** SQL Server Management Studio

**Note:** This *Tech Note* assumes that you are familiar with Microsoft SQL Server Management Studio and Wonderware System Platform products. If you have any questions regarding the corresponding Microsoft products, contact Microsoft Technical Support at [www.microsoft.com](http://www.microsoft.com) for further assistance.

## Procedure

There are several possible reasons why the Transaction Log cannot be reused, such as Open Transactions, Replication, a Transactional Backup, etc. In order to find out which one applies to your case, execute the following statement:

```
SELECT name, log_reuse_wait_desc  
FROM sys.databases
```

The **log\_reuse\_wait\_desc** column shows only one of the reasons. After fixing the first one, and querying the sys.databases view again, you might see a different log\_reuse\_wait reason.

For more information about the log\_reuse\_wait\_desc column: <http://msdn.microsoft.com/en-us/library/ms345414.aspx>.

Then do the following:

- Stop all current operations that use the database in question. For this example, because we are using the Historian Runtime database, we would first do a Shutdown and Disable Historian operation.
- You can use the SQL Server Activity Monitor's Processes to see what applications are using the database and you can right-click on the corresponding SessionID, then select **Kill Process** to drop the connection.

## Backup the Transaction Log

Before doing anything more, back up the database, including the data (.MDF) and the Transaction Log (.LDF) files.

1. Right-click on the database (Runtime database is used in this example) and click **Properties**, then click **Task/Back Up**.
2. In the **Back Up Database** dialog box, select the database name and set the Backup Type to **FULL**.
3. In the **Destination** area choose the location on the disk where you want to store the backup and then click **OK** (Figure 1 below).

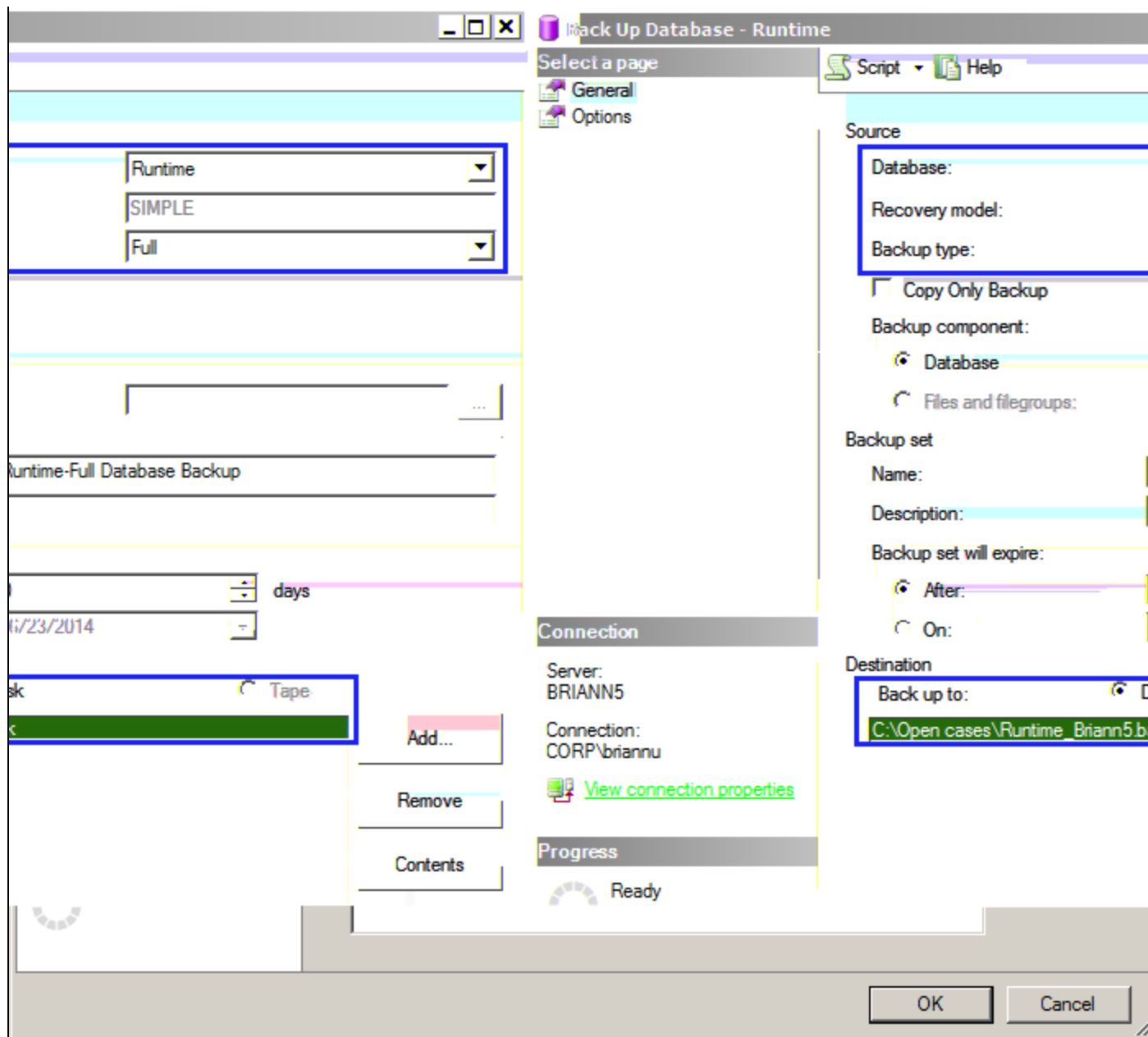


FIGURE 1: CONFIGURE DATABASE BACKUP OPTIONS

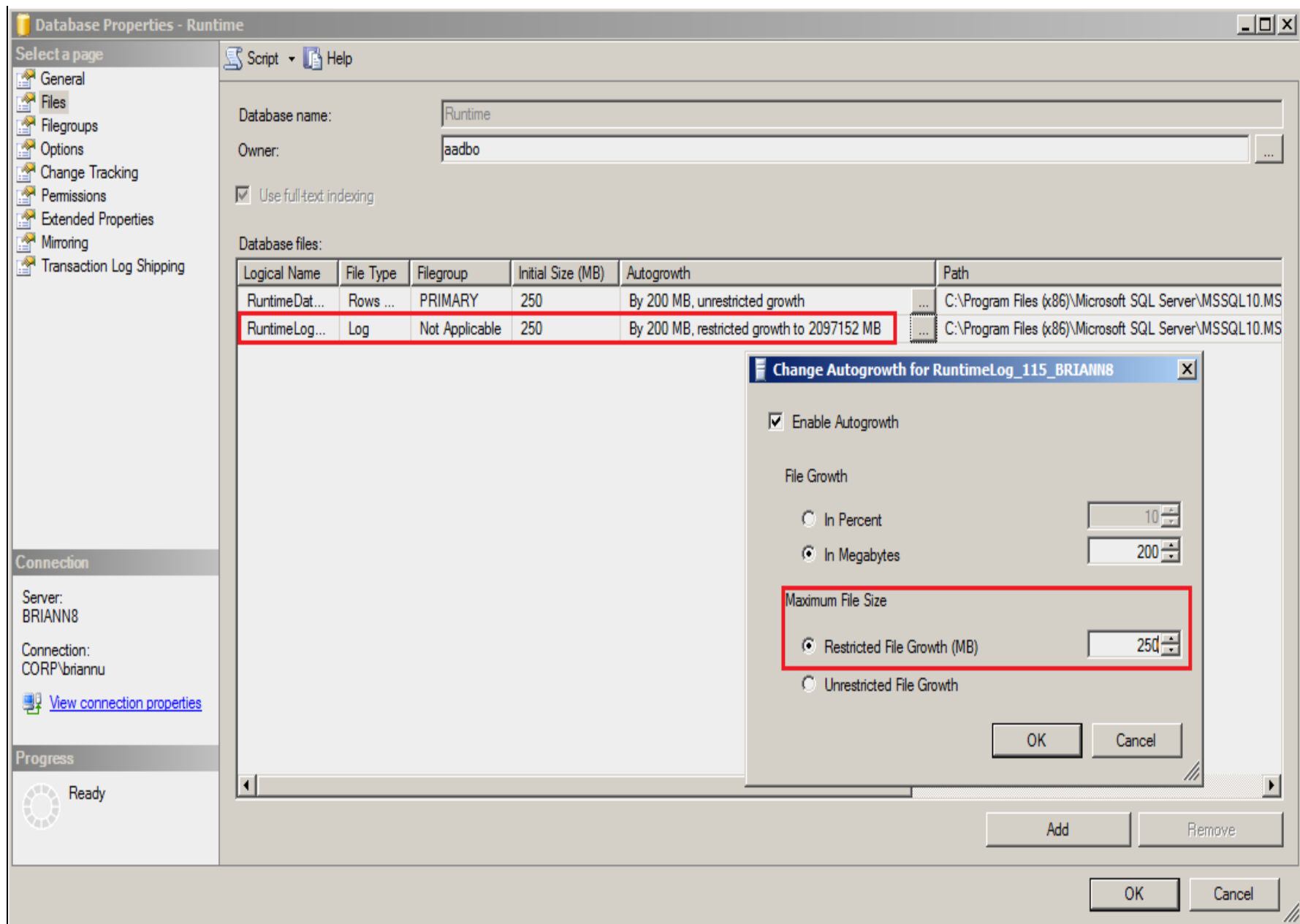
## Configure Autogrowth and Truncate Options

You can configure the following options to mitigate and solve the issue:

### Limit Autogrowth for Transaction Log File Size

When a new database is created, the file growth for the database's Transaction Log is set to 2 Tera-bytes by default. In addition, since changing the file size alone should help resolve the issue. Nevertheless, reduce the Transaction Log file Autogrowth file to an appropriate file size.

1. Right-click on the database and choose **Properties**.
2. In the **Database Properties** window, go to the **Files** page and click the Transaction Log file's **Autogrowth** ellipse.
3. Change the Restricted File Growth for the Transaction Log to a proper size. In this example, it is reduced from 2097152 MB to **250 MB**.
4. Click **OK** to allow it to reduce the Transaction Log size.

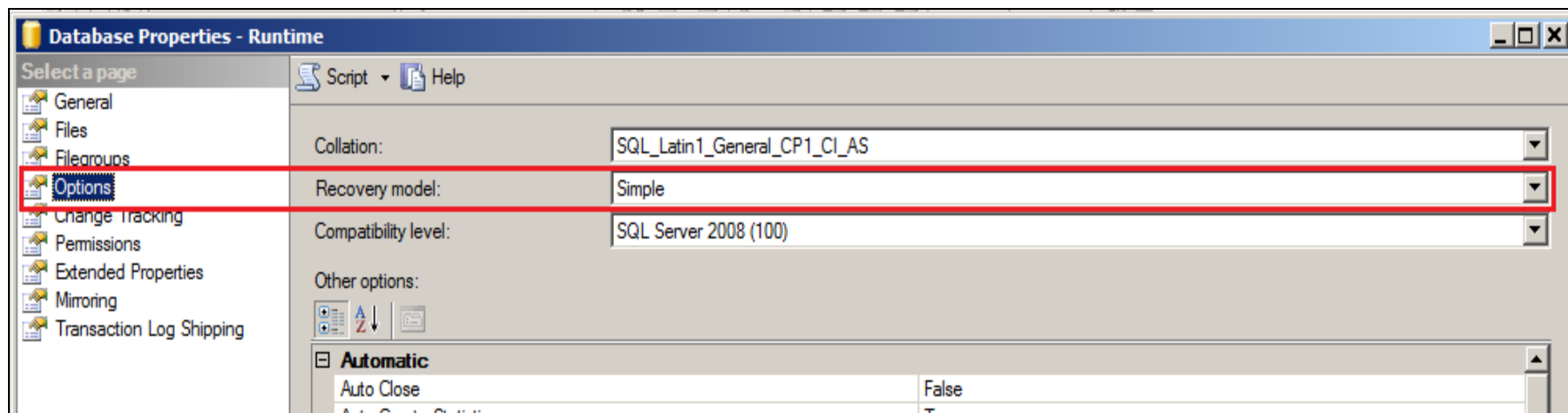


**FIGURE 2: CHANGE AUTOGROWTH OPTION**

**Note:** The file size in this *Tech Note* is used for example only. Your setting will depend on your system requirements, how many transactions occur, and how long you need to keep the log. This operation does not have any negative effect on Historian Data collection.

## Truncate the Transaction Log

1. Right-click on the database and click **Properties**.
2. Go to **Options** and change the **Recovery Model** to **Simple**.



**FIGURE 3: SIMPLE RECOVERY MODEL**

Now you need to shrink the Transaction Log to free up the space.

1. Right-click again on the database and click **Task/Shrink/Files**.
2. In the **Shrink File** window, select the **Log** File type. The name of the transactional log file is displayed in the **File name** drop down list.
3. In the **Shrink action** area, click **Release unused space** and then click **OK** (Figure 4 below).

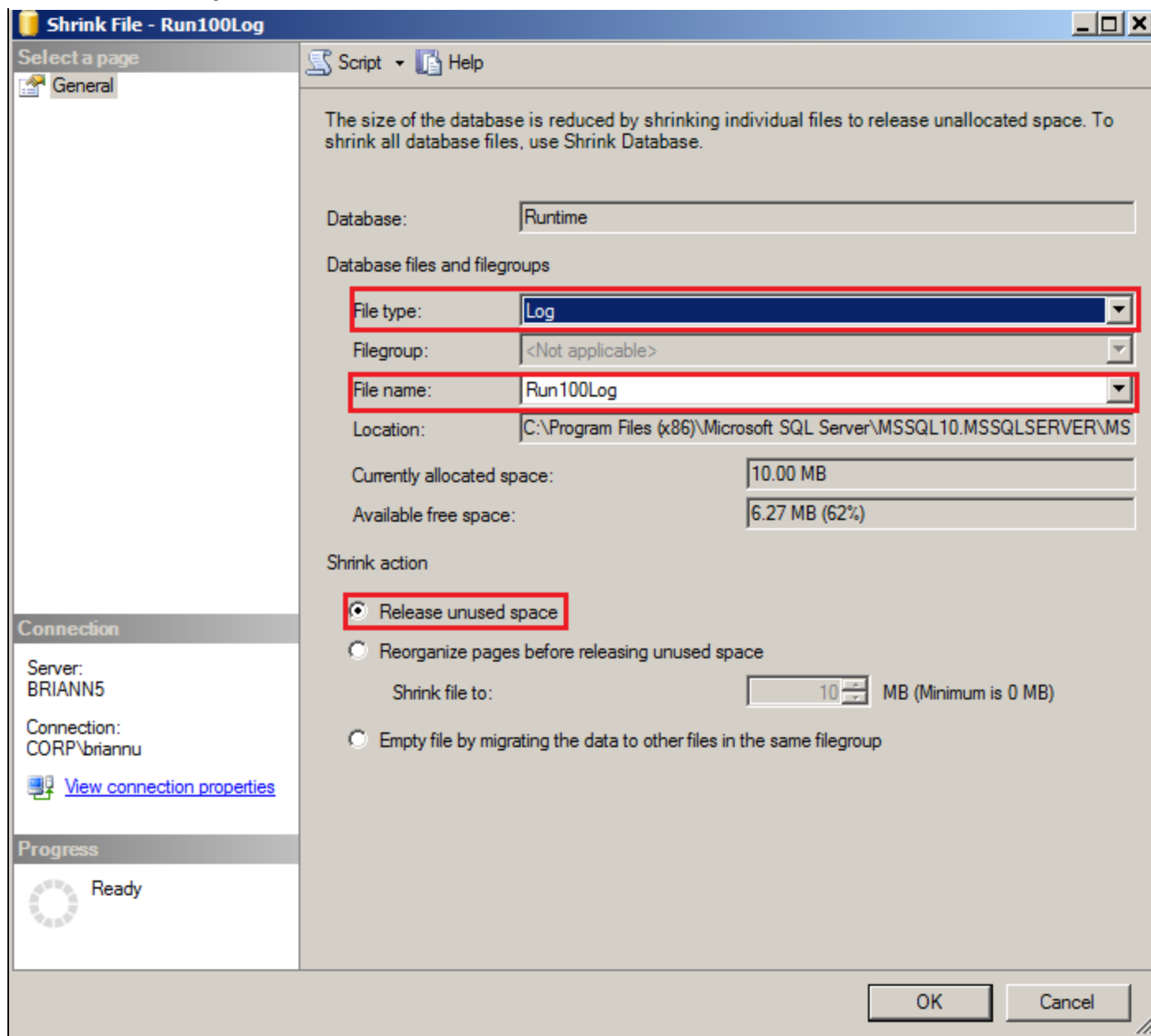


FIGURE 4: SHRINK LOG FILE

### Option: Use TSQL to Shrink the Database

You can use a Transact SQL query to accomplish the same thing.

Execute the following query to find out the log file name (we are using the Historian Runtime database in this example):

```
USE Runtime  
GO  
sp_helpdb 'Runtime'
```

Execute the following query to shrink the Transaction Log file:

```
USE Runtime  
GO  
DBCC SHRINKFILE (N'Database_Name_Log', 0, TRUNCATEONLY)  
GO
```

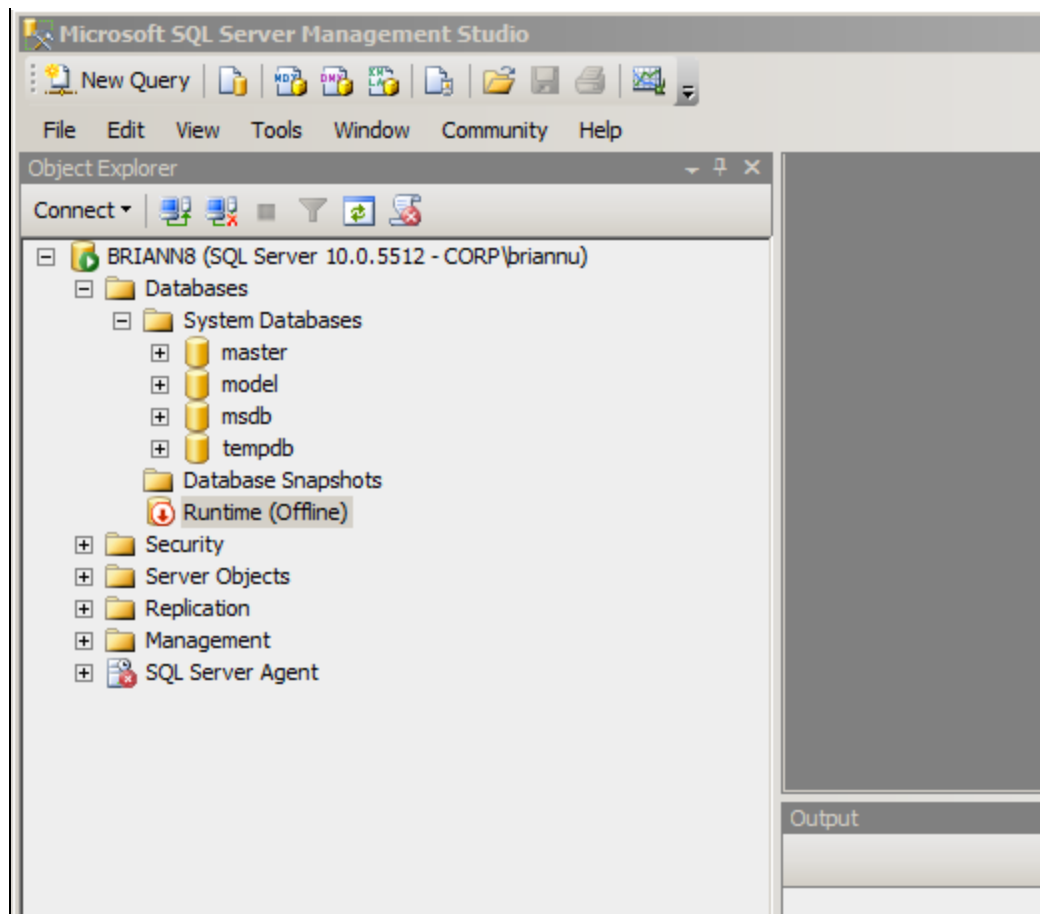
## Creating a New Transaction Log File

If for some reason, it is impossible to perform any of the above options, you can create a new Transaction Log file.

For Historian Server, creating a new Transaction Log should have no impact on the Runtime data collection. However, it can potentially impact tag/namespace updates (e.g. from a "deploy" operation, a Database Configuration Export and Import, or from the SMC). However, this should not be a significant problem—worst case scenario, you might need to re-deploy and might lose data for a few newly-created tags (e.g. the ones that don't exist in the database yet, but which have history collected)—and even that scenario is pretty unlikely.

1. Stop all current operations that are using the database. For this example, since I am using the Historian Runtime database, I would first need to perform a Shutdown and disable Historian task.
2. From the SSMS, navigate to the corresponding database, then right-click the database name and click **Tasks/Take Offline**. Figure 5 (below) shows the (Runtime) database offline.





**FIGURE 5: TAKE THE DATABASE OFFLINE**

3. Make a copy of the corresponding Transaction Log file by opening Windows Explorer and navigating to the Transaction Log folder. See [Figure 2 above](#) for the name and location of the Transaction Log file.

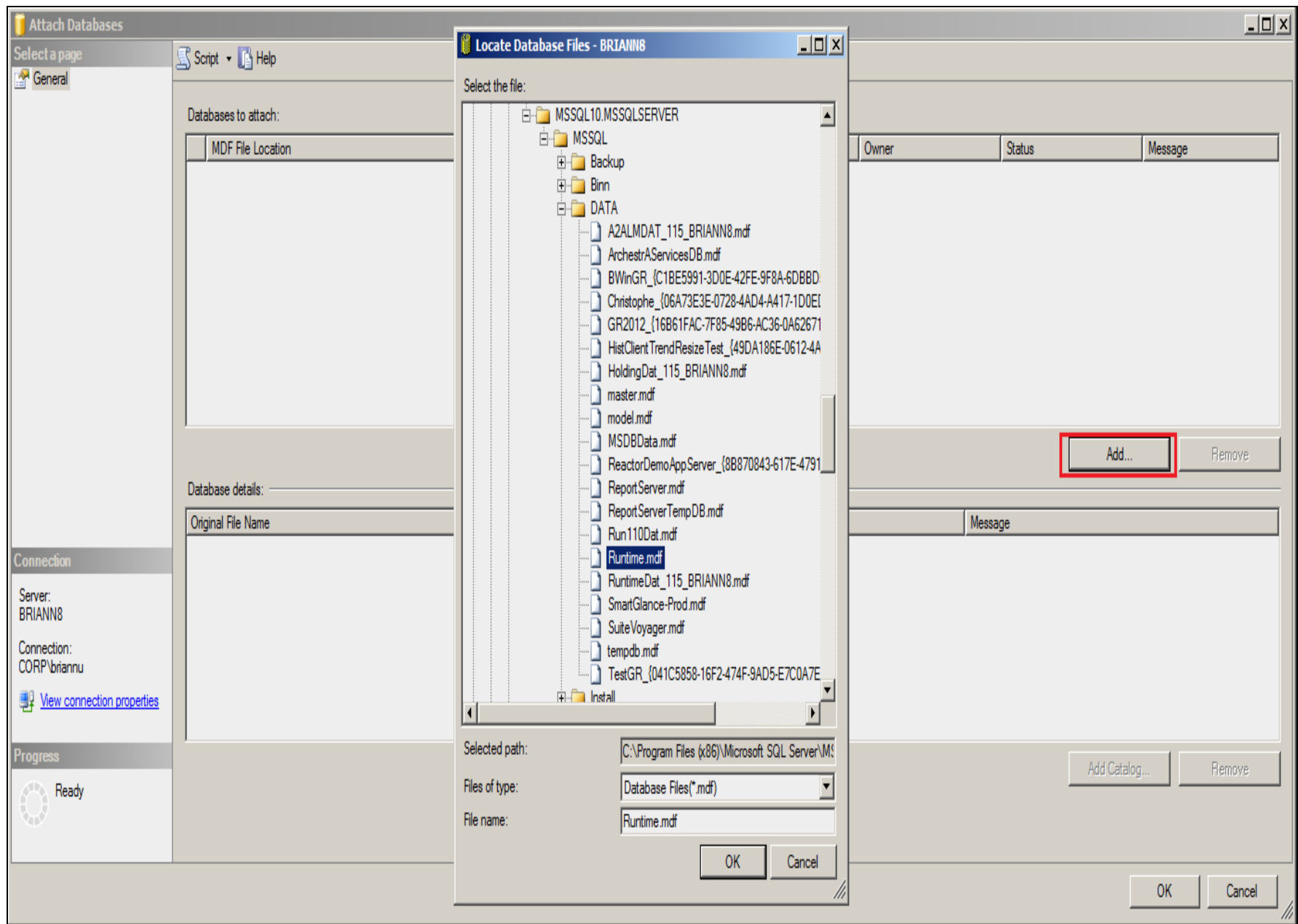
In this example, the file is at:

**C:\Program Files (x86)\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\DATA**

4. Detach the database by opening the SSMS, then navigating to the corresponding database.
5. Right-click on the database name and click **Tasks/Detach**.
6. In the **Detach Database** window, ensure you see the correct name of the database then click **OK**.
7. Rename the Transaction log file to something else.
8. From the SSMS, right-click on Databases and click **Attach**, then click **Add** and select name of the database.

In this example, the name of the Historian database is **Runtime.mdf**.

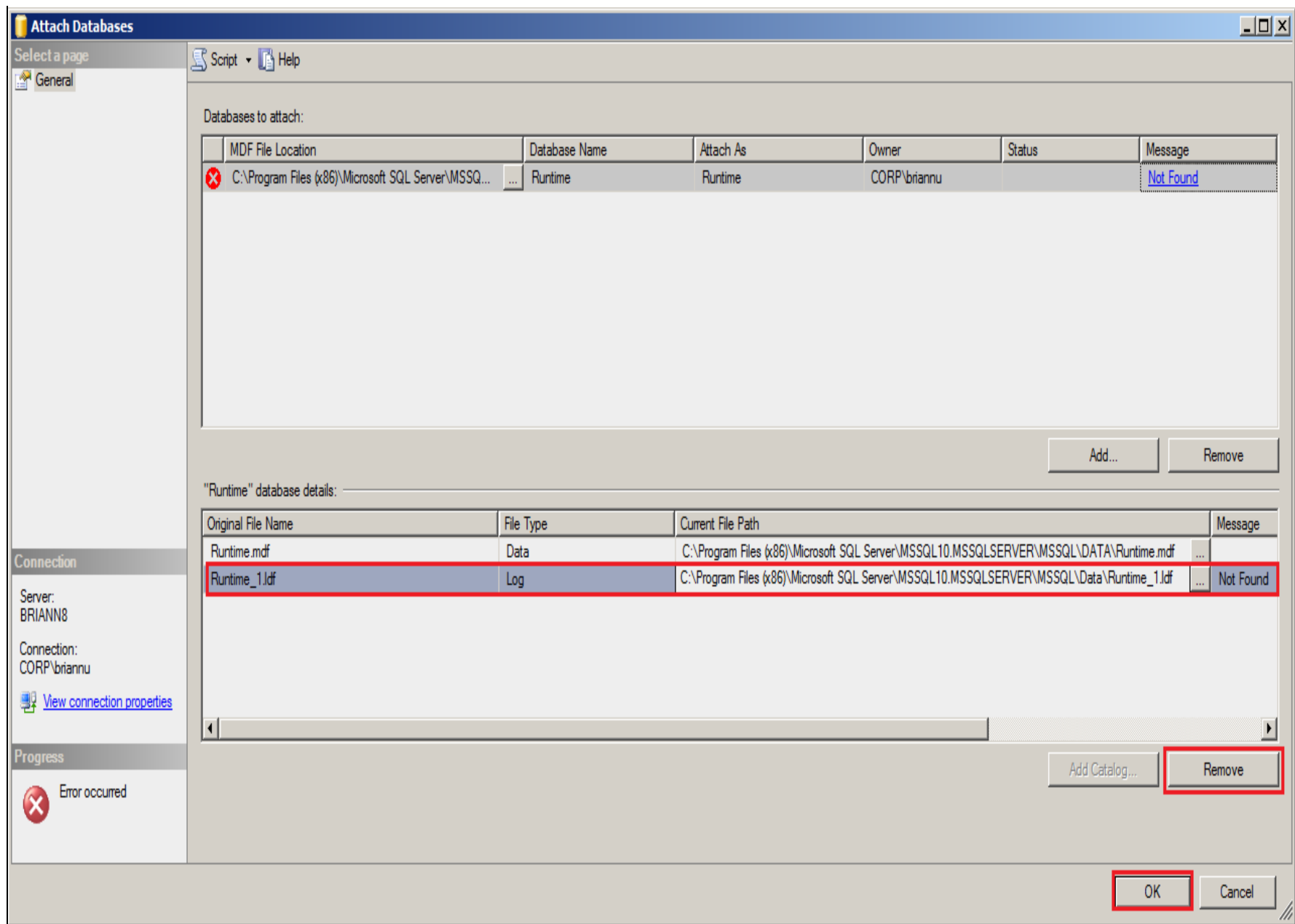
9. Click **OK** to proceed (Figure 6 below).



**FIGURE 6: ADD RUNTIME.MDF**

In the **Attach Databases** window, you see the log file location and a **Not Found** Message (Figure 7 below).

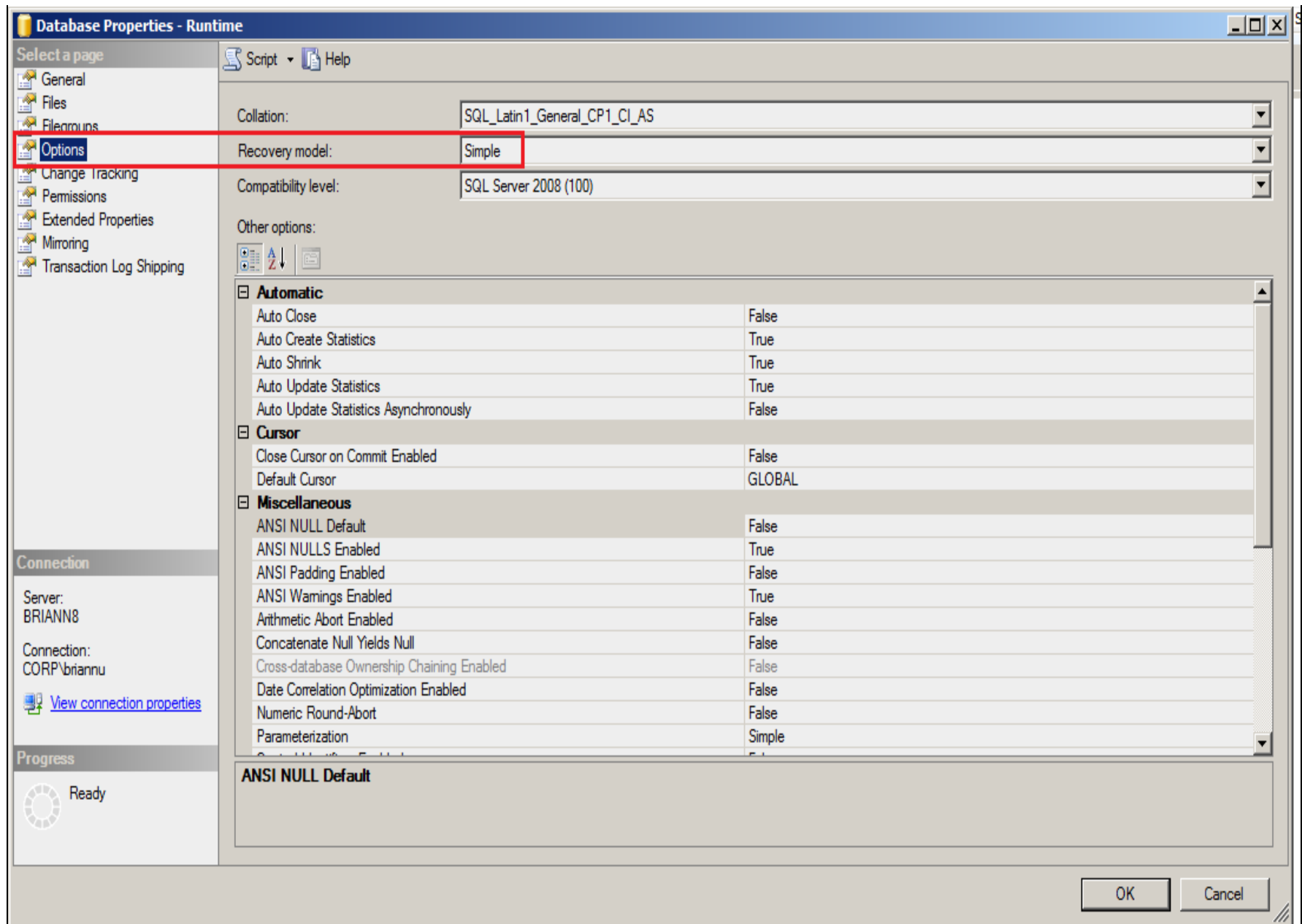
10. Click **Remove** then **OK** to proceed.



**FIGURE 7: RUNTIME DATABASE AND RUNTIME LOG**

A new Transaction Log file is created. In this example it is called **Runtime.ldf**.

- Right-click the Runtime db and click **Properties**, then click **Options** and make sure the recovery mode is set to **Simple** (Figure 8 below).



**FIGURE 8: RECOVERY MODEL = SIMPLE**

12. Start the operations using the database.

B. Nguyen

For technical support questions, send an e-mail to [wwwsupport@invensys.com](mailto:wwwsupport@invensys.com).

 [Back to top](#)

©2014 Invensys Systems, Inc. All rights reserved. No part of the material protected by this copyright may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, broadcasting, or by any information storage and retrieval system, without permission in writing from Invensys Systems, Inc.

[Terms of Use](#).