This article describes how to change the location of the data files and the log files for any Microsoft SQL Server 2012 database which are configured with SQL Server AlwaysOn with minimal downtime.

**Considerations**

We have three node windows 2008/2012 cluster where SQL server is installed and configured for AlwaysOn. The location of AlwaysOn databases on all the three nodes are similar (C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data). You run out of disk space on drive C:\ and decide to move the physical file of AlwaysOn to a new drive (D:\data\).

Node1 –Default instance (Primary replica)

Node2 –Default instance (Secondary replica –Synchronous)

Node3 –Default instance (Secondary replica –Asynchronous)

One of the solutions to accomplish this is completely remove the AlwaysOn configuration and reconfigure with the new location. But this process is time consuming and requires a lot of downtime. To overcome this we can follow the below steps which would minimise the downtime required to complete the relocation of database files.

**Benefits**

·         Downtime time is time required for two failovers for any number of replicas.

·         Downtime is only for application/users using that databases (unlike SQL instance restart that means downtime for applications/users using remaining databases too).

**Prerequisites**

* Perform a full database backup on all the databases participating in SQL Server AlwaysOn.
* Make sure that you have system administrator (sa) permissions.
* Make sure that you know the name and the current location of all data files and log files for the database.

**Note** You can determine the name and the current location of all files that a database uses by using the **sp\_helpfile**stored procedure:

use < database\_name>

go

sp\_helpfile

   go

* You should have exclusive access to the database that you are moving. If you have problems during the process, and if you cannot access a database that you have moved or if you cannot start SQL Server, examine the SQL Server error log and SQL Server Books Online for more information about the errors that you are experiencing.

***Note: - We have seen scenarios where this doesn’t work on RTM version. So please upgrade the SQL instance with Service pack 1 before following this article***

**Planned Relocation Practice**

**NOTE: - For this scenario to work, below steps should be done in the same order.**

**In order to move the physical files we need to follow the below steps:**

1.      Disable read-only access for all the secondary replicas.

2.      Modify the location of the data and transaction log files on all the replicas using the ALTER DATABASE…MODIFY FILE option.

3.      Perform the failover of AlwaysOn group to any synchronous replica (***Node2 in this scenario***).

***Note: This would clear all the file handles on the secondary replicas. On the new primary replica the database files will be used from the original location which can be verified using SP\_HELPDB < DBNAME>.***

4.      Move the physical files (MDF/LDF/NDF) to the new location on all the secondary replicas.

***Note: At this point the synchronization between the replicas are broken.***

5.      Initiate the database recovery using ALTER DATABASE…SET ONLINE on all the secondary replicas to resume the synchronization.

6.      Perform the failover of AlwaysOn group back to original node (***From Node2 to Node1 in this scenario***).

7.      Follow the step-4 & step-5 on the Node2 to fix the file location and resume the synchronization.

8.      Finally enable the read-only access for all the secondary replicas.

AlwaysOn Group is online as before and the data files & transaction log files are moved to the new location.

**Example**

The following example moves the AdventureWorks data file and log file to a new location as part of a planned relocation.

**Environment:**

Node1 –Default instance (Primary replica)

Node2 –Default instance (Secondary replica –Synchronous)

Node3 –Default instance (Secondary replica –Asynchronous)

AlwaysOn Database –AdventureWorks

Data file location: C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data\AdventureWorks.mdf

Log file location  : C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data\AdventureWorks.ldf

|  |
| --- |
| --- YOU MUST EXECUTE THE FOLLOWING SCRIPT IN SQLCMD MODE.  --- Select “QUERY” from the “MENU” bar and select “SQLCMD MODE”    --Disable read-only access for all the secondary replicas  :Connect NODE1    USE[master]  GO  ALTER AVAILABILITY GROUP [AdventureWorksAG] MODIFY REPLICA ON N'NODE1' WITH (SECONDARY\_ROLE(ALLOW\_CONNECTIONS = NO))  GO  ALTER AVAILABILITY GROUP [AdventureWorksAG] MODIFY REPLICA ON N'NODE2' WITH (SECONDARY\_ROLE(ALLOW\_CONNECTIONS = NO))  GO  ALTER AVAILABILITY GROUP [AdventureWorksAG] MODIFY REPLICA ON N'NODE3' WITH (SECONDARY\_ROLE(ALLOW\_CONNECTIONS = NO))  GO    --Modify the location of the data and transaction log files on all the replicas  :Connect NODE1    ALTER DATABASE [AdventureWorks] MODIFY FILE (NAME='AdventureWorks',FILENAME='D:\DATA\AdventureWorks.mdf')  go  ALTER DATABASE [AdventureWorks] MODIFY FILE (NAME='AdventureWorks\_log',FILENAME='D:\DATA\AdventureWorks\_log.ldf')  go    :Connect NODE2    ALTER DATABASE [AdventureWorks] MODIFY FILE (NAME='AdventureWorks',FILENAME='D:\DATA\AdventureWorks.mdf')  go  ALTER DATABASE [AdventureWorks] MODIFY FILE (NAME='AdventureWorks\_log',FILENAME='D:\DATA\AdventureWorks\_log.ldf')  go    :Connect NODE3    ALTER DATABASE [AdventureWorks] MODIFY FILE (NAME='AdventureWorks',FILENAME='D:\DATA\AdventureWorks.mdf')  go  ALTER DATABASE [AdventureWorks] MODIFY FILE (NAME='AdventureWorks\_log',FILENAME='D:\DATA\AdventureWorks\_log.ldf')  go      --Perform the failover of AlwaysOn group  :Connect NODE2    ALTER AVAILABILITY GROUP [AdventureWorksAG] FAILOVER;  GO      --Move the physical files (MDF/LDF/NDF) to the new location on all the secondary replicas.    :Connect NODE1    --Enable XP\_CMDSHELL  sp\_configure 'show advanced options',1  go  reconfigure  go  sp\_configure 'xp\_cmdshell',1  go  reconfigure  go    --MOVE FILES  xp\_cmdshell'move "C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data\AdventureWorks\*.\*" D:\data\'  go    --Disable XP\_CMDSHELL  sp\_configure 'show advanced options',1  go  reconfigure  go  sp\_configure 'xp\_cmdshell',0  go  reconfigure  go      :Connect NODE3  --Enable XP\_CMDSHELL  sp\_configure 'show advanced options',1  go  reconfigure  go  sp\_configure 'xp\_cmdshell',1  go  reconfigure  go    --MOVE FILES  xp\_cmdshell'move "C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data\AdventureWorks\*.\*" D:\data\'  go    --Disable XP\_CMDSHELL  sp\_configure 'show advanced options',1  go  reconfigure  go  sp\_configure 'xp\_cmdshell',0  go  reconfigure  go          --Initiate the database recovery    :Connect NODE1    ALTER DATABASE [AdventureWorks] SET ONLINE  GO    :Connect NODE3    ALTER DATABASE [AdventureWorks] SET ONLINE  GO    --Perform the failover of AlwaysOn group back to original node    :Connect NODE1    ALTER AVAILABILITY GROUP [AdventureWorksAG] FAILOVER;  GO    --To fix the file location and resume the synchronization on Node2  :Connect NODE2    --Enable XP\_CMDSHELL  sp\_configure 'show advanced options',1  go  reconfigure  go  sp\_configure 'xp\_cmdshell',1  go  reconfigure  go    --MOVE FILES  xp\_cmdshell'move "C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data\AdventureWorks\*.\*" D:\data\'  go    --Disable XP\_CMDSHELL  sp\_configure 'show advanced options',1  go  reconfigure  go  sp\_configure 'xp\_cmdshell',0  go  reconfigure  go    :Connect NODE2  ALTER DATABASE [AdventureWorks] SET ONLINE  GO      --Finally enable the read-only access for all the secondary replicas    :Connect NODE1    USE[master]  GO  ALTER AVAILABILITY GROUP [AdventureWorksAG] MODIFY REPLICA ON N'NODE1' WITH (SECONDARY\_ROLE(ALLOW\_CONNECTIONS = ALL))  GO  ALTER AVAILABILITY GROUP [AdventureWorksAG] MODIFY REPLICA ON N'NODE2' WITH (SECONDARY\_ROLE(ALLOW\_CONNECTIONS = ALL))  GO  ALTER AVAILABILITY GROUP [AdventureWorksAG] MODIFY REPLICA ON N'NODE3' WITH (SECONDARY\_ROLE(ALLOW\_CONNECTIONS = ALL))  GO |