

**Cross-cluster Migration of AlwaysOn Availability Groups for Operating System Upgrades**

SQL Server technical article

**Writers:** Min He (Microsoft), Margi Showman (Microsoft)

**Technical reviewers:** Robert Dorr, Wayne Chen, Jimmy May, Zhenyu Zhao, Sonia Parchani

**Content Program Manager:** Glenn Minch (Microsoft)

**Published:** December, 2012

**Applies to:** AlwaysOn Availability Groups on Microsoft SQL Server 2012 with Service Pack 1 (SP1)

**Summary:** SQL Server 2012 SP1 introduces support for cross-cluster migration of AlwaysOn Availability Groups deployments to another Windows Server Failover Clustering (WSFC) cluster. This document introduces the migration process and new Transact-SQL options that support migration. The document then presents planning considerations, preparation steps, and a procedure for cross-cluster migration. Finally, this document discusses possible follow-up tasks.

Copyright

This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

Some examples depicted herein are provided for illustration only and are fictitious.  No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

© 2012 Microsoft. All rights reserved.

Contents

[Introduction 5](#_Toc339295216)

[Requirements 5](#_Toc339295217)

[Terms and Definitions 5](#_Toc339295218)

[Abbreviations 5](#_Toc339295219)

[Glossary of Terms 6](#_Toc339295220)

[Overview of Cross-Cluster Migration 7](#_Toc339295221)

[Overview of Preparation of the New WSFC Cluster (No Downtime) 7](#_Toc339295222)

[Overview of Data Migration (No Downtime) 7](#_Toc339295223)

[Overview of Resource Migration (Planned Downtime) 8](#_Toc339295224)

[Planning (No Downtime) 10](#_Toc339295225)

[Restrictions 10](#_Toc339295226)

[General Guidelines 10](#_Toc339295227)

[Typical Planning Considerations 10](#_Toc339295228)

[Migration Batches 11](#_Toc339295229)

[Why Are Migration Batches Necessary? 11](#_Toc339295230)

[Recommendations 11](#_Toc339295231)

[Example: Minimizing Overlap Among Availability Groups 12](#_Toc339295232)

[Preparation of the New WSFC Cluster 12](#_Toc339295233)

[Data Migration (No Downtime) 14](#_Toc339295234)

[Recommendations 14](#_Toc339295235)

[Overview of Data Migration Steps 14](#_Toc339295236)

[Resource Migration (Planned Downtime) 15](#_Toc339295237)

[Requirements 15](#_Toc339295238)

[Restrictions 15](#_Toc339295239)

[Recommendations 16](#_Toc339295240)

[Overview of Resource Migration Steps for Using MigrateAG.ps1 16](#_Toc339295241)

[Default Steps Performed by MigrateAG.ps1 17](#_Toc339295242)

[Follow Up: After Migrating Every Availability Group in Your Current Batch (Optional) 18](#_Toc339295243)

[Related AlwaysOn Availability Groups Tasks 19](#_Toc339295244)

[Related Content 20](#_Toc339295245)

[Conclusion 20](#_Toc339295246)

[Appendix A: How To Procedures 20](#_Toc339295247)

[Procedure: Use AclClusterPermission.ps1 to Grant Or Revoke Cluster Registry Permissions 20](#_Toc339295248)

[Procedure: Pre-Staging the Windows Server 2012 Cluster Account in Active Directory Server 21](#_Toc339295249)

[Procedure: Switch the HADR Cluster Context of each Destination Instance 22](#_Toc339295250)

[Procedure: Start Data Synchronization 22](#_Toc339295251)

[Procedure: Create an XML Configuration File for the Current Batch of Availability Groups 23](#_Toc339295252)

[Describe Each Availability Group (Required) 24](#_Toc339295253)

[To Specify Availability Group Name and Destination Instances (Required) 25](#_Toc339295254)

[To Keep the Original Availability Group Online (Optional) 26](#_Toc339295255)

[To Specify New Availability Group Listener Values (Optional) 26](#_Toc339295256)

[Example 1: XML Configuration File for Using Default Listener Behavior 28](#_Toc339295257)

[Example 2: XML Configuration File for a New Listener 29](#_Toc339295258)

[Procedure: Use MigrateAG.ps1 to Migrate Availability Group Resources 30](#_Toc339295259)

# Introduction

Windows Server Failover Clustering (WSFC) does not support operating system rolling upgrades. To address this restriction, SQL Server 2012 SP1 introduces support for cross-cluster migration. A cross-cluster migration moves one or more AlwaysOn availability groups to a new version Windows Server Failover Clustering (WSFC) cluster with little downtime. Performing a cross-cluster migration enables you to maintain your service level agreements (SLAs) when you upgrade to Windows Server 2012. The success of a cross-cluster migration depends on thorough planning and preparation of the destination WSFC cluster. The actual cross-cluster migration process must be completed for each batch of availability groups that you are migrating.

This white paper describes how to perform a cross-cluster migration for availability groups.

# Requirements

* The requirements for the Windows operating system and for SQL Server are as follows.

| WSFC cluster | Windows requirement | SQL Server requirement |
| --- | --- | --- |
| Source cluster | Windows Server 2008 or Windows Server 2008 R2 | SQL Server 2012 or SQL Server 2012 SP1 (Enterprise edition or above) |
| Destination cluster | The Windows version must be the same as or newer than the Windows version of the source cluster. Cross-cluster migration of AlwaysOn Availability Groups is intended primarily for operating system upgrade to a Windows Server 2012 cluster. | A minimum of SQL Server 2012 SP1 (Enterprise edition or above)  |

* If an existing availability group is deployed on failover cluster instances (FCIs), the database administrator must create equivalent FCIs on the destination cluster before beginning to migrate the availability group. Keep in mind that on a given Windows Cluster node, only one server instance can host an availability replica for a given availability group.
* Down load the [PowerShellForOSUPG.zip](http://download.microsoft.com/download/D/2/0/D20E1C5F-72EA-4505-9F26-FEF9550EFD44/PowerShellForOSUPG.zip)[[1]](#footnote-1) file from the SQL Server 2012 SP1 download site. Uncompress the .zip file to install the Windows PowerShell scripts that support cross-cluster migration of availability groups. The script files are as follows:
* AclClusterPermission.ps1
* MigrateAG.ps1

The use of these files is described later in this document.

# Terms and definitions

## Abbreviations

| Term | Definition |
| --- | --- |
| FCI | SQL Server failover cluster instance |
| HA | High availability |
| VIP | Virtual IP address |
| VNN | Virtual network name |
| WSFC | Windows Server Failover Clustering |

## Glossary of terms

| Term | Definition |
| --- | --- |
| cross-cluster migration | The process of migrating one or more availability groups to another WSFC cluster. This process is intended primarily for migrating an AlwaysOn Availability Groups deployment to a Windows Server 2012 WSFC cluster.  |
| HADR cluster context | The WSFC cluster that manages the metadata for availability replicas hosted by a given instance of SQL Server. Migrating availability groups to another WSFC cluster requires changing the HADR cluster context of each destination instance from the destination cluster to the source cluster and, later, back from the source cluster to the destination cluster. This capability is new in SQL Server 2012 SP1. |
| source cluster | For a given availability group, the WSFC cluster that currently hosts the primary replica and availability group listener. See also destination cluster. |
| destination cluster | The WSFC cluster that will host a migrated availability group. See also, source cluster. |
| destination instance | An instance of SQL Server 2012 SP1 (either a stand-alone instance or failover cluster instance) that is installed on a node of the destination cluster and is enabled for AlwaysOn Availability Groups. Each destination instance can host one destination replica per availability group.  |
| destination replica | A secondary replica that resides on the destination cluster during availability group migration, rather than residing on the source cluster.  |
| migration batch | A group of availability groups that need to be migrated together. If a destination instance hosts destination replicas for more than one availability group, those availability groups belong to a single migration batch. For more information, see [Migration Batches](#_Migration_Batches) later in this document. |
| data migration | The process of configuring two secondary replicas (destination replicas) on the destination cluster and initiating full data synchronization with the primary replica on the source cluster. This is the first phase of migrating a given availability group. |
| availability group resource migration | The process of migrating the WSFC cluster resources of the availability groups in a given migration batch to the destination cluster. The availability group resources include the WSFC resource group and listener of each availability group listener in the batch. This is the second phase of migrating a given availability group.  |
| running exposed | A condition that exists when a single instance of SQL Server is accepting updates without any high-availability protection from a secondary replica or an FCI.  |

# Overview of cross-cluster migration

A cross-cluster migration involves migrating a set of availability groups from an existing WSFC cluster (the source cluster) to another WSFC cluster (the destination cluster). To support cross-cluster migration, SQL Server 2012 SP1 must be installed on two or more nodes of the destination cluster, and these server instances must be enabled for AlwaysOn. These server instances are known as destination instances, and the nodes that host them are known as destination nodes.

A cross-cluster migration involves the following:

* Preparation of the destination cluster (no downtime)
* Data migration (no downtime)
* Availability group resource migration (planned downtime)

## Overview of preparation of the new WSFC cluster (no downtime)

Depending on your migration plan, a number of preparation steps might be necessary. For example, to create the destination cluster by reusing some existing cluster nodes from the original WSFC cluster, a Windows cluster administrator might need to evict nodes from the original WSFC cluster. First, however, a database administrator (DBA) would need to remove any availability replicas from those nodes.

If the WSFC cluster to which you are going to migrate a set of availability groups (the destination cluster) does not exist yet, a Windows administrator needs to create the destination cluster. In this discussion, the destination cluster is assumed to be running on Windows Server 2012. But the destination cluster could be running on any Windows version that is the same as or newer than the Windows version of the source cluster.

On every node of the destination cluster that will host an availability replica (that is, on every destination node), a DBA must install an instance of SQL Server 2012 SP1 (either as a stand-alone instance or as part of a failover cluster instance). Then, the DBA must configure these server instances to support AlwaysOn Availability Groups. During availability group migration, these server instances will serve as destination instances.

In SQL Server 2012 RTM and SP1, an availability group supports up to four secondary replicas. This limitation applies to an availability group which is being migrated across two WSFC clusters. Before initiating data migration for a given availability group, the DBA must ensure that the existing availability group has no more than two secondary replicas. This enables the creation of two secondary replicas on destination instances to serve as destination replicas for HA protection.

For more information, see [Preparation of the New WSFC Cluster (No Downtime)](#_Migration_Batch_Size) later in this document.

## Overview of data migration (no downtime)

During the data-migration phase, you need to perform the following basic steps:

1. On each destination instance, switch the HADR cluster context to the source cluster. A new ALTER SERVER CONFIGURATION SET option exists for this purpose. For more information, see [Change the HADR Cluster Context of Server Instance (SQL Server)](http://msdn.microsoft.com/library/jj573601.aspx) (http://msdn.microsoft.com/library/jj573601.aspx).
2. For each availability group to be migrated in a given batch:
* Choose two destination instances.
* Seed new secondary databases on both instances.
* Create and configure a synchronous-commit secondary replica on one destination instance and an asynchronous-commit secondary replica on the other destination instance. These two secondary replicas will serve as the destination replicas for the availability group.

The following figure (Figure 1) illustrates the data-migration process for a single availability group, AG1. Note that, typically, multiple availability groups would need to be migrated as a single batch (a migration batch). For more information, see [Migration Batches](#_Migration_Batches) later in this document.



Figure 1: data migration for a single availability group

Notice that until you start to migrate the resources of a given availability group, SQL Server replication, log shipping, change data capture, and FILESTREAM and FileTable remain fully functional on the source availability databases.

For more information, see [Data Migration (No Downtime)](#_Phase_2:_Data_1) later in this document.

## Overview of resource migration (planned downtime)

This phase migrates the WSFC cluster resources of the availability groups (the availability group resources) to the destination cluster. For each availability group in the current migration block, a new primary replica must be created on the destination instance that hosts the synchronous-commit destination replica. Also, the availability group listener must be re-created on the destination cluster.

When you migrate a given availability group, its synchronous-commit destination replica becomes the new primary replica. The asynchronous-commit destination replica remains a secondary replica but, in most cases it is a synchronous-commit replica in the migrated availability group. Thereafter, you can add up to three additional secondary replicas at any time.

Note: On the source cluster, availability groups that are not part of the current migration batch continue to accept workloads.

To migrate resources for each availability group in the current migration batch, follow these steps:

1. On the source cluster, connect to the server instance that hosts the primary replica of each availability group, and then stop the update workload. This requires you to delete the availability group listener and, optionally, take the availability group offline from Windows Cluster Administrator.

Note: SQL Server 2012 SP1 introduced a new ALTER AVAILABILITY GROUP option for taking an availability group offline. If your primary instance on source cluster is still on the SQL Server 2012 RTM version, connect to a destination instance and then use the OFFLINE option to take the availability group offline. For more information, see [Take an Availability Group Offline (SQL Server)](http://msdn.microsoft.com/library/jj573599.aspx) (http://msdn.microsoft.com/library/jj573599.aspx).

1. On the destination cluster, switch the HADR cluster context of each destination instance back to the LOCAL cluster.
2. For each availability group in the migration batch, connect to the destination instance that hosts the synchronous-commit destination replica, and then re-create the availability group using its original name and the availability group listener using its original name and VNN.

**Important:** For your applications to connect seamlessly to the availability group listener of a migrated availability group, the new destination listener must use the same name and VNN as the deleted source listener. Using the same VNN also enables any FILESTREAM share to continue providing file system access under the same UNC path.

To reuse the same VNN, you need to prepare the Active Directory server before you start migration. For more information, see [Procedure: Pre-Staging the Windows Server 2012 Cluster Account in Active Directory Server](#_Procedure_(Step_6):) later in this document.

The following figure (Figure 2) illustrates the resource-migration process for a single availability group, AG1.



Figure 2: resource migration for a single availability group

For more information, see [Resource Migration (Planned Downtime)](#_Phase_3:_Availability) later in this document.

# Planning (no downtime)

The first step in planning a cross-cluster migration is preparing a migration plan. The migration plan should ensure that the destination cluster can fully support migrated availability groups. The plan should also minimize downtime during availability group resource migration. To begin, evaluate the number of availability groups hosted on your source cluster and determine where they are deployed across the cluster. This information enables you to assess the number of permanent nodes required for your destination cluster and to determine whether a secondary replica needs to be removed to meet the limit of four secondary replicas for migrating a given availability group.

## Restrictions

* The in-place operating system upgrade of an AlwaysOn Availability Groups deployment is not supported.
* If the availability replicas of several availability groups overlap on a set of destination instances, those availability groups need to be migrated together as a single batch (in a batch migration). For more information, see [Batch Migration of a Multiple Availability Groups](#_Batch_Migration_of) earlier in this document.
* If you are migrating an availability group over a SQL Server FCI, you need a destination WSFC node for each FCI node.

## General guidelines

* Avoid running exposed. Running exposed refers to the situation in which an instance of SQL Server accepts data changes on one or more databases without any HA protection. In the event of an unrecoverable node failure, data loss occurs. Ideally, your migration plan can avoid databases or server instances running exposed on either the source cluster or destination cluster. In some situations, however, you might need to make tradeoffs between long downtime, adding more redundant hardware, and running exposed.
* Make sure WSFC quorum mode and voting are properly configured on destination cluster. For more information about quorum settings, see [WSFC Quorum Modes and Voting Configuration (SQL Server)](http://msdn.microsoft.com/library/hh270280.aspx) (http://msdn.microsoft.com/library/hh270280.aspx).
* Batch migrations require extra downtime for each of the availability groups in the batch—the more availability groups with overlapping destination replicas, the longer their collective downtime during migration. By adding temporary nodes to the destination cluster, you can facilitate separating availability groups into smaller migration batches, reducing downtime. To reuse the existing hardware, you can evict existing some nodes from your existing WSFC cluster(s), upgrade them to the new Windows operating system, and then add them to your destination cluster.
* As part of your planning effort, we recommend that you document the original configuration of each availability group on the source cluster.

## Typical planning considerations

Typical planning considerations include:

* How many permanent nodes does the destination cluster need?
* How many temporary nodes does the destination cluster need for you to optimize migration batch sizes? Can you reuse existing hardware for temporary nodes?
* Which destination instances should host the destination replicas for each of the availability groups?

## Migration batches

If the availability replicas of several availability groups overlap on a set of destination instances, those availability groups need to be migrated together in a migration batch. For example, consider three availability groups (AG1, AG2, and AG3) that overlap across three destination instances (Node3\inst1, Node4\inst1, and Node5\inst1), as follows.

| Availabilitygroup | Node3\inst1 | Node4\inst1 | Node5\inst1 |
| --- | --- | --- | --- |
| AG1 | Destination replica | Destination replica |  |
| AG2 |  | Destination replica | Destination replica |
| AG3 | Destination replica |  | Destination replica |

AG1 and AG3 do not have directly overlapping replicas; however, each group overlaps with AG2. Therefore, groups AG1, AG2, and AG3 need to be migrated in a single batch.

### Why are migration batches necessary?

Migration batches are necessary because if you switch a server instance to a remote HADR cluster context, the server instance cannot host any availability replicas. Therefore, you must switch the HADR cluster context the source cluster before you create any destination replicas. On a given destination instance, the HADR cluster context must continue to point to the source cluster until every necessary destination replica has been created on the instance and any synchronous-commit destination replicas have synchronized with the primary replica. Eventually, as part of migrating availability group resources, the DBA must set the HADR cluster context back to the local, destination cluster. Thereafter, as long at the server instance is hosting any local availability replicas, its HADR cluster context cannot be switched to a remote cluster.

### Recommendations

If you need to migrate multiple availability groups, minimize their overlap on destination instances. There are several ways to minimize overlap, as follows:

* On the destination cluster, create only two destination replicas per availability group: one synchronous-commit secondary replica and one asynchronous-commit secondary replica. You can add additional secondary replicas after migration completes.
* To reduce overlap among availability groups, add one or more additional, temporary WSFC nodes to the destination cluster, and deploy a destination instance on each of these nodes.
* Avoid creating any "ring configurations" on the destination cluster. For example, if you have four destination nodes (node1, node2, node3, and node4), create two separate migration batches. Put the destination replicas of half the availability groups on node1 and node2, and put the destination replicas of the remaining availability groups on node3 and node4. For more information, see [Example: Minimizing Overlap Among Availability Groups](#_Example:_Mininizing_Overlap) later in this section.

To optimize performance after availability group migration, place some of the synchronous-commit destination replicas—the future primary replicas—on one destination instance and place the others on the other destination instance. After the migration completes, you can add additional secondary replicas and/or use planned manual failovers to move availability replicas to different WSFC nodes with insignificant downtime.

### Example: minimizing overlap among availability groups

In the following example, three availability groups—AG1, AG2, and AG3—must be migrated to a destination Windows Server 2012 WSFC cluster. This destination cluster contains an even number of nodes (Node3, Node4, Node5, and Node6). On each node, the default instance of SQL Server 2012 SP1 is enabled for AlwaysOn.

The even number of destination nodes (four) facilitates separating the example availability groups into two batches. For example:

* For AG1:
* Create the synchronous-commit secondary replica on Node3.
* Create the asynchronous-commit secondary replica on Node4.

When you re-create the availability group on the destination cluster, the primary replica will be on Node3.

* For AG2:
* Create the synchronous-commit secondary replica, on Node4.
* Create the asynchronous-commit secondary replica on Node3.

When you re-create the availability group on the destination cluster, the primary replica will be on Node4.

* For AG3:
* Create the synchronous-commit secondary replica on Node5.
* Create the asynchronous-commit secondary replica on Node6.

When you re-create the availability group on the destination cluster, the primary replica will be on Node5.

The result is two batches of availability groups on the destination cluster:

* AG1 and AG2 form one batch on Node3 and Node4.
* AG3 forms a separate batch on Node5 and Node6.

The following table summarizes the deployment of destination replicas in these two example migration batches. The four destination nodes (Node3 – Node6) are part of a Windows Server 2012 cluster.

| Availabilitygroup | Node3 | Node4 | Node5 | Node6 |
| --- | --- | --- | --- | --- |
| AG1 | Synchronous secondary | Asynchronous secondary |  |  |
| AG2 | Asynchronous secondary | Synchronous secondary |  |  |
| AG3 |  |  | Synchronous secondary | Asynchronous secondary |

## Preparation of the new WSFC cluster

Depending on your migration plan, the following preparation steps might be necessary.

| Step | Link |
| --- | --- |
| 1. Optionally, to free up hardware resources, evict one or more nodes from the source WSFC cluster. First, however, the database administrator must remove or move any availability replicas that are hosted on to-be-evicted nodes.
 | [Evict a node from the cluster](http://msdn.microsoft.com/library/cc784955%28WS.10%29.aspx) (http://msdn.microsoft.com/library/cc784955(WS.10).aspx) |
| 1. Create a Windows Server 2012 WSFC cluster (the destination cluster).
 | [Windows Server 2012 documentation](http://technet.microsoft.com/library/hh831579.aspx) (http://technet.microsoft.com/library/hh831579.aspx) |
| 1. Install an instance of SQL Server 2012 SP1 on each node of the Windows Server 2012 cluster. Keep in mind that each node can host only one AlwaysOn-enabled instance of SQL Server.
 | [MICROSOFT® SQL SERVER® 2012 SERVICE PACK 1 (SP1)](http://www.microsoft.com/download/details.aspx?id=30375) ([http://www.microsoft.com/download/details.aspx?id=30375](http://www.microsoft.com/en-us/download/details.aspx?id=30375)) |
| 1. Configure each new server instance that will host an availability replica to support AlwaysOn Availability Groups.
 | [Configuration of a Server Instance for AlwaysOn Availability Groups (SQL Server)](http://msdn.microsoft.com/library/hh510260.aspx) ([http://msdn.microsoft.com/library/hh510260.aspx](http://msdn.microsoft.com/en-us/library/hh510260.aspx))[Prerequisites, Restrictions, and Recommendations for AlwaysOn Availability Groups (SQL Server)](http://msdn.microsoft.com/library/ff878487.aspx) (http://msdn.microsoft.com/library/ff878487.aspx) |
| 1. On each node of the Windows Server 2008 R2 cluster, grant cluster registry permissions to the service account of each SQL Server 2012 SP1 instance on Windows Server 2012 node. If any server instances run under different service accounts, this step needs to be taken for each service account.
 | [Procedure: Use the AclClusterPermission.ps1 Script to Grant Or Revoke Cluster Registry Permissions](#_Procedure:_Use_AclClusterPermission) in Appendix A. |
| 1. For each availability group listener to be migrated, grant the destination Windows Server 2012 WSFC cluster full control permission on the VNN computer object in the Active Directory Server.
 | [Procedure: Pre-Staging Windows Server 2012 Cluster Account in the Active Directory Server](#_Procedure_(Step_6):) in Appendix A.  |

# Data migration (no downtime)

The data-migration phase involves creating and configuring two new secondary replicas on the destination cluster for each availability group in the migration batch.

## Recommendations

* To create the destination replicas and possibly to seed new secondary databases, we recommend that you use the [Add Replica Wizard](http://msdn.microsoft.com/library/hh213239.aspx) (http://msdn.microsoft.com/library/hh213239.aspx).
* When you add secondary replicas to an availability group, be careful to configure the new replicas to support the existing readable secondary, read-only routing, and backup on secondary configurations of the availability group. This is necessary in order to maintain these configurations.

## Overview of data migration steps

The following table summarizes the steps for performing data migration for a migration batch that consists of two availability groups (AG1 and AG2).

Table: summary of data migration for two availability groups

| Step# | Source WSFC (Windows Server 2008 R2) | Destination WSFC (Windows Server 2012) |
| --- | --- | --- |
| Node1\inst1(AG1 Primary)(AG2 Secondary) | Node2\inst1(AG1 Secondary)(AG2 Primary) | Node3\inst1 | Node4\inst1 |
| 1 |  |  | Switch HADR cluster context to source cluster. | Switch HADR cluster context to source cluster. |
| 2 |  |  | Seed new secondary databases for AG1 and AG2.**\*** | Seed new secondary databases for AG1 and AG2.**\*** |
| 3 | Use Add Replica to Availability Group Wizard to configure two secondary replicas on destination cluster. |  | **🡪** Synchronous-commit secondary replica for AG1.  | **🡪** Asynchronous-commit secondary replica for AG1. |
| 4 |  | Use Add Replica to Availability Group Wizard to configure two secondary replicas on destination cluster. | **🡪** Asynchronous-commit secondary replica for AG2.  | **🡪** Synchronous-commit secondary replica for AG2. |

**\***If database-file paths are identical on the source and destination instances, the Add Replica to Availability Group Wizard can seed the secondary databases automatically (select **Full** initial data synchronization).

# Resource migration (planned downtime)

After every synchronous-commit destination replica in the migration batch is synchronized, you can migrate the availability group resources. Resource migration entails downtime. The amount of downtime depends partly on the number of availability groups in the current migration batch. Smaller batches experience less downtime than bigger batches.

Migrating the availability group resources for a migration batch involves re-creating every availability group and its listener on the destination cluster. Typically, each availability group listener is deleted, and its VNN and VIP are reused on the new availability group listener.

To use MigrateAG.ps1 to migrate the resources of a given batch of availability groups, you need to create an XML configuration file. The values specified in your current configuration file are read by a PowerShell script (MigrateAG.ps1), which automates the actual resource migration for the current batch.

Note: For information about downloading a MigrateAG.ps1, see [Requirements](#_Requirements) earlier in this document.

## Requirements

* By default, MigrateAG.ps1 deletes the original availability group listener of each availability group and reuses the same VNN for the new listener on the destination cluster. However, for MigrateAG.ps1 to reuse a VNN, the destination cluster requires full control permissions on the object. For more information, see [Procedure: Pre-Staging the Windows Server 2012 Cluster Account in Active Directory Server](#_Procedure_(Step_6):) in Appendix A.

## Restrictions

* MigrateAG.ps1 requires that the destination instance for the current migration batch all use the same login. If any destination instances use different logins, you cannot use MigrateAG.ps1 to perform resource migration. For information about the steps involved in resource migration, see [Default Steps Performed by MigrateAG.ps1](#_Default_Steps_Performed) later in this document.
* On destination instances, the asynchronous-commit secondary databases will continue to be secondary databases in the migrated availability groups. Therefore, asynchronous-commit secondary databases must remain in the RESTORING state.

## Recommendations

Before you begin the resource-migration process, we recommend that you complete the following preparatory tasks:

1. Generate scripts for the backup jobs of every availability database on each source server instance, along with any other jobs that you require on a given database. For information about backup on secondary replicas, see [Active Secondaries: Backup on Secondary Replicas (AlwaysOn Availability Groups)](http://msdn.microsoft.com/library/hh245119.aspx) (http://msdn.microsoft.com/library/hh245119.aspx).
2. For log shipping configurations whose primary database is an availability database (on any of the availability replicas), generate scripts that will facilitate re-creation of these configurations. For information about using log shipping with AlwaysOn Availability Groups, see [Prerequisites for Migrating from Log Shipping to AlwaysOn Availability Groups (SQL Server)](http://msdn.microsoft.com/library/hh758463.aspx) (http://msdn.microsoft.com/library/hh758463.aspx).
3. To speed up database recovery operations during availability group migration, remove replication subscriptions on the AlwaysOn primary databases before you migrate availability groups. After availability group migration, reconfigure the replication publication and subscription settings. For information about using Replication with AlwaysOn Availability Groups, see [AlwaysOn Availability Groups: Interoperability (SQL Server)](http://msdn.microsoft.com/library/hh710077.aspx) (http://msdn.microsoft.com/library/hh710077.aspx).

## Overview of resource migration steps for using MigrateAG.ps1

The following table summarizes the steps for migrating the resources of the availability groups in a given migration batch.

| Step | Link |
| --- | --- |
| 1. Create an XML configuration file for the availability groups in the current migration batch (no downtime).
 | [Procedure: Create an XML Configuration File for the Current Batch of Availability Groups](#_Procedure:_Create_an), in Appendix A. |
| 1. Verify that all synchronous-commit destination replicas in the migration batch are SYNCHRONIZED (no downtime).
 | [sys.dm\_hadr\_availability\_replica\_states](http://msdn.microsoft.com/library/ff878537.aspx) (http://msdn.microsoft.com/library/ff878537.aspx) |
| 1. To perform the actual resource migration, run the MigrateAG.ps1 PowerShell script on the XML configuration that you created for the current migration batch (planned downtime).

**Caution:** The MigrateAG.ps1 script automates migration of availability group resources, reducing configuration complexity and avoiding errors. However, this assumes that your configuration file uses the correct XML elements and specifies valid values. Therefore, we recommend that you use MigrateAG.ps1 only to generate scripts, not run them (this is the default behavior). This gives you a chance to review the generated scripts before you run them. | [Procedure: Use MigrateAG.ps1 to Migrate Availability Group Resources](#_Procedure:_Use_MigrateAG.ps1), in Appendix A |

### Default steps performed by MigrateAG.ps1

The following table summarizes the resource migration of the example availability groups (AG1 and AG2). The XML configuration file for these two example availability groups is described in [Example 1: XML Configuration File for Using Default Listener Behavior](#_Example_1:_), in Appendix A.

Table: summary of availability group resource migration

| Step# | Source WSFC (Windows Server 2008 R2) | Destination WSFC (Windows Server 2012) |
| --- | --- | --- |
| Node1\inst1(AG1 Primary)(AG2 Secondary) | Node2\inst1(AG1 Secondary)(AG2 Primary) | Node3\inst1 | Node4\inst1 |
| 1 | Delete AG1's availability group listener. | Delete AG2's availability group listener. | 🡨 Downtime starts on AG1. | 🡨 Downtime starts on AG2. |
| 2 | Take AG1 offline. | Take AG2 offline. |  |  |
| 3 |  |  | Switch HADR Cluster Context to LOCAL. | Switch HADR Cluster Context to LOCAL. |
| 4 |  |  | After AG1's synchronous-commit destination replica synchronizes, recover its secondary databases. | After AG2's synchronous-commit destination replica synchronizes, recover its secondary databases. |
| 5 |  |  | Create availability group AG1, with primary replica hosted by Node3. | Create availability group AG2, with primary replica hosted by Node4. |
| 6 |  |  | Join secondary replica to AG2. | Join secondary replica to AG1. |
| 7 |  |  | Add listener to AG1 using original name (AG1\_Listener) and VNN.🡨 Downtime ends. | Add listener to AG2 using original name (AG2\_Listener) and VNN.🡨 Downtime ends. |

# Follow up: after migrating every availability group in your current batch (optional)

| Follow-up task | Links |
| --- | --- |
| After you run MigrateAG.ps1, we recommend that verify your new availability group configurations. | [Monitoring of Availability Groups (SQL Server)](http://msdn.microsoft.com/library/ff877954.aspx) (http://msdn.microsoft.com/library/ff877954.aspx) |
| SQL Server Agent jobs:* On the destination instance, re-create jobs for the databases that you have migrated.
* On the source server instance, delete the jobs of migrated databases.
 | * [Create Jobs](http://technet.microsoft.com/library/ms186273.aspx) (http://technet.microsoft.com/library/ms186273.aspx)
* [Delete Jobs](http://technet.microsoft.com/library/hh510242.aspx) (http://technet.microsoft.com/library/hh510242.aspx)
 |
| Optionally, remove existing SQL Server replication, change data capture, and/or log shipping configurations on the source WSFC cluster, and then re-create these configurations on the new WSFC cluster. Removing old configurations avoids potential conflicts with new configurations. | [Configure Replication for AlwaysOn Availability Groups (SQL Server)](http://technet.microsoft.com/library/hh710046.aspx) ([http://technet.microsoft.com/library/hh710046.aspx](http://technet.microsoft.com/en-us/library/hh710046.aspx))[Replication, Change Tracking, Change Data Capture, and AlwaysOn Availability Groups (SQL Server)](http://msdn.microsoft.com/library/hh403414.aspx) (http://msdn.microsoft.com/library/hh403414.aspx)[Prerequisites for Migrating from Log Shipping to AlwaysOn Availability Groups (SQL Server)](http://msdn.microsoft.com/library/hh758463.aspx) (http://msdn.microsoft.com/library/hh758463.aspx) |

# Related AlwaysOn Availability Groups asks

[Add a Secondary Replica to an Availability Group (SQL Server)](http://technet.microsoft.com/library/hh213247.aspx) ([http://technet.microsoft.com/library/hh213247.aspx](http://technet.microsoft.com/en-us/library/hh213247.aspx))

[Change the HADR Cluster Context of Server Instance (SQL Server)](http://msdn.microsoft.com/en-us/library/jj573601.aspx) (http://msdn.microsoft.com/library/jj573601.aspx)

[Configure Replication for AlwaysOn Availability Groups (SQL Server)](http://technet.microsoft.com/library/hh710046.aspx) ([http://technet.microsoft.com/library/hh710046.aspx](http://technet.microsoft.com/en-us/library/hh710046.aspx))

[Create a Database Mirroring Endpoint for AlwaysOn Availability Groups (SQL Server PowerShell)](http://technet.microsoft.com/library/hh510204.aspx) ([http://technet.microsoft.com/library/hh510204.aspx](http://technet.microsoft.com/en-us/library/hh510204.aspx))

[Enable and Disable AlwaysOn Availability Groups (SQL Server)](http://technet.microsoft.com/library/ff878259.aspx) (http://technet.microsoft.com/library/ff878259.aspx)

[Join a Secondary Database to an Availability Group (SQL Server)](http://technet.microsoft.com/library/ff878535.aspx) ([http://technet.microsoft.com/library/ff878535.aspx](http://technet.microsoft.com/en-us/library/ff878535.aspx))

[Join a Secondary Replica to an Availability Group (SQL Server)](http://technet.microsoft.com/en-us/library/ff878473.aspx) ([http://technet.microsoft.com/library/ff878473.aspx](http://technet.microsoft.com/en-us/library/ff878473.aspx))

[Manually Prepare a Secondary Database for an Availability Group (SQL Server)](http://technet.microsoft.com/library/ff878349.aspx) (http://technet.microsoft.com/en-us/library/ff878349.aspx)

[Monitoring of Availability Groups (SQL Server)](http://msdn.microsoft.com/library/ff877954.aspx) (http://msdn.microsoft.com/library/ff877954.aspx)

[Recover a Database Without Restoring Data (Transact-SQL)](http://msdn.microsoft.com/library/ms188696.aspx) (http://msdn.microsoft.com/library/ms188696.aspx)

[Remove a Secondary Replica from an Availability Group (SQL Server)](http://msdn.microsoft.com/library/hh213149.aspx) ([http://msdn.microsoft.com/library/hh213149.aspx](http://msdn.microsoft.com/en-us/library/hh213149.aspx))

[Take an Availability Group Offline (SQL Server)](http://msdn.microsoft.com/library/jj573599.aspx) (http://msdn.microsoft.com/library/jj573599.aspx).

[Use the AlwaysOn Dashboard (SQL Server Management Studio)](http://msdn.microsoft.com/library/hh213474.aspx) (http://msdn.microsoft.com/library/hh213474.aspx)

# Related content

* [Windows Server Failover Clustering (WSFC) with SQL Server](http://technet.microsoft.com/library/hh270278.aspx) ([http://technet.microsoft.com/library/hh270278.aspx](http://technet.microsoft.com/en-us/library/hh270278.aspx))
* [Failover Cluster Step-by-Step Guide: Configuring Accounts in Active Directory](http://technet.microsoft.com/library/cc731002%28WS.10%29.aspx) (http://technet.microsoft.com/library/cc731002(WS.10).aspx)

# Conclusion

SQL Server 2012 SP1 introduces support for the cross-cluster migration of availability groups from an existing WSFC cluster (the source cluster) to another WSFC cluster (the destination cluster). The cross-cluster migration process involves preparing the destination cluster and deploying destination replicas in a configuration that minimizes downtime during the migration of availability group resources to the destination cluster. This document describes the migration phases and, in appendix A, describes how to perform tasks that are specific to the cross-cluster migration process.

**For more information:**

* [SQL Server Web site](http://www.microsoft.com/sqlserver/)

(<http://www.microsoft.com/sqlserver/>)

* [SQL Server TechCenter](http://technet.microsoft.com/sqlserver/)

([http://technet.microsoft.com/sqlserver/](http://technet.microsoft.com/en-us/sqlserver/))

* [SQL Server DevCenter](http://msdn.microsoft.com/en-us/sqlserver/)

([http://msdn.microsoft.com/sqlserver/](http://msdn.microsoft.com/en-us/sqlserver/))

**Feedback requested:** Did this paper help you? Please give us your feedback. Tell us on a scale of 1 (poor) to 5 (excellent), how would you rate this paper and why have you given it this rating? For example:

* Are you rating it high due to having good examples, excellent screen shots, clear writing, or another reason?
* Are you rating it low due to poor examples, fuzzy screen shots, or unclear writing?

This feedback will help us improve the quality of white papers we release.

Send feedback.

# Appendix A: How-to procedures

This appendix contains the following procedures:

* [Use AclClusterPermission.ps1 to Grant Or Revoke Cluster Registry Permissions.](#_Procedure:_Use_AclClusterPermission)
* [Pre-Staging the Windows Server 2012 Cluster Account in Active Directory Server](#_Procedure_(Step_6):)
* [Switch the HADR Cluster Context of each Destination Instance.](#_Procedure:_Switch_the)
* [Start Data Synchronization](#_Procedure:_Start_Data)
* [Create an XML Configuration File for the Current Batch of Availability Groups](#_Procedure:_Create_an)
* [Use MigrateAG.ps1 to Migrate Availability Group Resources](#_Procedure:_Use_MigrateAG.ps1)

## Procedure: use AclClusterPermission.ps1 to grant or revoke cluster registry permissions

The AclClusterPermission.ps1 PowerShell script is included in the PowerShellForOSUPG.zip file on the Download site. For more information, see [Requirements](#_Requirements) earlier in this document.

To grant cluster and cluster registry permissions

1. Run PowerShell as a member of the Administrators group.
2. To grant Windows Server 2012 cluster permissions from the service account of each destination instance, run the AclClusterPermission.ps1 PowerShell script on each host destination node, as follows.

AclClusterPermission.ps1 -Action add –Account service\_account\_name

To revoke cluster registry permissions

1. Run PowerShell as a member of the Administrators group.
2. To revoke Windows Server 2012 cluster permissions from the service account of each destinations instance, run the AclClusterPermission.ps1 PowerShell script on each host destination node, as follows.

AclClusterPermission.ps1 –Action clean –Account service\_account\_name

Note: to show the PowerShell script usage, use this command. AclClusterPermission.ps1 –Help

## Procedure: prestaging the Windows Server 2012 cluster account in Active Directory Server

By default, as part of migrating resources for a given availability group, the MigrateAG.ps1 PowerShell script deletes the original availability listener from the source cluster. The script then attempts to reuse the VNN and VIP of the deleted listener for the new listener on the destination cluster. However, deleting a listener does not delete its VNN computer object from the Active Directory Server. To enable the script to reuse the VNN, an Active Directory Server administrator needs to grant the destination cluster FULL control permissions on the VNN computer objects of each availability group listener that is to be deleted from the source cluster.

If the destination cluster lacks FULL control permissions on a given VNN object, you risk significantly increased downtime. When it attempts to reuse the VNN of a deleted listener, the MigrateAG.ps1 script encounters a “computer object already exists” error, and the create listener operation fails.

To grant Full Control permission on a VNN computer object in the Active Directory Server

).

| Steps |
| --- |
| 1. On a domain controller, on the Start menu, point to Administrative Tools, and then click Active Directory Users and Computers.
 |
| 1. In the console tree, right-click Computers or the default container in which computer accounts are created in your domain.
 |
| 1. Find the VNN computer object of the availability group listener that you want to migrate.
 |
| 1. On the View menu, make sure that Advanced Features is selected. With Advanced Features selected, you can see the Security tab in the properties of accounts (objects).
 |
| 1. Right-click the VNN computer account, and then click Properties.
 |
| 1. On the Security tab, click Add.
 |
| 1. Click Object Types and make sure that Computers is selected. Then click OK.
 |
| 1. Under Enter the object name to select, type the cluster name account, and then click OK. If a message appears, saying that you are about to add a disabled object, click OK.
 |
| 1. Make sure that the cluster name account is selected. Then, next to Full Control, select the Allow check box.
 |
| 1. Repeat Steps 3–9 for the VNN computer object of each availability group listener that you want to migrate.
 |

For screen shots, see [Failover Cluster Step-by-Step Guide: Configuring Accounts in Active Directory](http://technet.microsoft.com/en-us/library/cc731002%28v%3DWS.10%29.aspx#BKMK_steps_precreating2) ([http://technet.microsoft.com/library/cc731002(v=WS.10).aspx#BKMK\_steps\_precreating2](http://technet.microsoft.com/library/cc731002%28v%3DWS.10%29.aspx#BKMK_steps_precreating2)

## Procedure: switch the HADR cluster context of each destination instance

On every destination instance:

| Step | Link |
| --- | --- |
| 1. Switch the HADR cluster context to the Windows Server 2008 R2 cluster that is currently hosting the availability group. This step enables these server instances to host availability replicas for availability groups on the source cluster.
 | [Change the HADR Cluster Context of Server Instance (SQL Server)](http://msdn.microsoft.com/en-us/library/jj573601.aspx) (http://msdn.microsoft.com/en-us/library/jj573601.aspx) |

## Procedure: start data synchronization

**Important**: Availability group data synchronization cross two different clusters is designed only for Windows upgrade scenarios. Microsoft does not support long-term production use of this feature.

For each availability group in the current migration batch, choose two destination instances to host a destination replica. On each destination instance, seed secondary databases for the availability group. Then create a secondary replica on each destination replica. These secondary replicas act as your destination replicas during resource migration. One of the destination replicas must use synchronous-commit availability mode. The other destination replica can use asynchronous-commit availability mode.

The following table summarizes these steps.

|  |  |
| --- | --- |
| Step | Link |
| 1. Seed secondary databases:
2. On the primary replica, back up the primary databases and transaction logs.
3. On each instance of SQL Server 2012 SP1 that is to host a destination replica for the availability group, restore these backups (using RESTORE WITH NORECOVERY).
 | [Manually Prepare a Secondary Database for an Availability Group (SQL Server)](http://msdn.microsoft.com/library/ff878349.aspx) (http://msdn.microsoft.com/library/ff878349.aspx)Note: If database-file paths are identical on source and destination instances, the Add Replica to Availability Group Wizard can seed the secondary databases automatically (select **Full** initial data synchronization).  |
| 1. Connect to the primary replica, and then add a secondary replica to serve as a destination replica on each of the two destination instances. Use synchronous-commit availability mode for one destination replica and asynchronous-commit availability mode for the other.

**Note:** The synchronous-commit destination replica becomes the new primary replica after migration.  | [Add a Secondary Replica to an Availability Group (SQL Server)](http://msdn.microsoft.com/library/hh213247.aspx) (http://msdn.microsoft.com/library/hh213247.aspx)[Use the Add Replica to Availability Group Wizard (SQL Server Management Studio)](http://msdn.microsoft.com/library/hh213239.aspx) (http://msdn.microsoft.com/library/hh213239.aspx) |
| 1. Connect to each destination instance, and then join the destination replica and its secondary databases to the availability group.
 | [Join a Secondary Replica to an Availability Group (SQL Server)](http://msdn.microsoft.com/library/ff878473.aspx) ([http://msdn.microsoft.com/library/ff878473.aspx](http://msdn.microsoft.com/en-us/library/ff878473.aspx))[Join a Secondary Database to an Availability Group (SQL Server)](http://msdn.microsoft.com/library/ff878535.aspx) (http://msdn.microsoft.com/library/ff878535.aspx)Note: The Add Replica to Availability Group Wizard automatically joins the new secondary replica to the availability group, and it optionally, joins its secondary databases to the availability group.  |

## Procedure: create an XML configuration file for the current batch of availability groups

Before you run MigrateAG.ps1 for a given migration batch, you must create an XML configuration file that describes every availability group in the batch. MigrateAG.ps1 uses the configuration values to generate two scripts for each availability group, as follows:

* Teardown\_ag\_name.ps1
* CreateAG\_ag\_name.ps1

ag\_name stands for the name of an availability group.

These generated scripts perform the actual resource migration. By default, MigrateAG.ps1 simply generates these scripts, without running them. You then run them manually when you are ready to perform the actual resource migration. We recommend using this default behavior and taking time to review your scripts before you run them.

The high-level XML syntax for this file is presented here. Ellipses (**…**) denote omitted child elements that are described later in this section. Note that the elements that you need to use in a given configuration file depend on various factors that are discussed later in this section.

**<AgMigration>**

 **<MigrateConfig>**

 **<AvailabilityGroup>**

 <Name>group\_name</Name>

 <ReplicasOnDestinationWSFC> … </ReplicasOnDestinationWSFC>

 <Offline>Y</Offline>

 <ListenerOnDestinationWSFC> … </ListenerOnDestinationWSFC>

 **</AvailabilityGroup>**

 **</MigrateConfig>**

**</AgMigration>**

This section describes how to create an XML configuration file, as follows:

* [Describe each availability group (required)](#_Describe_Each_Availability)

### Describe each availability group (required)

The **MigrateConfig** element block must contain a different **AvailabiltyGroup** element block for each availability group in the current migration batch. The basic syntax is as follows.

 **<MigrateConfig>**

 **<AvailabilityGroup>**

 <Name>group\_name</Name>

 <ReplicasOnDestinationWSFC> … </ReplicasOnDestinationWSFC>

 <Offline>Y</Offline>

 <ListenerOnDestinationWSFC> … </ListenerOnDestinationWSFC>

 **</AvailabilityGroup>**

 **<AvailabilityGroup>**

 …

 **</AvailabilityGroup>**

 …

 **</MigrateConfig>**

For a given availability group, the **AvailabiltyGroup** element block must specify the name and destination instances of the availability group. In addition, an **AvailabilityGroup** element might contain the following optional elements:

* The **Offline** element. To keep the original availability group online, specify the Offline element with a value of 'N' (no). For more information, see [To Keep the Original Availability Group Online (Optional)](#_To_Keep_the) later in this section.
* The **ListenerOnDestinationWSFC** element block. To retain the original (source) availability group listener, specify the **ListenerOnDestinationWSFC** element block. For more information, see [To Specify New Availability Group Listener Values (Optional)](#_To_Specify_New) later in this section.

#### To specify availability group name and destination instances (required)

For each availability group in the current migration batch, you must specify the availability group name. You must also specify the destination instance that will host the new primary replica and the destination instance that will host the new secondary replica. The **AvailabilityGroup** tags for this step are as follows.

 **<AvailabilityGroup>**

 <Name>group\_name</Name>

 <ReplicasOnDestinationWSFC>

 <SQLInstance isPrimary="Y">server\_instance\_1</SQLInstance>

 <SQLInstance isPrimary="N">server\_instance\_2</SQLInstance>

 </ReplicasOnDestinationWSFC>

 …

 **</AvailabilityGroup>**

The following table describes how to customize these parameters. For information about specifying server instance names, see [CREATE AVAILABILITY GROUP (Transact-SQL)](http://msdn.microsoft.com/library/ff878399.aspx) (http://msdn.microsoft.com/library/ff878399.aspx).

|  |  |
| --- | --- |
| XML tag | Description |
| <Name>group\_name</Name> | Sets the availability group name. Replace group\_name with the availability group name. This name must be the same as the name of the original availability group on the source cluster. |
| <ReplicasOnDestinationWSFC> | Begins the description of the host destination instances.  |
| <SQLInstance isPrimary="Y">server\_instance\_1</SQLInstance> | Sets the server instance for the primary destination. Replace server\_instance\_1 with the destination instance that currently hosts the synchronous-commit secondary replica. Ensure that this replica is in “synchronized state” before you run the MigrateAG.ps1 script. |
| <SQLInstance isPrimary="N">server\_instance\_2</SQLInstance> | Sets the server instance for a secondary destination. Replace server\_instance\_2 with the destination instance that currently hosts the asynchronous-commit secondary replica. |
| </ReplicasOnDestinationWSFC> | Ends the description of the host destination instances. |

#### To keep the original availability group online (optional)

If all clients connect to the availability group listener, you have the option of keeping the availability group online. To keep an availability group online, specify the **Offline** element and set the value to 'N' (no), as follows.

 **<AvailabilityGroup>**

 …

 <Offline>**N**</Offline>

 …

 **</AvailabilityGroup>**

#### To specify new availability group listener values (optional)

By default, the migration script deletes the original availability group listener on the source cluster and uses the same VNN and VIP on the destination cluster. If want this default behavior, omit the **ListenerOnDestinationWSFC** element block from the **AvailabilityGroup** element block.

However, you might prefer to retain the original listener and use a new VNN and new IP address (or addresses) for your new listener. This approach might work well if, for example, you have a centralized client connection string for each availability group listener and you can easily modify these strings to use a new VNN. To retain the original listener, specify a **ListenerOnDestinationWSFC** element block for the new availability group listener. Be sure to specify the **DeleteOldListener** element and set the value to 'N' (no). Replace the parameter names with actual values.

Important: If you retain the original listener, you must take the availability group OFFLINE (the default behavior of MigrateAG.ps1).

The **ListenerOnDestinationWSFC** element block contains the following elements:

 **<AvailabilityGroup>**

 …

 <**ListenerOnDestinationWSFC**>

 <DeleteOldListener>**N**</DeleteOldListener>

 <DnsName>dns\_name</DnsName>

 <Port>listener\_port</Port>

 <DHCP>

 <SubnetIP>four\_part\_ipv4\_address</SubnetIP>

 <SubnetMask>four\_part\_ipv4\_mask</SubnetMask>

 </DHCP>

 <IPAddress>

 <IPV4>

 <IPV4Address>four\_part\_ipv4\_address</IPV4Address>

 <IPV4Mask>four\_part\_ipv4\_mask</IPV4Mask>

 </IPV4>

 <IPV6>ipv6\_address</IPV6>

 </IPAddress>

 </**ListenerOnDestinationWSFC**>

 **</AvailabilityGroup>**

The following table describes the parameters of the **ListenerOnDestinationWSFC** element block. For more information about the availability group listener parameters, see [CREATE AVAILABILITY GROUP (Transact-SQL)](http://msdn.microsoft.com/library/ff878399.aspx) (http://msdn.microsoft.com/library/ff878399.aspx).

|  |  |
| --- | --- |
| Tag and parameter | Description |
| <DeleteOldListener>N</DeleteOldListener> | Retains the original availability group listener. |
| <DnsName>dns\_name</DnsName> | Sets the new availability group listener. Replace dns\_name with a unique DNS host name for the new availability group listener. |
| <Port>listener\_port</Port> | Sets a port for the new availability group listener. If the new listener will be using IP addressing, optionally replace listener\_port with the port number to be used by the new availability group listener. The default port number, 1433, is supported. However, if you have security concerns, we recommend that you use a different port number. |
| <DHCP> | Begins the description of a DHCP address, if used. <DHCP>and <IPAddress> are mutually exclusive. |
| <SubnetIP>four\_part\_ipv4\_address</SubnetIP> | Specifies the IPv4 four-part address. For example, 10.120.19.155. |
| <SubnetMask>four\_part\_ipv4\_mask</SubnetMask> | Specifies the IPv4 four-part mask. For example, 255.255.254.0. |
| </DHCP> | Ends the DHCP address description. |
| <IPAddress> | Begins the description of one or more IP addresses, if used. <IPAddress> and <DHCP> are mutually exclusive.Note that for a given availability group listener, you can specify multiple IP addresses if they are in different subnets. Use a separate <IPV4> or <IPV6> tag for each IP address. |
| <IPV4> | Begins the IPV4 address.  |
| <IPV4Address>four\_part\_ipv4\_address</IPV4Address> | Specifies the IPv4 four-part address. For example, 10.120.19.155. |
| <IPV4Mask>four\_part\_ipv4\_mask</IPV4Mask> | Specifies the IPv4 four-part mask. For example, 255.255.254.0. |
| </IPV4> | Ends the IPV4 address. |
| <IPV6>ipv6\_address</IPV6> | Specifies the IPV6 address. For example, 2001::4898:23:1002:20f:1fff:feff:b3a3. |
| </IPAddress> | Ends the IP address description. |

### Example 1: XML configuration file for using default listener behavior

This example shows the XML configuration file for two example availability groups, AG1 and AG2. Their destination instances reside on Node3 and Node4 of the destination cluster. For AG1, the synchronous-commit destination replica is hosted by a server instance named inst1 on Node3. For AG2, the synchronous-commit destination replica is hosted by a server instance named inst1 on Node4.

The availability group will be taken offline, so the **Offline** element is omitted.

The original availability group listener will be deleted and its VNN and VIPs will be reused for the new listener, so the **ListenerOnDestinationWSFC** element block is omitted. If you reuse the VNN, grant the destination cluster Full control permission on the VNN computer object in the Active Directory Server.

<AgMigration>

<MigrateConfig>

<AvailabilityGroup>

 <Name>AG1</Name>

 <ReplicasOnDestinationWSFC>

 <SQLInstance isPrimary="Y">node3\inst1</SQLInstance>

 <SQLInstance isPrimary="N">node4\inst1</SQLInstance>

 </ReplicasOnDestinationWSFC>

</AvailabilityGroup>

<AvailabilityGroup>

 <Name>AG2</Name>

 <ReplicasOnDestinationWSFC>

 <SQLInstance isPrimary="Y">node4\inst1</SQLInstance>

 <SQLInstance isPrimary="N">node3\inst1</SQLInstance>

 </ReplicasOnDestinationWSFC>

</AvailabilityGroup>

</MigrateConfig>

</AgMigration>

### Example 2: XML configuration file for a new listener

The following example configuration file includes all of the XML elements. The configuration file is for a two example availability groups, AG3 and AG4.

In this example, the **AvailabilityGroup** element block specifies the following elements for both availability groups:

* The **Offline** element, which is set to 'N' to keep the source availability group online.
* The **Name** element (AG3 or AG4).
* The **SQLInstance** element, which specifies the destination replicas. The new primary replicas are the synchronous-commit destination replicas. The new secondary replicas are the asynchronous-commit destination replicas.
* The **ListenerOnDestinationWSFC** element block, which is specified with **DeleteOldListener** set to 'N'. The **DnsName** element specifies the name of the new listener. The optional **Port** element specifies a nondefault listening port (7777). Finally, for each availability group, the **IPAddress** element specifies two IP4 addresses (in different subnets).

<AgMigration>

 <MigrateConfig>

 <AvailabilityGroup>

 <Name>AG3</Name>

 <ReplicasOnDestinationWSFC>

 <SQLInstance isPrimary="Y">node5</SQLInstance>

 <SQLInstance isPrimary="N">node6</SQLInstance>

 </ReplicasOnDestinationWSFC>

 <Offline>N</Offline>

 <ListenerOnDestinationWSFC>

 <DeleteOldListener>Y</DeleteOldListener>

 <DnsName>AG3\_ListenerWin2012</DnsName>

 <Port>7777</Port>

 <IPAddress>

 <IPV4>

 <IPV4Address>10.120.19.0</IPV4Address>

 <IPV4Mask>255.255.254.0</IPV4Mask>

 </IPV4>

 <IPV4>

 <IPV4Address>10.120.19.155</IPV4Address>

 <IPV4Mask>255.255.254.0</IPV4Mask>

 </IPV4>

 </IPAddress>

 </ListenerOnDestinationWSFC>

 </AvailabilityGroup>

 <AvailabilityGroup>

 <Name>AG4</Name>

 <ReplicasOnDestinationWSFC>

 <SQLInstance isPrimary="Y">node6\inst1</SQLInstance>

 <SQLInstance isPrimary="N">node5\inst1</SQLInstance>

 </ReplicasOnDestinationWSFC>

 <Offline>N</Offline>

 <ListenerOnDestinationWSFC>

 <DeleteOldListener>Y</DeleteOldListener>

 <DnsName>AG4\_ListenerWin2012</DnsName>

 <Port>7777</Port>

 <IPAddress>

 <IPV4>

 <IPV4Address>10.120.19.1</IPV4Address>

 <IPV4Mask>255.255.254.0</IPV4Mask>

 </IPV4>

 <IPV4>

 <IPV4Address>10.120.19.156</IPV4Address>

 <IPV4Mask>255.255.254.0</IPV4Mask>

 </IPV4>

 </IPAddress>

 </ListenerOnDestinationWSFC>

 </AvailabilityGroup>

 </MigrateConfig>

</AgMigration>

## Procedure: use MigrateAG.ps1 to migrate availability group resources

This procedure shows you how to run MigrateAG.ps1, which is a Windows PowerShell script file that helps you automate migration. It starts with the syntax and arguments, and then it describes the steps you take to use MigrateAG.ps1 in your own environment.

MigrateAG.ps1 generates two scripts; it does not perform the migration unless you specify otherwise. These generated scripts perform the actual resource migration. By default, MigrateAG.ps1 simply generates these scripts without performing any actual migration steps. We recommend that you review your generated scripts before running them. When you are ready to perform the actual resource migration, run the generated scripts manually, starting with Teardown\_ag\_name.ps1, followed by CreateAG\_ag\_name.ps1.

For each availability group in a given migration batch, the MigrateAG.ps1 script generates two scripts:

* Teardown\_ag\_name.ps1
* CreateAG\_ag\_name.ps1

ag\_name is the name of an availability group.

Here is the syntax for running MigrateAG.ps1.

MigrateAG.ps1 [-Help] [-SqlAuth [-LoginId -Password]] [-Migrate [-ParallelJob]] [-ScriptPath PathForScriptsOutput] -ConfigFile ConfigXMLFilePath

**–ConfigFile** is the only mandatory argument; use or omit other options based on your requirements and your environment. Here are the parameter descriptions:

* **-Help** lists the syntax.
* **-SqlAuth**, specifies that SQL Server Authentication is used. By default the script uses Windows authentication.
* **-LoginId** contains the login ID, if you specify SQL Server Authentication.
* **-Password** contains the password for the login ID, if you specify SQL Server Authentication. If you not set**–LoginId** and –**Password** in the script, the system prompts you for this information.
* **-Migrate** specifies that the migration should be completed beside script generation. By default only scripts are produced.
* **-ParallelJob** specifies migration running in PowerShell jobs in parallel. You may want to specify this option if you have multiple AGs in your migration batch. It is only used if the **–Migrate** option specified.
* **-ScriptPath** contains the output path of script files generated. By default they are put into the current working directory.
* **-ConfigFile** contains the configuration XML. If relative path is used, the script looks in the current working directory.

To run the MigrateAG script

1. Run SQL Server PowerShell as Administrator.
2. To allow script execution, run the following SQL Server PowerShell command:

Set-ExecutionPolicy –ExecutionPolicy RemoteSigned –Scope Process

1. 3. Call MigrateAG.ps1 with configuration files and other optional parameters specified.
* The following example use a configuration file named AgMigrateConfig01.xml, which resides in the C:\PowerShellForAgMigration directory.

MigrateAG.ps1 –ConfigFile C:\PowerShellForAgMigration\AgMigrateConfig01.xml

* Use SQL Server Authentication and migrate the availability groups with parallel job styles. The scripts are output to C:\scripts.

MigrateAG.ps1 –SqlAuth –Migrate –ParalleJob –ScriptPath c:\scripts\ -ConfigFile C:\PowerShellForAgMigration\AgMigrateConfig01.xml

1. http://download.microsoft.com/download/D/2/0/D20E1C5F-72EA-4505-9F26-FEF9550EFD44/PowerShellForOSUPG.zip [↑](#footnote-ref-1)