

InfoPlus ID: D12-NN

**Setup of AlwaysOn Availability Group**

Version 1.0

|  |  |
| --- | --- |
| **KCT Number** | KCT0003604 |
| **KCT Name** | Corporate Production – Build multi-subnet SQL Always-On DR cluster between Clustered SQL Instances |
| **Document Version** | Version 1 |
| **Service Owner** | SQLOps |
| **Approval Method** | SQLOpEsc |
| **Last Updated By** | Yusuf Anis |
| **Last Updated On** | July 7th, 2014 |

This solution provides HA via a local SQL Failover Cluster Instance and provides Disaster Recovery via an Availability Group to a second FCI in a secondary data center via Asynchronous synchronization.

Table of Contents

[Purpose 3](#_Toc366667159)

[Audience 3](#_Toc366667160)

[Objective 3](#_Toc366667161)

[Ticket Flow 3](#_Toc366667162)

[Technical Scope for Execution 3](#_Toc366667163)

[Scoping Details / Template 4](#_Toc366667164)

[Pre-Execution Steps for creating new AlwaysOn Availability Group 5](#_Toc366667165)

[Execution Steps 5](#_Toc366667166)

[Dummy names & configuration settings used in this setup. 5](#_Toc366667167)

[Preview of the Setup 6](#_Toc366667168)

[Preparation Steps 6](#_Toc366667170)

[Availability Group Creation 8](#_Toc366667171)

[Adding a Listener to an AOAG 19](#_Toc366667172)

[Quorum Voting in DR Setup 27](#_Toc366667173)

[Escalation Matrix 28](#_Toc366667174)

[Appendix 28](#_Toc366667175)

[Document Administration Page 28](#_Toc366667176)

# Purpose

<http://infoplus30/GetDocument.aspx?DocumentID=D12-LV>

# Audience

<http://infoplus30/GetDocument.aspx?DocumentID=D12-LV>

# Objective

<http://infoplus30/GetDocument.aspx?DocumentID=D12-LV>.

# Ticket Flow

<http://infoplus30/GetDocument.aspx?DocumentID=D12-KX>

# Technical Scope for Execution

**In Scope:**

This document has following scope of activities for the subject KCT.

1. Create a New SQL Server 2012 AlwaysOn Availability Group with instances residing in different subnets.

These changes will be a **scheduled activity** (**SHOULD ONLY BE EXECUTED DURING PROVIDED DOWNTIME WINDOW. ANY DOUBT ON DOWNTIME WINDOW SHOULD GET CLARIFIED WITH REQUESTOR BEFORE EXECUTION**) specified in the RFC.

1. Configuring cluster properties of Availability Group resources, focusing on requirements for multi-subnet.

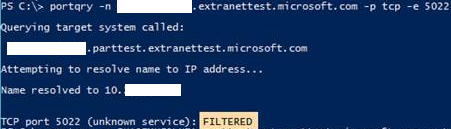
**Out of Scope:**

Following activities are out of scope.

1. Adding Nodes to an existing Availability Group
2. Creating the Windows Failover Cluster with File Share Quorum
3. Synchronizing additional User Databases within an Availability Group

# Scoping Details / Template

1. Verify if servers in request are pinging and TS from local machine.
2. From all the hosts involved, verify port access to all the other hosts. You may use telnet or portqry for this.



1. Use this template to capture details. All the requesters are expected to provide AlwaysOn requests using this template.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ticket Info** | | | | |
| RFC# |  | | | |
| TM# |  | | | |
| **Environment Verified** | | | | |
| RDP / SQL Permissions granted to PSITADM? | Yes | | No | |
| Service running under? | Local | | Domain | |
| Business justification if running under local account. |  | | | |
| Symmetric storage Checked? | Yes | | No | |
| **AOAG Details** | | | | |
| Configuration Type | **1** | **2** | **3** | **4** |
| Complete Instance Name(s) |  | | | |
| Availability Group Name |  | | | |
| Listener Name |  | | | |
| IP address(es) |  | | | |
| For each Replica choose one (async for multi-subnet) | Async / Sync | | | |
| Failover type (AUTO only if Sync & not a FCI) | Manual / Auto | | | |
| Sync & Failover Options | Async / Sync | | Manual / Auto | |
| Primary |  | |  | |
| Replica 1 |  | |  | |
| Replica 2 (if needed) |  | |  | |
| Replica 3 (if needed) |  | |  | |
| Replica 4 (if needed) |  | |  | |

**Possible Blockers**

Following could be possible blockers while implementing this change activity.

1. Nodes which are in request are not able to ping or TS.
2. Nodes are not yet part of the same Windows Failover Cluster.
3. WSFC is using a disk based quorum, otherwise node-majority with file share for when nodes span across multi-subnet.
4. Listener IP(s) / DNS pinging from local machine.
5. Listener IP(s) not from the correct subnet.
6. #Cluster Account does not have object creation right in the AD or the Resource Names have not been pre-staged.
7. More of a manual intervention required if
   1. \*Nodes do not have symmetric storage.
   2. \*SQL Services are running on managed accounts.

Contact requestor to have correct information for any of these listed blockers.

# Pre-Execution Steps for creating new AlwaysOn Availability Group

1. Verify the possible blockers.
2. Windows patches can be installed before or after this configuration.
3. SQL Server 2012 patch level should be SP1 CU2.
4. SLA for processing the request.

# Execution Steps

Steps to be followed for executing the RFC for this KCT.

### Dummy names & configuration settings used in this setup.

Following server hosts are running inside a pre-existing domain that spans across three subnets. The three subnets are identified as Sub[A/B/C], machines belonging to a specific subnet have it specified in the name. There are two SQL FCI configured (one each) inside SubA & SubB, the file-share witness is hosted at SubC.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **IP** | **Type** | **Comments** |
| w62subAdc | 192.168.1.1 | Domain Controllers | Domain Controllers under domain **Tree.com**. |
| w62subBdc | 192.168.2.1 |
| w62subCdc | 192.168.3.1 |
| w62subAss | 192.168.1.21 | Storage Servers | Storage Servers & file share witness hosts. |
| w62subBss | 192.168.2.21 |
| w62subCfsWit | 192.168.3.21 |
| w62subAdcNA | 192.168.1.101 | Clustered hosts inside a multi-subnet cluster | Nodes in Primary subnet [subA] to host clustered SQL instance SQAL. |
| w62subAdcNB | 192.168.1.102 |
| w62subBdcN1 | 192.168.2.101 | Nodes in Secondary subnet [subB] to host clustered SQL instance SQNU. |
| w62subBdcN2 | 192.168.2.102 |

WSFC & SQL Failover clustered instances in place.

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **IP** | **Comments** |
| Domain | Tree.Com | 192.168.1.1  192.168.2.1  192.168.3.1 | Domain spanning across three subnets & holding cluster nodes and storage servers. |
| WSFC | HADRclus | 192.168.1.31  192.168.2.31 | Windows Server Failover Cluster spanning across two subnets, holding two SQL FCI(s) & SQL AAG. |
| SQL FCI | SQAL | 192.168.1.32 | SQL Clustered Instance in Primary subnet |
| SQL FCI | SQNU | 192.168.2.32 | SQL Clustered Instance in Secondary subnet |

Domain based service accounts being used.

|  |  |
| --- | --- |
| **Service Accounts** | **Purpose** |
| Sqleng | Domain level service account to be used with SQL Server DB engine. |
| Sqlage | Domain level service account to be used with SQL Server Agent. |

CAP names & IP to use while configuring AO availability groups & listener.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **IP** | **Type** | **Comments** |
| HADRaag | na | Availability Group Name | The Availability Group name, it will also appear as the failover cluster resource group. |
| HADRlist | 192.168.1.33  192.168.2.33 | Availability Group Listener Name / CAP | Name & IP for clients to connect with the databases under the Availability Group. |
| AOseed | na | User Database | Database used to demonstrate AAG setup. |

### Preview of the Setup

### This solution provides HA via a local SQL Failover Cluster Instance and provides Disaster Recovery via an Availability Group to a second FCI in a secondary data center via Asynchronous synchronization.

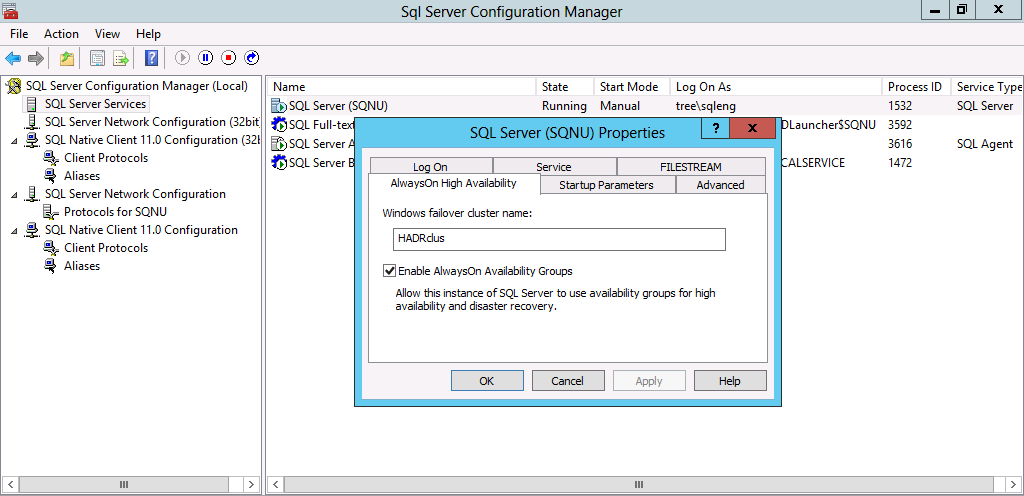
\* This configuration could be further scaled with additional nodes in each datacenter hosting additional FCIs with AGs connecting them.

### Preparation Steps

1. This is applicable when clustered SQL instances are involved. Each replica instance from an availability group must reside on a different node of the (same) WSFC. To avoid a failure later, cross check the resource ownership, these should not go beyond the designated nodes. Like for this setup no resources from SQAL should have nodes w62subBdcN1 or w62subBdcN2 as the possible owner.
2. Identify a shared location & give RW access to the service account running SQL Engine. This is generally the backup drive on the Primary server, i.e. E: drive. Initial AOseed database is a copy of model DB otherwise for bigger databases verify free space available. We can also use the setup option join only (after restoring the databases manually on the secondary instance(s)).
3. Create firewall exceptions on all the clustered nodes participating in the AAG group. These exceptions will be inbound & named as [**sqlAG**]. TCP Ports may vary depending on the setup.

|  |  |
| --- | --- |
| * + 1. **1433**     2. **5022** | Could be any other or additional ports depending how the SQL instance is configured & mirrored endpoints are registered. |

1. Enable AAG for each of the identified SQL instances using SQL Server Configuration Manager on each node (If SQL Server instances are clustered, this needs to be done on the Active Node Only. Offline & online the SQL resource to make the change effective). Right click on SQL Server Engine service and select Properties, then select the AlwaysOn High Availability tab & check against Enable AlwaysOn Availability Group.



In case of Windows Core, this can be done easily through PSH, required the SQLPS modules are imported successfully.

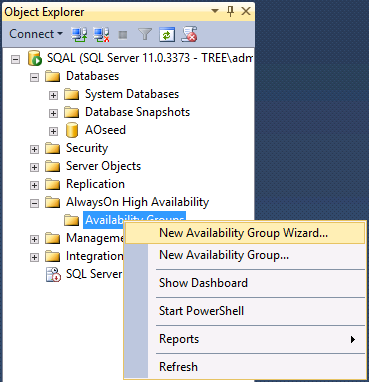


* If using GUI, the SQL Services need to be restarted manually otherwise PSH will restart them.
* On a FCI, this needs to be done on active node only.
* If the underlying WSFC is destroyed & recreated, the steps to enable AAG need to be redone i.e. this feature needs to be Unchecked, SQL service restarted, checked again and service restarted.

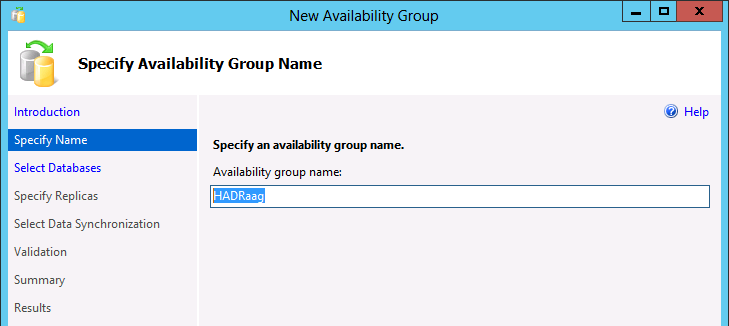
1. Connect to the SQL instance that will be the Primary replica & create a seed database named AOseed. The database should be owned by SA & must be in Full Recovery Model. The AAG wizard will try to restore the database MDF/NDF/LDF files on secondary instances at the same path as on primary. In case symmetric storage is not available the restore needs to be done manually followed by selecting option Connect Only in the wizard.
2. Take a full backup of the AOseed database to initialize the LSN chain.

### Availability Group Creation

* 1. Using **New Availability Group Wizard** to create an AOAG.
  + Using SSMS connect to the server/instance you want to make primary.
  + Right click on the Availability Groups & select New Availability Group Wizard

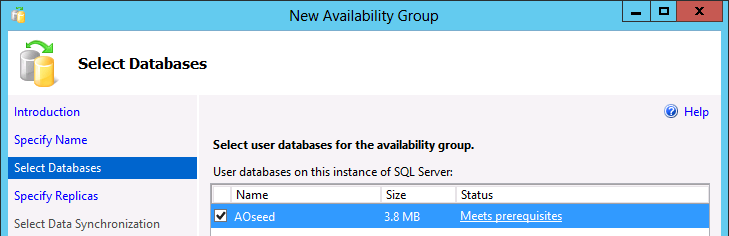


* 1. Provide an **Availability Group Name**
  + This will be the Cluster Resource group name (this is unique on the instance & cluster). It is not registered in DNS or AD.

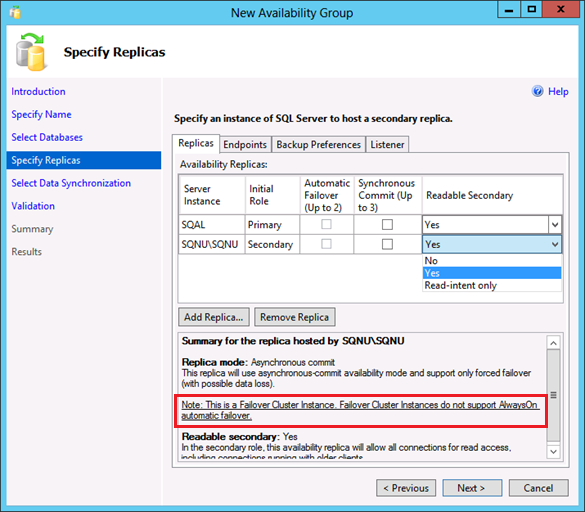


Hit Next to continue

* 1. Select **Participating Databases**
  + Select User Database(s) for Availability Group. Here it will be the AOseed database created earlier.
  + In case the database is not listed or doesn’t say meets pre-requisites check section **Preparation Steps,** make changes & refresh.

****

* 1. Add **Replica SQL Instances**
  + Click on **Add Replica** & provide the **server name** in connect to server dialogue.

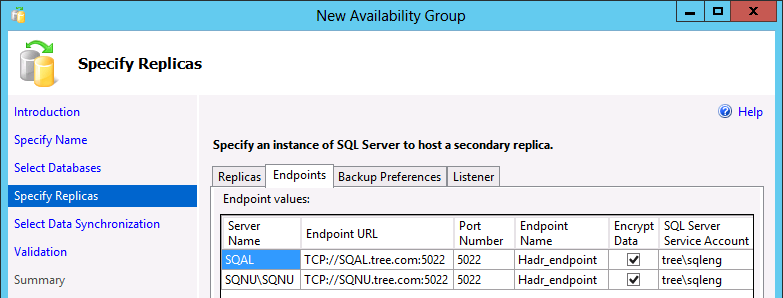


* + Once connected the screen looks something like above. Specify the replica mode & connection type for the instances that will be in secondary role.
  + Some key points to note here are;
    - Instance used to configure **AlwaysOn** will automatically be taken as **Primary**, replicas as added will be marked secondary.
    - Total of 5 sql server instances can participate in an AlwaysOn Availability Group though **only two will support Automatic Failover** (\*not supported with SQL FCI) & to provide for Automatic Failover the replicas should be doing Synchronous commit (\*should be avoided between multiple datacenters). At a time a **total of three out of five servers can be set to have Synchronous Commit**.
    - RO activities can be directed to the secondary replicas which will release RO load on the primary though causing a bit of management overhead on primary, the available options are
    - Yes - Allows all connections but only RO activities will be honored.
    - Read-intent only - Only connections specified as RO will be allowed.

Use ADO.NET 4.02, SQLNativeClient 11ODBC/OLE DB or Microsoft JDBC 4.0 driver (or above versions) for declaring Application Intent.

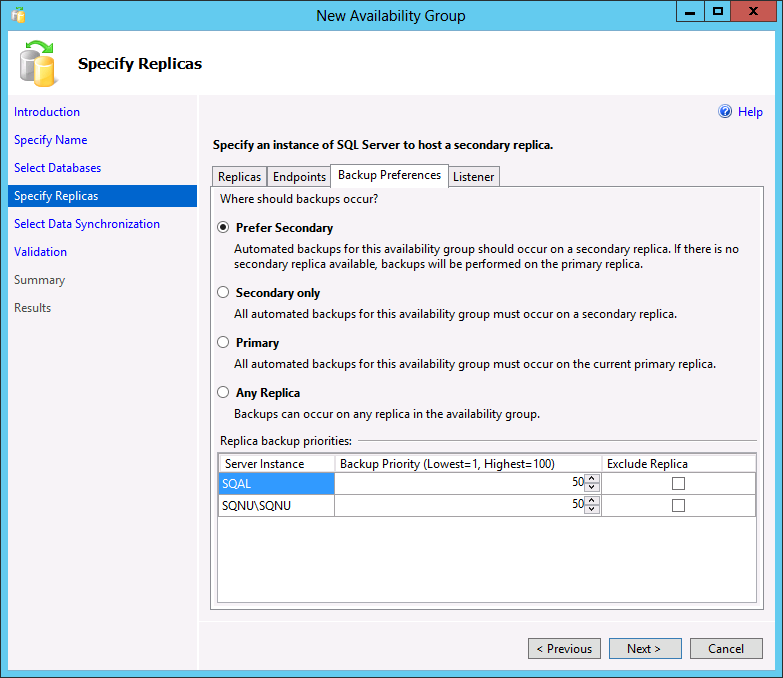
* + - Like RO activities, backups (FULL with copy only & TLOG), DBCC checks can also be directed to the secondary replicas.
  1. **Configure Endpoints & verify the parameters for each instance.**

This tab shows the default endpoint configuration & you‘ve the option to mark the communication as **encrypted** or otherwise, change the ports. It also shows the SQL Services account names that should have access to the replicas.



* + The Endpoints will be automatically created & shared amongst multiple AOAG on the same instance.
  + When setting up AOAG for multiple instances on the same machine, you will need to specify different port#.
  + Make sure those ports listed in the endpoint url against the server name, are inbound open in firewall on all nodes.
  1. **Backup** **Specifications**

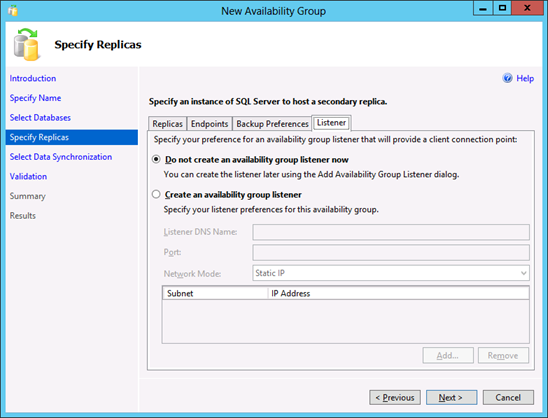
It’s suggested to have a **centralized backup** location. Backups operations can be directed to any of the participating servers or the selected ones can be excluded as well. Selection order is priority driven, where 100 is highest & 1 being lowest.



After completing the setup, to determine whether the current replica is the preferred backup replica use **sys.fn\_hadr\_backup\_is\_preferred\_replica** function it just checks if (top-priority replica is local).

* 1. **Listener Configuration**

The last tab is used to configure a Listener for this availability group (SSMS can add only one listener per AAG. In case multiple listeners are required use CLUadmin or PSH).

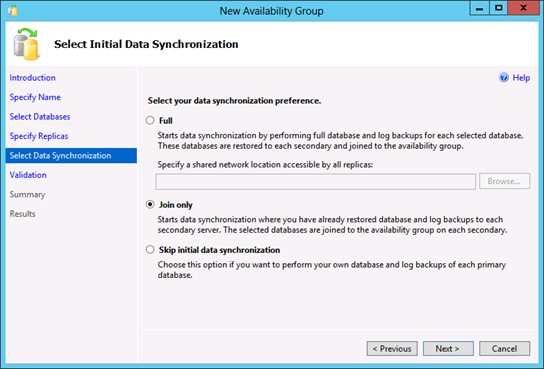


Creating a listener involves creation of new resources in cluster resource group, DNS registration & enabling account in the Active Directory, there can be complications under AD because of duplicate names, out of range IP addresses & permission restrictions. To make it easy to identify errors/warnings, we will create an AAG first & add the listener later.

* 1. **Initial Data Synchronization** **Preferences**

Hitting next brings us to this screen, where we can choose how the initial data synchronization will happen.

* + **FULL** – Initial Data Synchronization can be done through the wizard, which requires a shared folder with RW access for the SQL Service accounts. The wizard will do a full-copy only + log backup on this location & restore them on the replica, the database(s) will show as restoring unless they are synchronized.
  + **Join Only** – To be used in case there are multiple / big databases to be added and/or in presence of non-symetric storage. In such scenarios first backup & restore a FULL+LOG backup on the secondary with NORECOVERY (and move) option as applicable then select the option Join only.



**\*Avoid giving RW access to everyone, if feasible drop this folder after setup completes.**

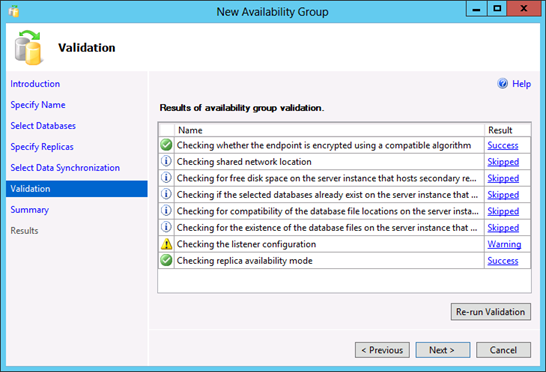
**\*Incase the shared location will stay for long & expected to be regularly used, it would be better suited on a NAS or UNC rather than one of the replicas.**

* 1. **Validations**

Following are checked on this validation screen.

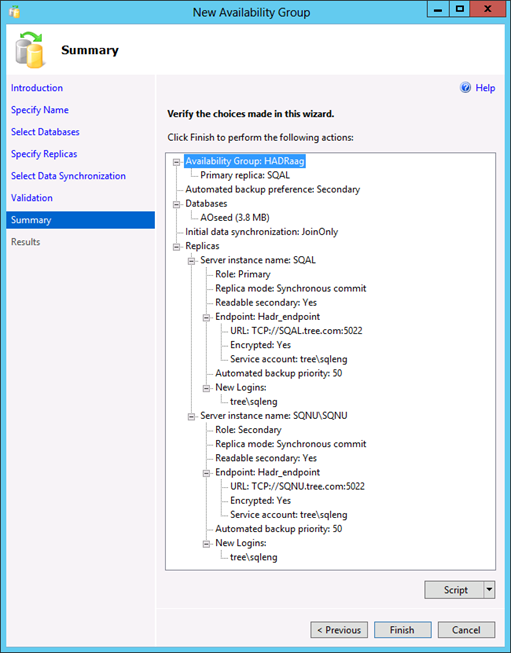
* + Encryption of endpoints at all the instances
  + Only in case the wizard will sync the databases, following are applicable
    - Verification of RW access to the shared network location
    - Checking of free disk space, database existance, symmetric storage, file name availability
  + In this case we are not creating a listener so it’s highlighted in yellow.
  + Availability mode of the replica

On selecting the option JOIN ONLY in the previous step, the points marked with ”**i**“ are skipped. The four checks before the Listener Configuration will be repeated for every replica added. Listener Configuration appears in yellow as listener creation is skipped.

****

* 1. **Summary Screen**

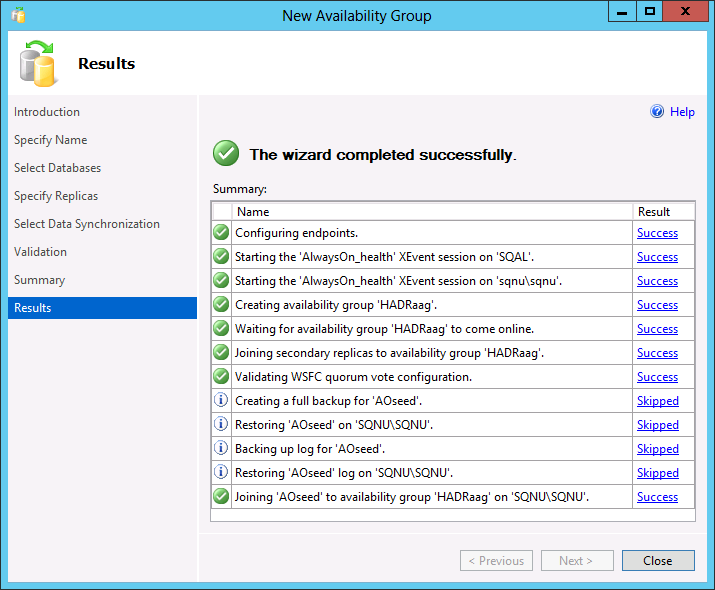
The screen below shows a review of the options selected, providing some idea about the expected actions that will be performed. These can be scripted for reference, & help if this fails. Once you’ve checked/scripted these, click next.



**\* It’s suggested to script out these commands as per feasibility, in case the process fails at any stage, one can use those scripts to start from the point onwards.**

* 1. **Progress through the setup/creation of AOAG**

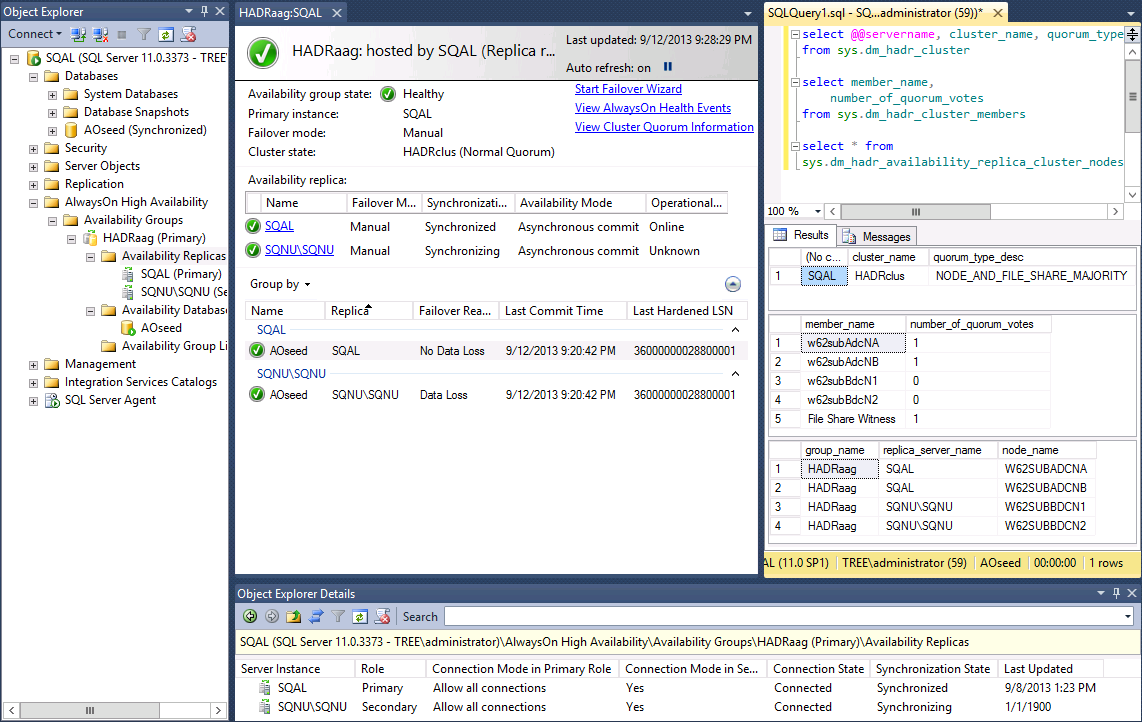
Progress on creation of AOAG is shown on this screen (click on the hyperlinks to the right for details).



After closing this the AlwaysOn Availability Group would look something like the screen on the next page. It also includes dashboard & queries to show additional information about the setup.

* 1. **AOAG listing in SSMS – Object Explorer**

Object explorer shows the Availability Group status as under, right click the AOAG name & open dashboard to see the synchronization status, on the primary replica this shows the status for all the participating instances, same action on a secondary will show status for that that instance only.



In the screen above the following parts show the specific information.

Connected to the instance using Instance Name (no listener available for now).

Dashboard : Replicas & DB sync status

Object Explorer : List of Replicas, databases, listeners (as applicable)

Object Explorer Details : AAG participating instances & their sync status along with allowed connections.

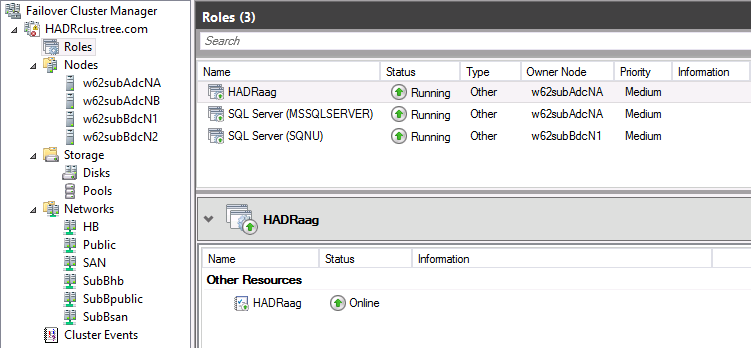
Query1 : Current host instance, cluster & quorum details

Query2 : Quorum votes

Query3 : SQL FCI participating nodes

* 1. **AlwaysOn Resources as shown under CluAdmin**

The status of the AO resource group can also be checked from Cluster Admin.



\*NOTE – we are yet to create a listener, so far data synchronization to the secondary has already started.

The above figure has the AOAG resource only, now we will add a listener using SSMS which will add a CAP & dependent IP addresses.

Next step is to add a listener, this can be done through either of SSMS / CLUADMIN / PSH. We will see how to use SSMS to add a Listener.

* 1. **Change AlwaysOn Availability Group session timeout.**

Default value is set to 10 but in case of multi subnet scenarios, high numbers of nodes in the cluster this should be relaxed. This is to avoid the possibility of a heavily loaded system missing PINGs and declaring a false failure

|  |  |  |
| --- | --- | --- |
| Availability Group Replica Property  |  |  | | --- | --- | | Session Timeout | 20 Seconds | |

**Run once for each replica.**

ALTER AVAILABILITY GROUP <**AGName**> MODIFY REPLICA ON <**ReplicaName**> WITH (SESSION\_TIMEOUT=20)

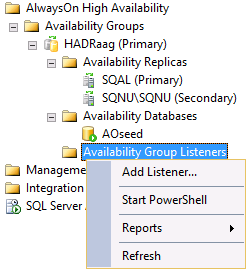
Use catalog view **sys.availability\_replicas**.**session\_timeout** to see the current set value.

### Adding a Listener to an AOAG

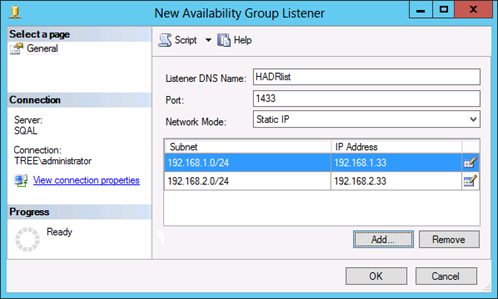
To create a listener we will need a unique name in AD, an available IP address and a port for communication (you may use the current SQL port#).

1. Select the **Availably Group** to **add a listener**

Expand the specific AOAG & right click on Availability Group Listeners, select Add Listener.



1. Provide values for **Availably Group Listener Name, IP Address & Port#**.

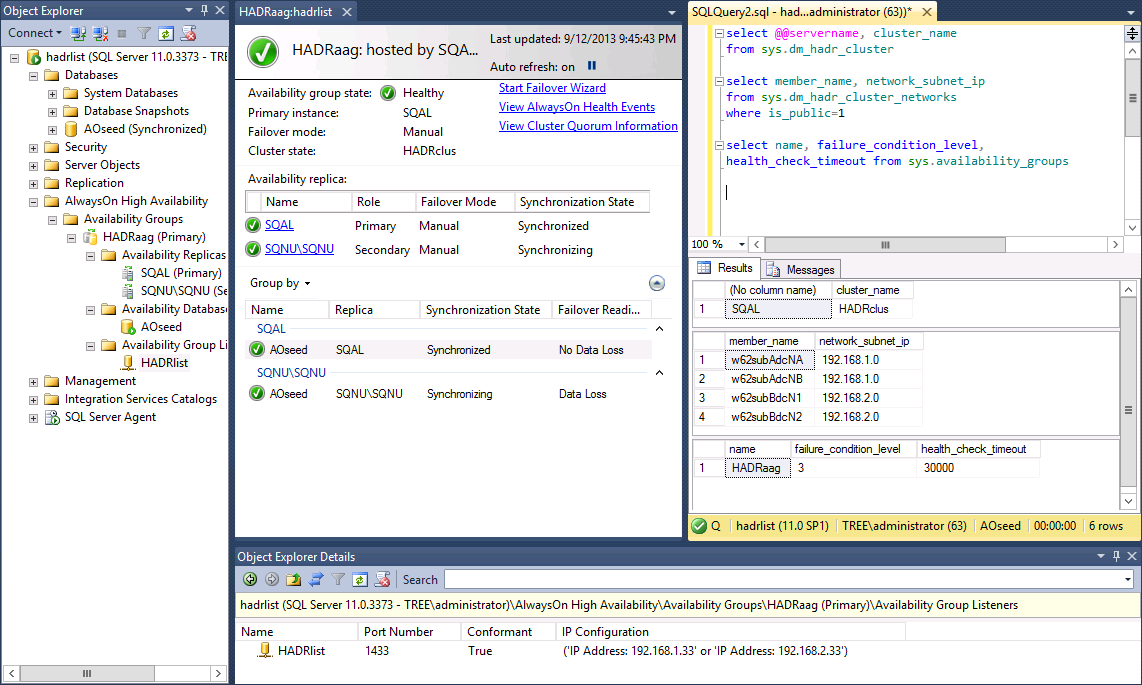
Listener configuration requires an IP address for each subnet & a port number to listen to (Listener name & IP are uniquely registered in AD. This behavior can be altered by setting RegisterAllProviderIP), though DHCP can be used but in production cases it’s suggested to have a Static IP. If port numbers other than 1433 are used they have to be mentioned in the connection string.

* + Don’t use names longer than 15 chars (LIMITED BY NETBIOS).
  + Listener ports are not detected by the Browser Service
  + As clusters are multi-subnet, it’s required to specify the IP address from the correct range.
  + Unless required an AAG will only host one listener & that can use the same port as being used by SQL Server. To add multiple listeners use CluAdmin or PSH.

Being a multi-subnet AAG it requires two IP addresses, one to be live in each subnet as the Availability Group moves over.

Except for the port #, no other settings for the listener can be changed later from SSMS. To rename the listener or change the IP address this has to be dropped & re-created, otherwise use PowerShell or ClusterAdmin (these may have implications +/- ve depending upon your environment).

On clicking OK the listener will be created & hosted under the AOAG like this.



In the screen above the following parts show the specific information.

Connected to the instance using Instance Name (no listener available for now).

Dashboard : Replicas & DB sync status

Object Explorer : All databases, AAG replicas & listener name & status.

Object Explorer Details : Listener IP (s) & Port # details.

Query1 : Current host instance, WSFC cluster name

Query2 : Subnets from all the nodes

Query3 : AAG group & failure condition level

\*Any operations that affect the Active Directory objects should be planned keeping in mind the time it takes for replication, (maybe) stale entries have to be manually cleared from AD.

The following views can be helpful to get listener properties.

**availability\_group\_listener\_ip\_addresses**

Returns a row for every conformant virtual IP address that is currently online for the listener.

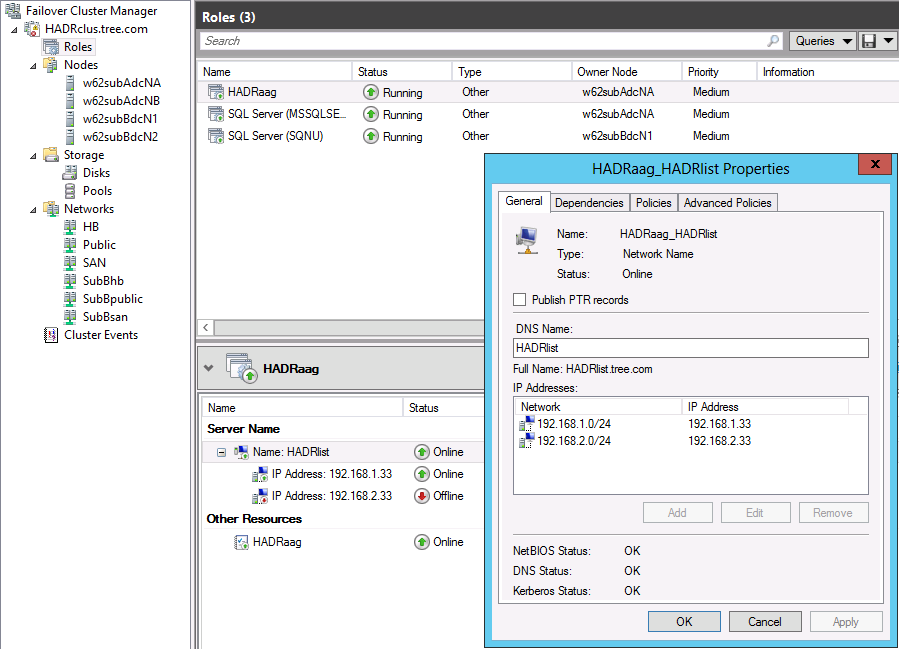
**availability\_group\_listeners**

For a the availability group, returns either zero rows indicating that no network name is associated with the availability group, or returns a row for each availability-group listener configuration.

**dm\_tcp\_listener\_states**

Returns a row containing dynamic-state information for each TCP listener.

This is how it’s shown under the cluster admin, the resources are online after creation.



|  |  |
| --- | --- |
| Listener Name is dependent on either of the two IP addresses.  If or not both the IP’s are registered in the DNS, is guided by the parameter RegisterAllProvidersIP. |  |

1. **Configure Listener Parameters**

There are a few parameters that need to be modified for a multi-subnet setup (An option to avoid this to add patch # 2654347 which will provide support to SQL Client for AlwaysOn features of Read-intent, read-only & multi-subnet failover. The hotfix needs to be installed on each client).

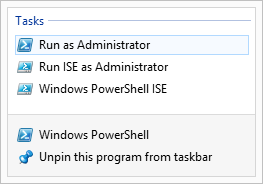
It’s required to bounce (offline/online) the cluster resource for changes to take effect.

**\*Win 2012 issues a warning to bounce the resources but there are no such reminders in older versions.**

As these parameters are Cluster Name’s Private Properties, they are not exposed through GUI. In the following steps PowerShell will be used to alter two such properties.

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Default Value** | **New Value** |
| HostRecordTTL | 1200 | 300 |
| Time in seconds before the IP / Name association information will be re-circulated over the network. | | |
| RegisterAllProvidersIP | 1 | 0 |
| Determine which IP addresses should be registered in DNS for your CAP (listener): either all IP addresses on which the network name resource depends (value: 1), or only the IP address that successfully comes online (that is, the IP address on the subnet of the node that currently owns that network name resource) (value: 0).  If set to 1 all the IP addresses associated with the Listener name will be registered, this may cause problems if the keyword **multi-subnet failover = true** is not used. For multi-subnet clustering, see [SQL Server Multi-Subnet Clustering (SQL Server)](http://msdn.microsoft.com/en-us/library/ff878716.aspx). | | |
| **Table 1** | | |

To make changes to these settings use PSH with administrator privileges.

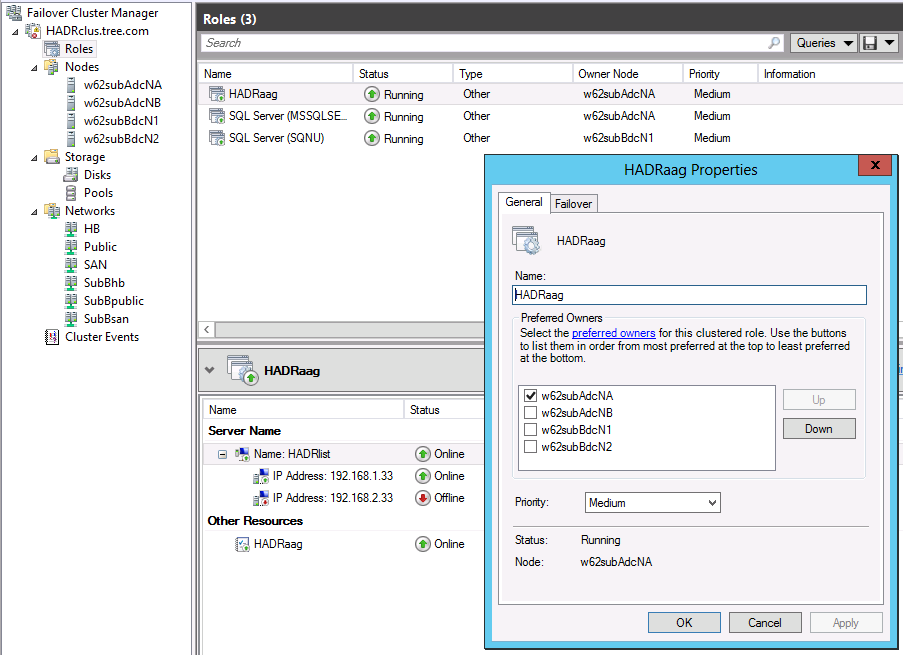
Some systems (mostly win 2003) may require to explicitly call in support for failover cluster commands by running **Import**-**Module FailoverClusters**.

The following commands illustrate changing the parameters listed in **Table 1**. When running these commands from PSH make sure to verify the resource names as these may not be the names seen in DNS, SSMS or CluAdmin. When created from SSMS the name will be “**AGName\_ListenerName**” & when created from CluAdmin it will be “**ListenerName**”

|  |
| --- |
| List all Resource Groups. |
|  |
| List all Resources from the AlwaysOn Availability Resource Group. |
|  |
| As per current standards in MSIT, for multi-subnet AOAG running for clients, not having the updates to support MultiSubnetFailover=True, we will change the highlighted properties. |
|  |
| Change the specified parameters |
|  |
| Resource needs to be bounced after changing these parameters |
| |  |  | | --- | --- | |  | The AlwaysOn resource is dependent on CAP (Listener), check the status of all the depended / resources in the next step. In most of the cases, the AlwaysOn resource needs to be brought online. | |
| Verify the status of all resources in the AlwaysOn Resource Group |
|  |

1. **Configure AOAG resource group properties for failover & fall back.**

In the figure below you can see the participating nodes of AOAG are checked (never modify OWNER settings from CluAdmin, typically in the cases when there’s a FCI instance involved. The resource will change owners as FCI bounces between nodes).



Because the failover method selected here is manual, the other node is not checked as preferred owner.

On the second tab

* + Increase the count of **Maximum failures to 20**, by default this number is node count minus 1 (N-1).
  + Select the option to **prevent failback**.

|  |  |
| --- | --- |
|  | Default value of Maximum Failures permitted in the window is **Node Count minus 1 (**N-1**)** |

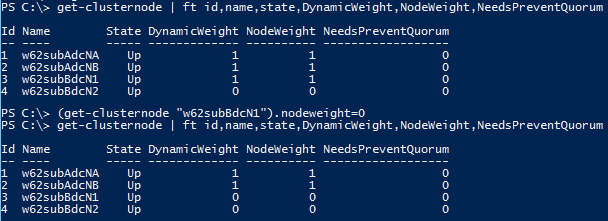
These changes don’t require to bounce the resource.

**At this point the AG has been built and is ready for QC and handoff.**

### Quorum Voting in DR Setup

NodeWeight settings are used to influence voting for the quorum, this may need a modification during a DR to support disaster recovery & multi-subnet scenarios for AlwaysOn Availability Groups. Since Win 2008, machines participating in a cluster can opt to vote or not to. Before Win 2012, patch 2494036 was required to enable a node to opt out of voting.

This is how PSH can be used to check voters & change it.



Similar TSQL to see the voting info would be

|  |
| --- |
| SELECT member\_name, member\_state\_desc, number\_of\_quorum\_votes  FROM sys.dm\_hadr\_cluster\_members; |

Except ‘Disk Only’ all the other quorum models rely on communications between all of the voting nodes in the cluster. Network communications between nodes on the same physical subnet should be considered reliable; the quorum vote should be trusted but nodes on another subnet may get non-responsive at times. Therefore it’s recommended to turn them off.



When validating WSFC quorum vote configuration, the AlwaysOn Availability Group Wizard shows a warning if any of the following conditions are true:

* + Cluster node that hosts the primary replica does not have a vote
  + Secondary replica is configured for automatic failover & that node does not has a vote.
  + [KB 2494036](http://support.microsoft.com/kb/2494036) is not installed on all cluster nodes that host availability replicas. In single-site deployments it can be ignored.

If the host are running an RTM build, following will also cause a warning.

* + Configured a replica to asynchronous availability mode
  + Configured a replica for manual failover mode

# Escalation Matrix

Following Escalation Metrics should follow:

1. Check in PRIMUS with error code.
2. Any issue or challenge reported need to get escalated to service owners.

# Appendix

# Document Administration Page

**Document History**

|  |  |  |
| --- | --- | --- |
| Date | Updated by | Change |
| Mar 20, 2013 | Steve Harrell | Draft of Document |
| Sep 12, 2013 | Yusuf Anis | Update & Internal Review |
| Sep 13, 2013 | Steve Harrell | Internal Review |
| July 7, 2014 | Yusuf Anis | Added Session\_TimeOut recommendations. |