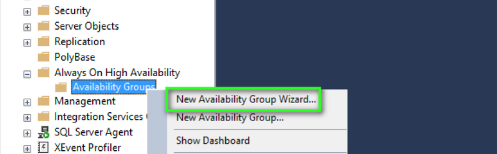
# HIGH AVAILABILITY SOLUTION – ALWAYS ON Part-1

***Objective:*** This document outlines the high availability solution known as "Always On," defines the associated terminology, and provides the necessary steps to configure the Always On feature etc.

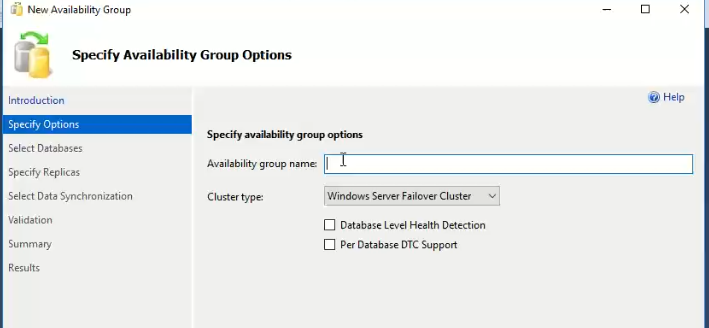
***Configuration:***

1. In order to configure the “Always on”, All the servers must be part of windows clustering mode.
2. Once clustering of servers configured then go to SQL Server configuration manager and turn on the “Always on high availability” feature.
3. Open SSMS and expand “Always on High Availability” and create an availability group.

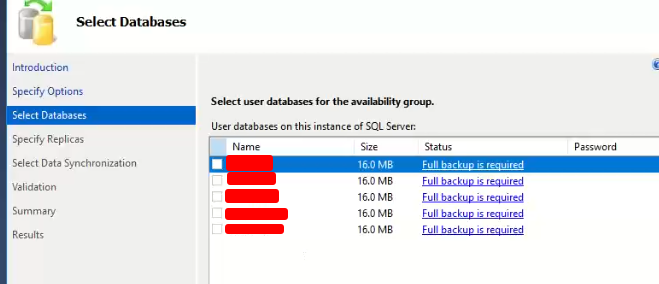


1. After clicking on “New availability group wizard” option you will see below interfaces-

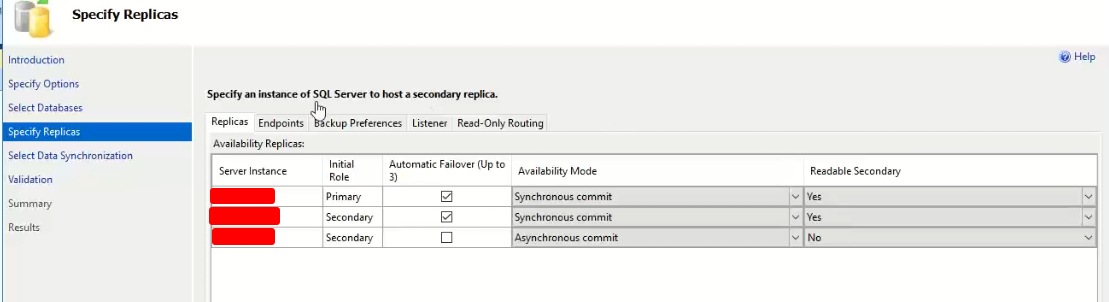
* Give the availability group name-



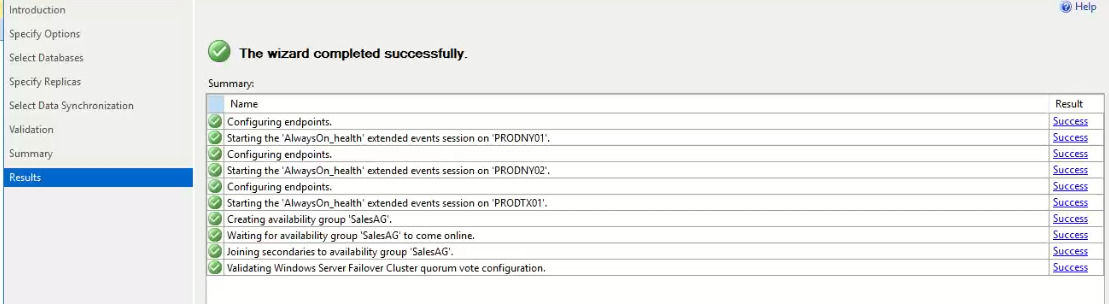
* Then select the databases to which you want to include in availability group [make sure before this step you must need to take full backups of all databases]



* Now you need to make a proper configuration setting for replicas-



* Click on next to proceed

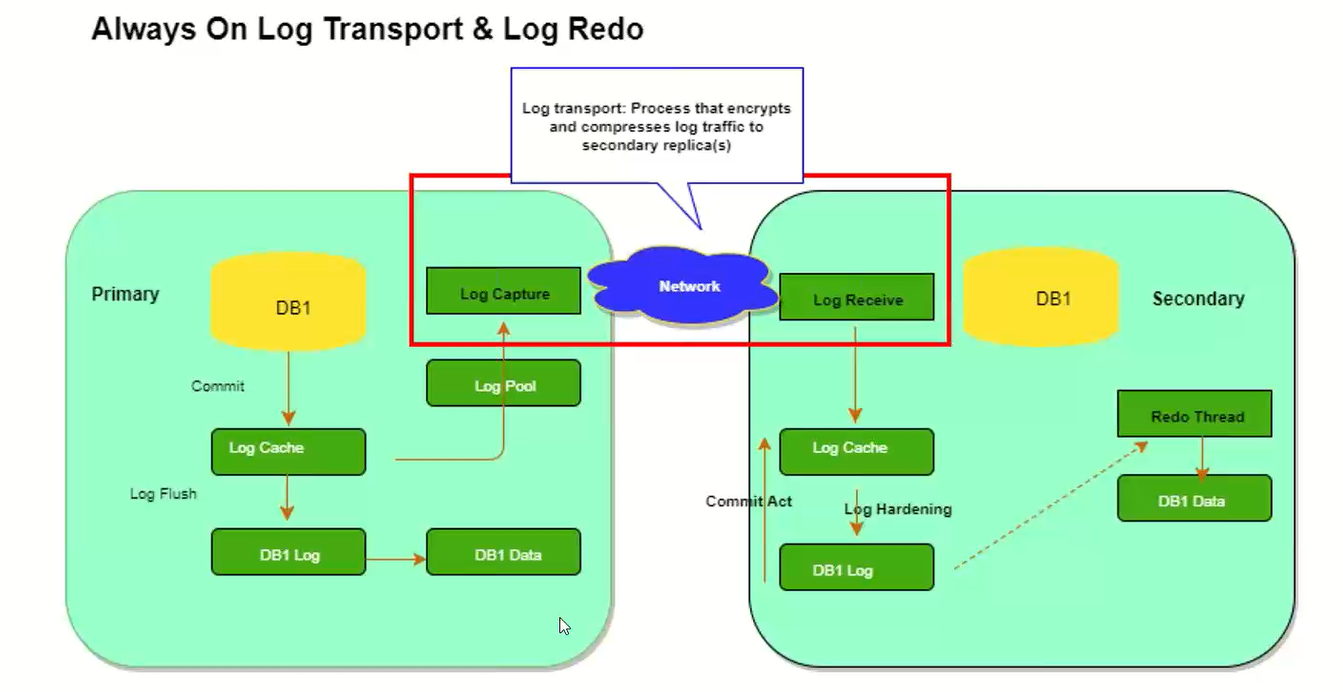


* Now you can see availability group created on object exploral.

1. In the same way you can create a listener for your instances so that it acts as clustering of sql servers.

Note: You must have SQL Server enterprise edition or evaluation edition to have always on feature.

***Working:***



1. When a transaction happens in the primary server, it will not commit immediately rather the transaction log saved in a log pool of memory.
2. From log pool transaction log transfer to the synchronized secondary replica and replay here.
3. Secondary replica sends the acknowledgment to the primary about replay of transaction log then the actual transaction was committed.

***Key Takeaways:***

1. We have maximum 8 replicas (1 primary and 7 secondary) and out of 7 secondaries we can have maximum 3 synchronous replicas.
2. Synchronous replica or synchronous commit means before primary replica commits the transaction all logs replay in secondary replicas and wait for the confirmation from secondary, once after the confirmation only transaction will commit.
3. One instance can have multiple Availability groups. If 1 database of node1 fails, system will automatically fail over to node2 or node3 (shown in the image).