DEPLOYING IBM® INFOSPHERE DATA REPLICATION® WITH A MICROSOFT® SQL SERVER® DATABASE PARTICIPATING IN AN ALWAYSON AVAILABILITY GROUP

Table of Contents

1 Deploying IIDR in Microsoft SQL Server AlwaysOn availability group

The following document describes the steps required to be taken in order to deploy IIDR with a database that is participating in a Microsoft SQL Server Availability Group. These steps are required when the IIDR will serve as a source for data replication.

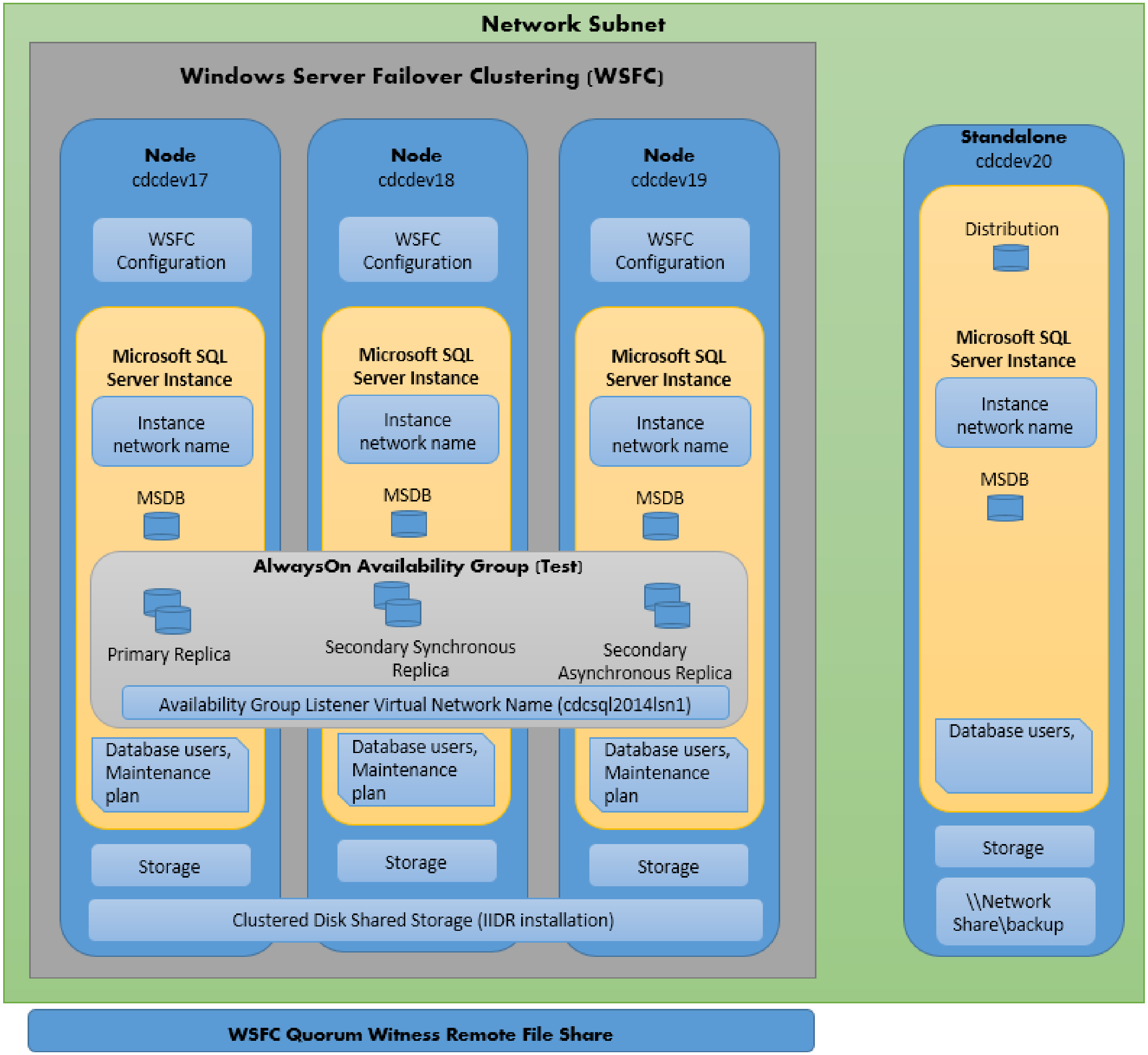
2 Overview

Before proceeding, ensure that all prerequisites for installing IIDR are met. For more information see:

https://www.ibm.com/support/knowledgecenter/en/SSTRGZ_11.4.0/com.ibm.cdcdoc.cdcformssql.doc/concepts/before_install.html

In order to be able to deploy IIDR in a Microsoft SQL Server AlwaysOn environment one needs to:

- Prepare the AlwaysOn availability group environment for replication
- Add symmetrical shared storage for IIDR deployment reachable from all AlwaysOn nodes
- Define a Microsoft Windows domain user that can be used to run the IIDR service
- Identify shared storage to which the transaction logs can be backed up. The storage must be symmetrically accessible from all AlwaysOn nodes
- Define maintenance plans to back up the transaction logs to the backup storage on any number of nodes
- Create the IIDR user that can connect to the database on all nodes where it may fail over
- Deploy IIDR on the shared symmetrical storage
- Create the IIDR service on all nodes to which IIDR may fail during planned or unplanned failover. Any number of nodes that are configured for Synchronous Commit availability mode could be used
- Add the IIDR service as a dependent resource to the AlwaysOn availability group role that was created in the Windows Server Failover Cluster
- Configure the new Datastore in IBM Management Console


Network Subnet
Windows Server Failover Clustering (WSFC)
Node
cdcdev17
WSFC Configuration
Microsoft SQL Server Instance
Instance network name
MSDB
Database users, Maintenance plan
Storage
Node
cdcdev18
WSFC Configuration
Microsoft SQL Server Instance
Instance network name
MSDB
Database users, Maintenance plan
Storage
Node
cdcdev19
WSFC Configuration
Microsoft SQL Server Instance
Instance network name
MSDB
Database users, Maintenance plan
Storage
AlwaysOn Availability Group (Test)
Primary Replica
Secondary Synchronous Replica
Secondary Asynchronous Replica
Availability Group Listener Virtual Network Name (cdcsql2014lsn1)
Clustered Disk Shared Storage (IIDR installation)
Standalone
cdcdev20
Distribution
Microsoft SQL Server Instance
Instance network name
MSDB
Database users,
Storage
\\Network Share\backup
WSFC Quorum Witness Remote File Share

3 Prepare the AlwaysOn availability group environment for replication

The following information is provided for guidance only. Please consult Microsoft documentation for exact details on the steps that need to be performed in order to enable and maintain replication for a database participating in an AlwaysOn availability group.

Adapted from:

http://blogs.msdn.com/b/alwaysonpro/archive/2014/01/30/setting-up-replication-on-a-database-that-is-part-of-an-alwayson-availability-group.aspx

More information at:

https://msdn.microsoft.com/en-CA/library/hh710046.aspx

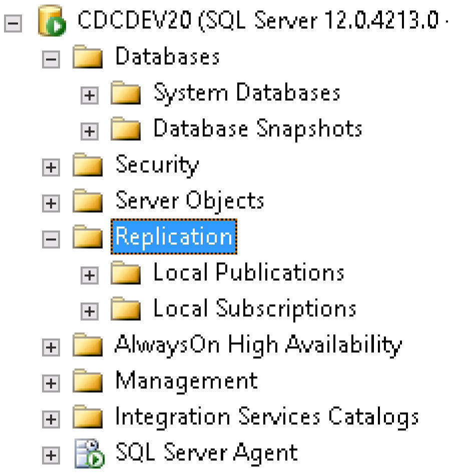
Review Microsoft documentation on AlwaysOn or Windows Server Failover Cluster terms and definitions:

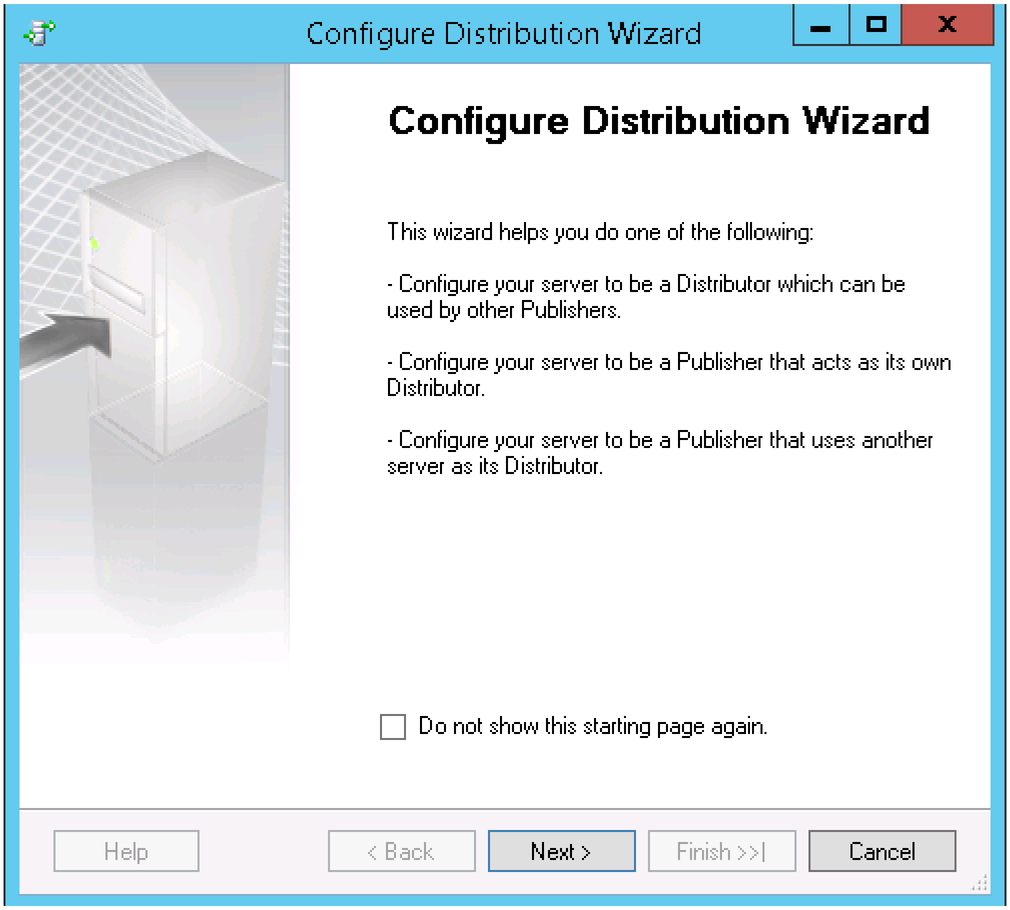
https://msdn.microsoft.com/en-CA/library/hh270278.aspx

The first step in preparing the environment for replication is to configure a standalone server that will serve as a remote distributor for the AlwaysOn availability group replica. In our example, connect to cdcdev20 and start configuring it to be able to serve as a remote distributor.

3.1 Configure the Remote Distributor.

Right Click Replication and select Enable Distribution.





Configure Distribution Wizard

This wizard helps you do one of the following:

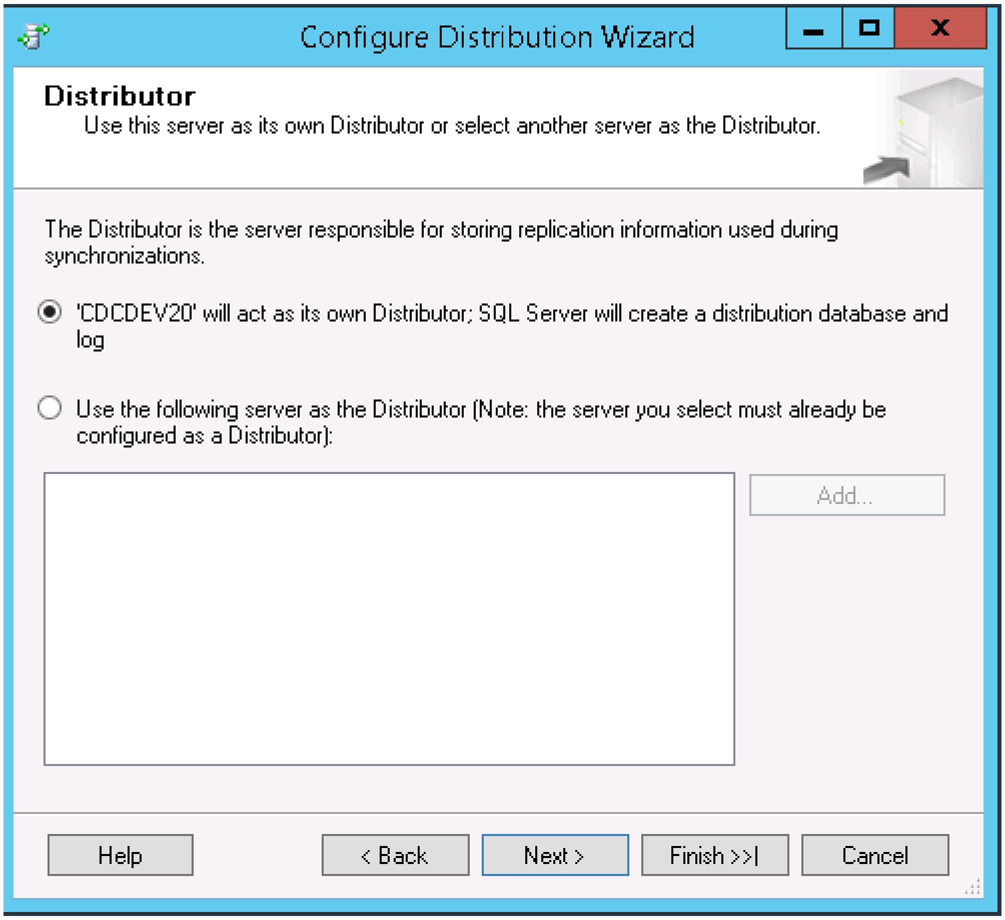
- Configure your server to be a Distributor which can be used by other Publishers.

- Configure your server to be a Publisher that acts as its own Distributor.

- Configure your server to be a Publisher that uses another server as its Distributor.

□ Do not show this starting page again.

We’ll select the first option as we want cdcdev20 as a distributor



Distributor

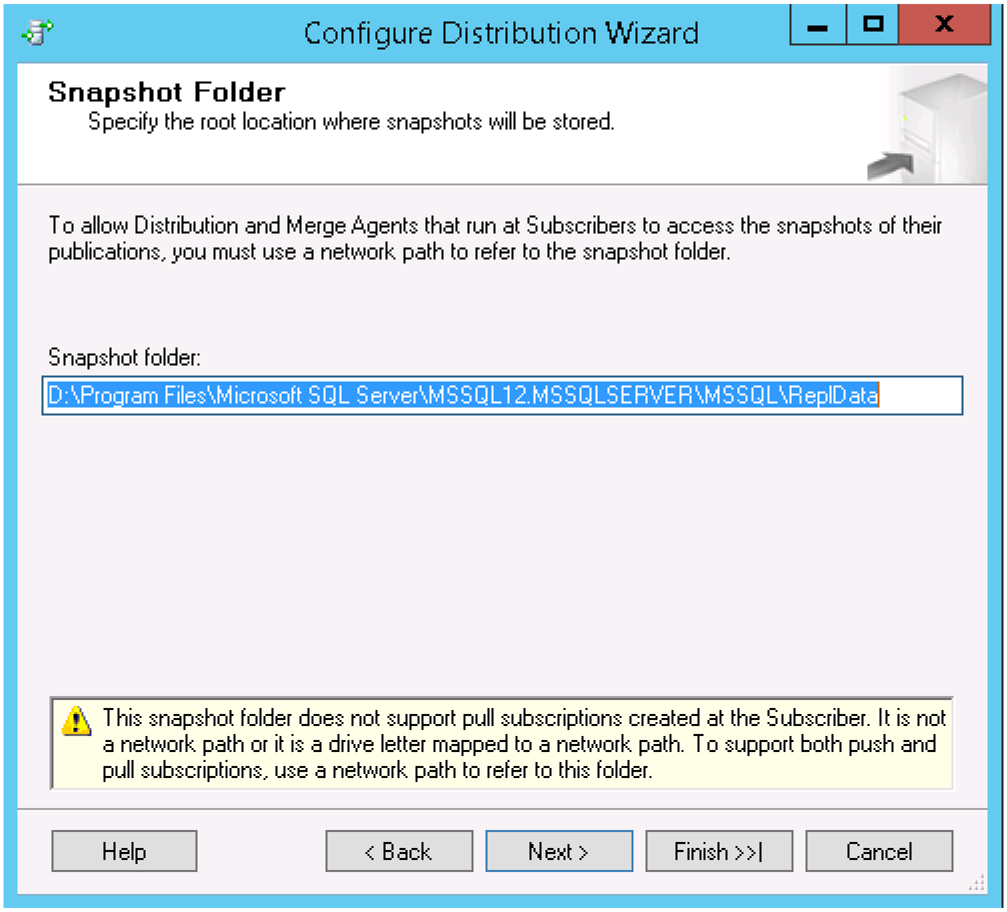
Use this server as its own Distributor or select another server as the Distributor.

The Distributor is the server responsible for storing replication information used during synchronizations.

● 'CDCDEV20' will act as its own Distributor; SQL Server will create a distribution database and log

○ Use the following server as the Distributor (Note: the server you select must already be configured as a Distributor):

Follow with the default snapshot folder:



Snapshot Folder

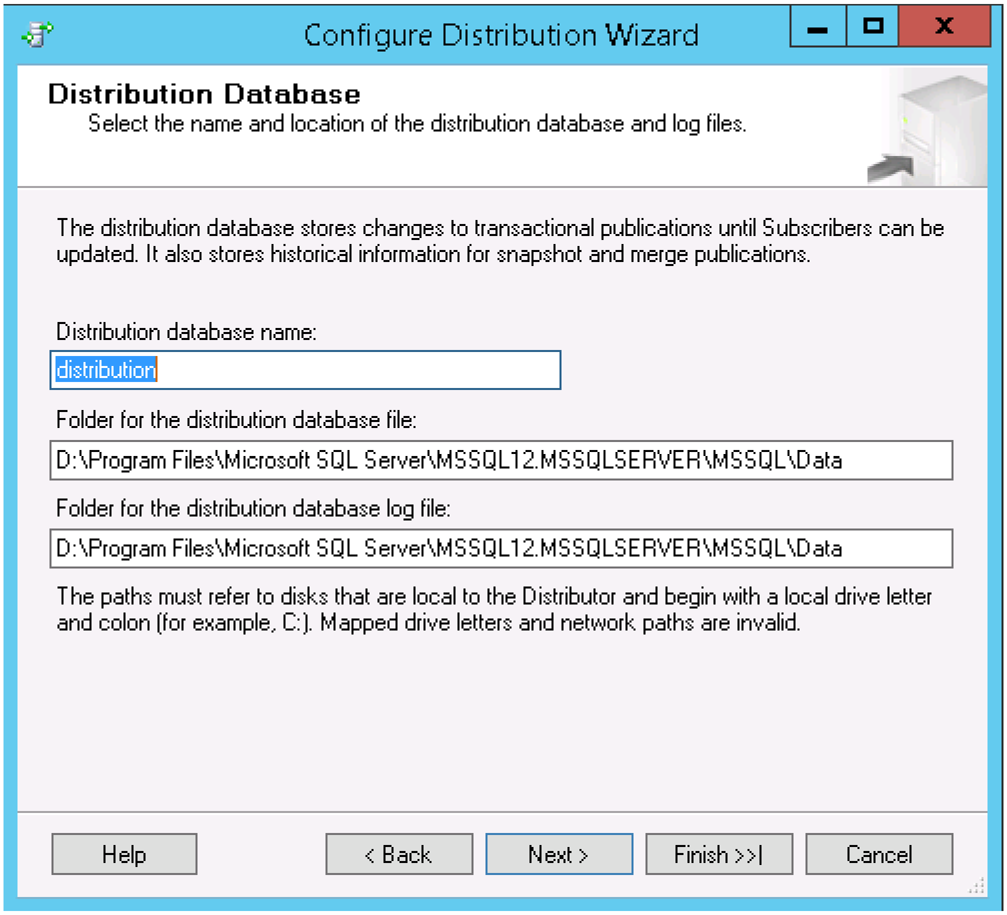
Specify the root location where snapshots will be stored.

To allow Distribution and Merge Agents that run at Subscribers to access the snapshots of their publications, you must use a network path to refer to the snapshot folder.

Snapshot folder: D:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\ReplData

This snapshot folder does not support pull subscriptions created at the Subscriber. It is not a network path or it is a drive letter mapped to a network path. To support both push and pull subscriptions, use a network path to refer to this folder.

Use the defaults for distribution database:



Distribution Database

Select the name and location of the distribution database and log files.

The distribution database stores changes to transactional publications until Subscribers can be updated. It also stores historical information for snapshot and merge publications.

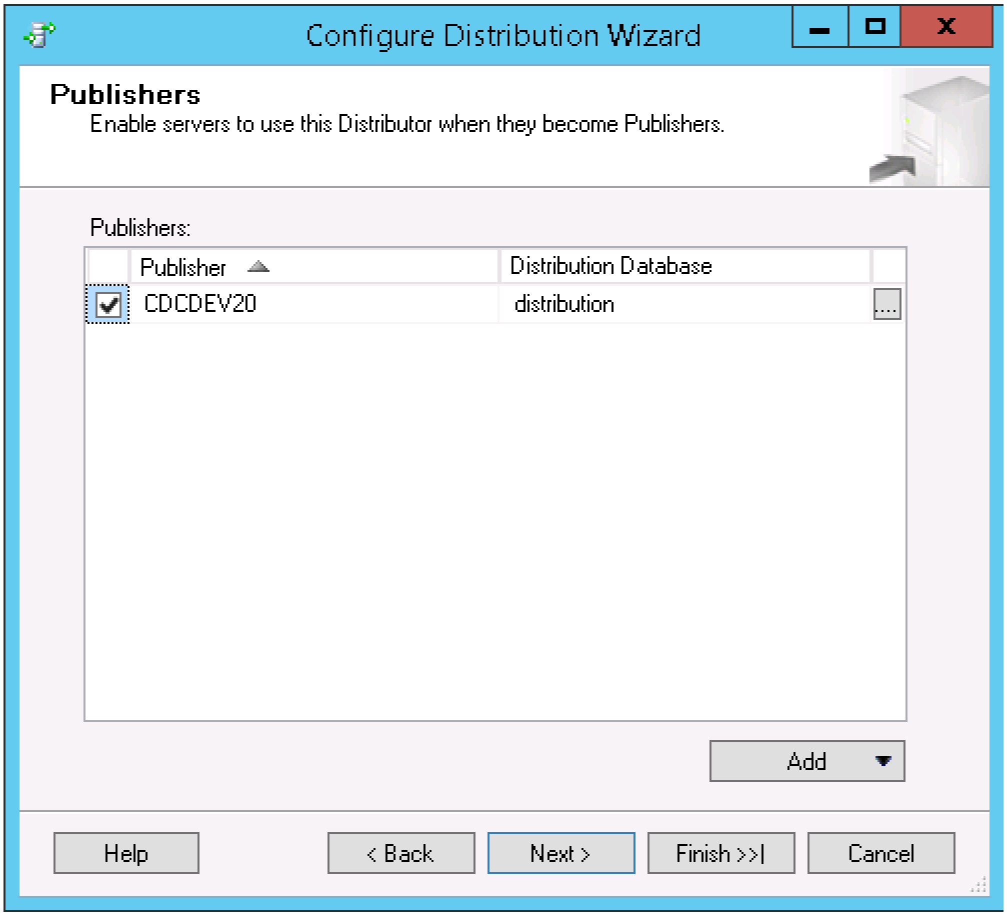
Distribution database name: distribution

Folder for the distribution database file: D:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\Data

Folder for the distribution database log file: D:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\Data

The paths must refer to disks that are local to the Distributor and begin with a local drive letter and colon (for example, C:). Mapped drive letters and network paths are invalid.

In the screen below we need to add the cdcdev17, cdcdev18 and cdcdev19 as possible publishers. Authenticate to all servers while adding the publishers.

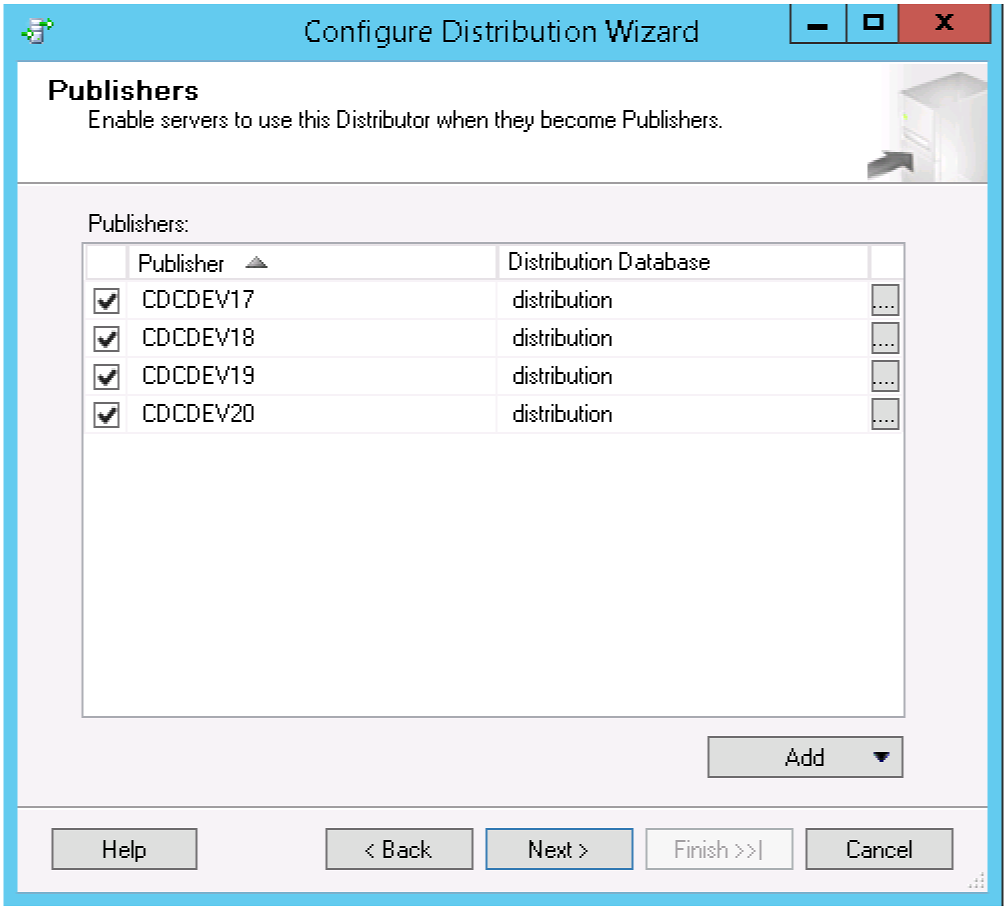


Publishers

Enable servers to use this Distributor when they become Publishers.

Publishers:

	Publisher	Distribution Database	
☑	CDCDEV20	distribution	...



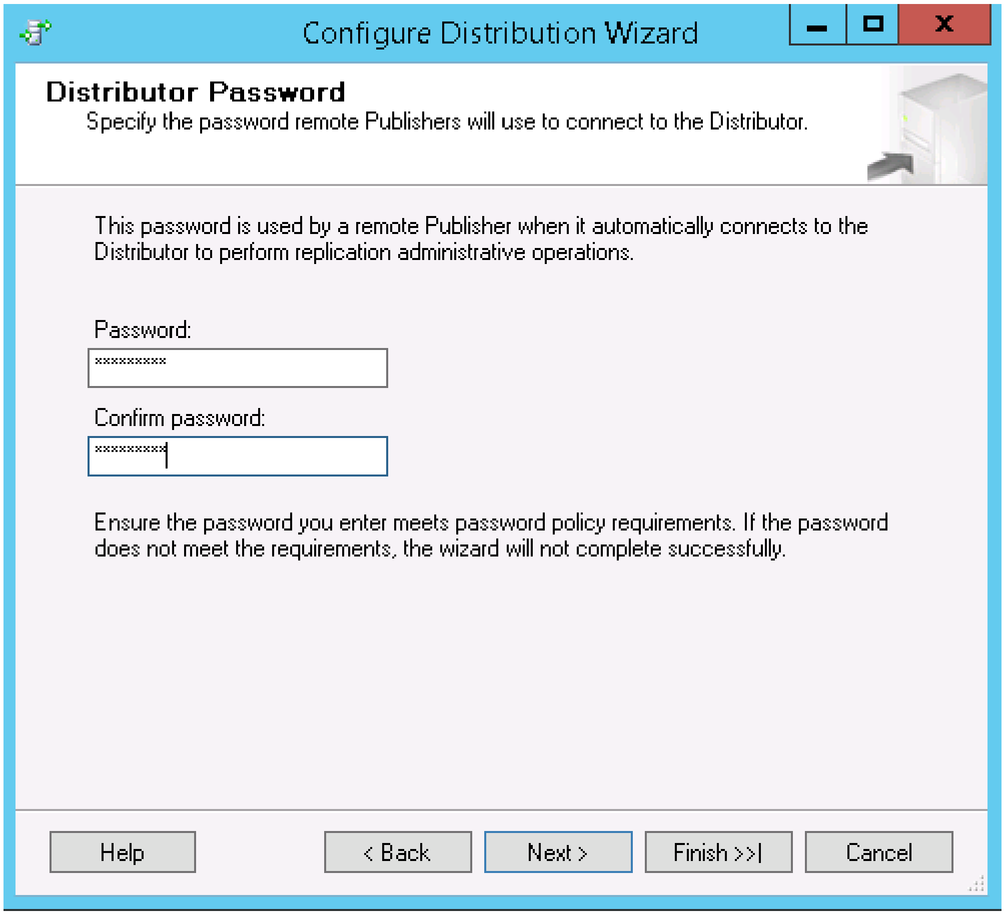
Publishers

Enable servers to use this Distributor when they become Publishers.

Publishers:

	Publisher	Distribution Database
☑	CDCDEV17	distribution
☑	CDCDEV18	distribution
☑	CDCDEV19	distribution
☑	CDCDEV20	distribution

Enter the password that the publishers will use when connecting to the remote distributor.



Distributor Password

Specify the password remote Publishers will use to connect to the Distributor.

This password is used by a remote Publisher when it automatically connects to the Distributor to perform replication administrative operations.

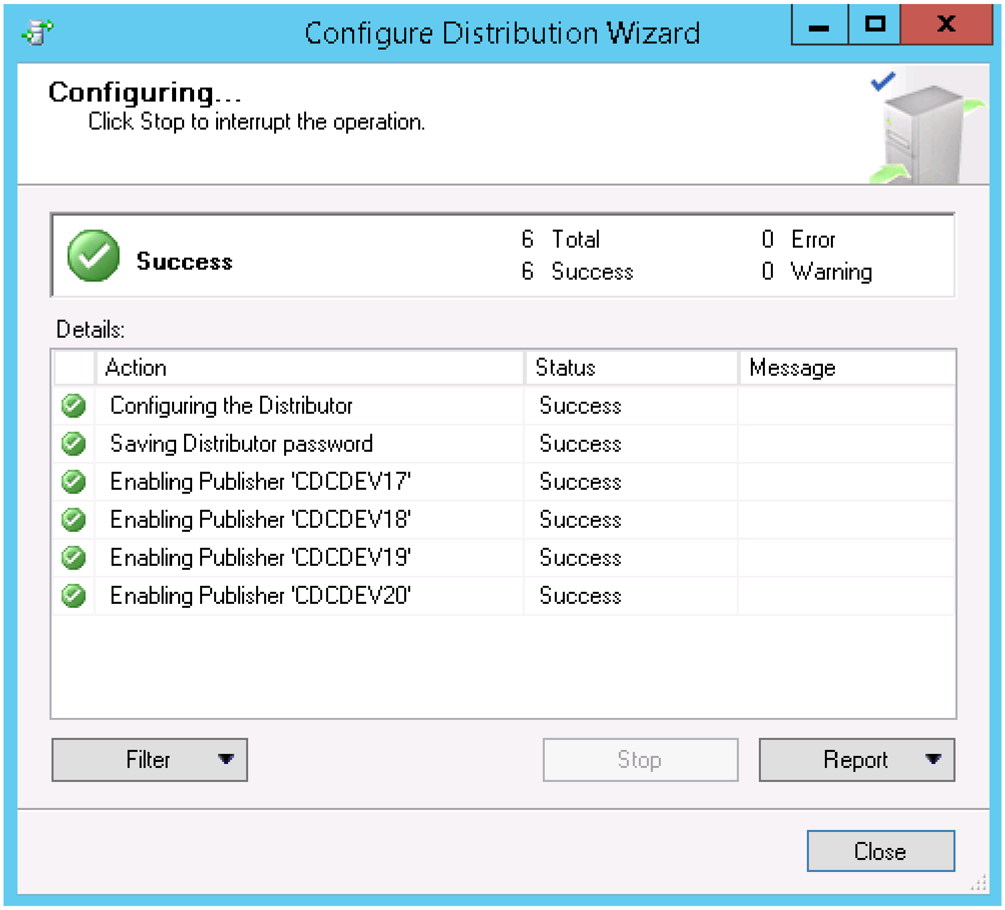
Password: **********

Confirm password: **********

Ensure the password you enter meets password policy requirements. If the password does not meet the requirements, the wizard will not complete successfully.

This information will be needed when configuring the publishers.

Click Next, make sure that “Configure Distribution” is checked then Next, then Finish.



Configuring...
Click Stop to interrupt the operation.

Success

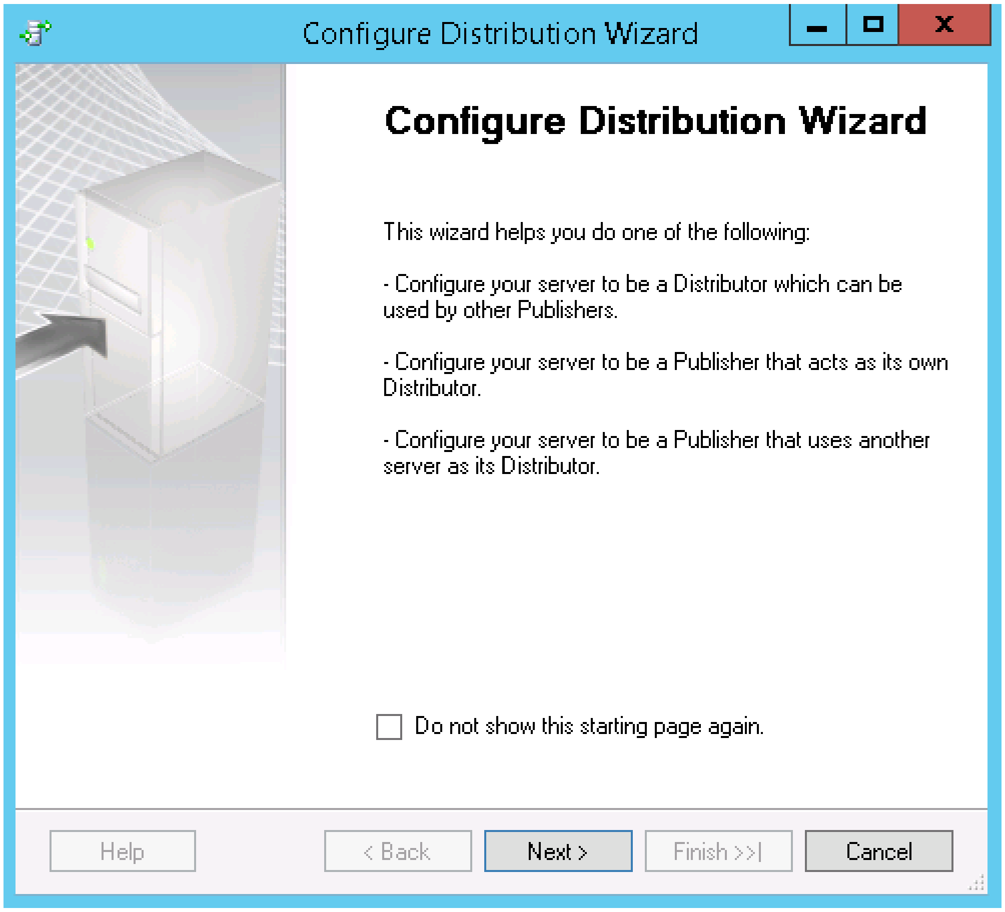
6 Total	0 Error
6 Success	0 Warning

Details:

Action	Status	Message
✓ Configuring the Distributor	Success	
✓ Saving Distributor password	Success	
✓ Enabling Publisher 'CDCDEV17'	Success	
✓ Enabling Publisher 'CDCDEV18'	Success	
✓ Enabling Publisher 'CDCDEV19'	Success	
✓ Enabling Publisher 'CDCDEV20'	Success	

3.2 Configure the remote distributors on the possible publishers

For the possible publishers, we need to configure distribution as a remote distribution that we created on cdcdev20. Right click Replication and choose Configure Distribution…



Configure Distribution Wizard

This wizard helps you do one of the following:

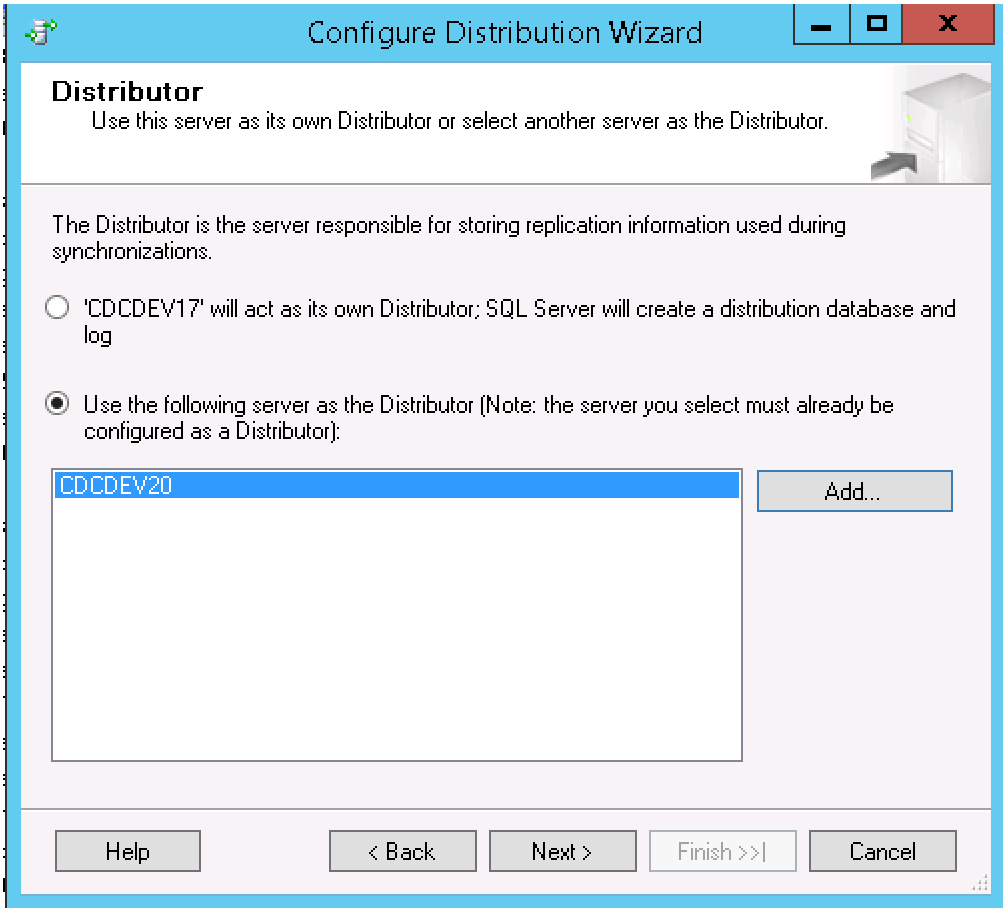
- Configure your server to be a Distributor which can be used by other Publishers.

- Configure your server to be a Publisher that acts as its own Distributor.

- Configure your server to be a Publisher that uses another server as its Distributor.

□ Do not show this starting page again.

Use cdcdev20 as a distributor for this Microsoft SQL Server instance.



Configure Distribution Wizard

Distributor

Use this server as its own Distributor or select another server as the Distributor.

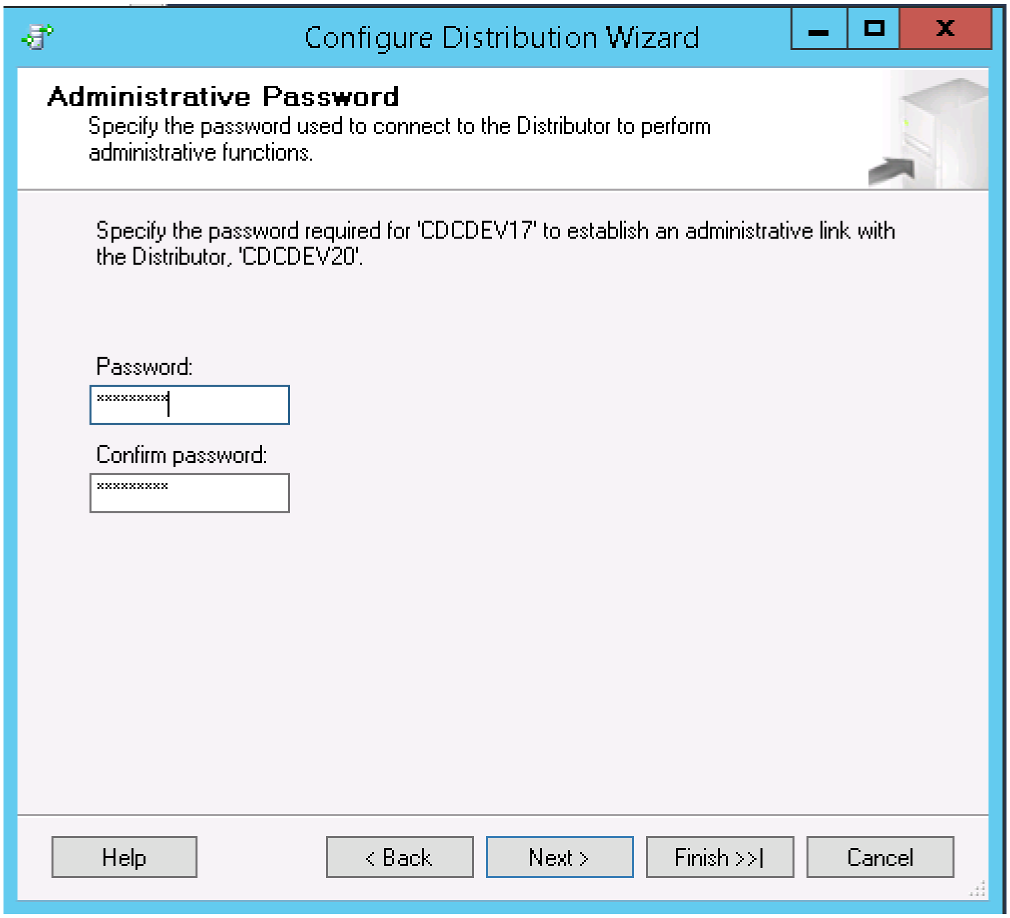
The Distributor is the server responsible for storing replication information used during synchronizations.

○ 'CDCDEV17' will act as its own Distributor; SQL Server will create a distribution database and log

● Use the following server as the Distributor (Note: the server you select must already be configured as a Distributor):

CDCDEV20

In the Administrative Password dialog, type in the same password that was used when configuring the remote distributor in the previous configuration step.



Administrative Password

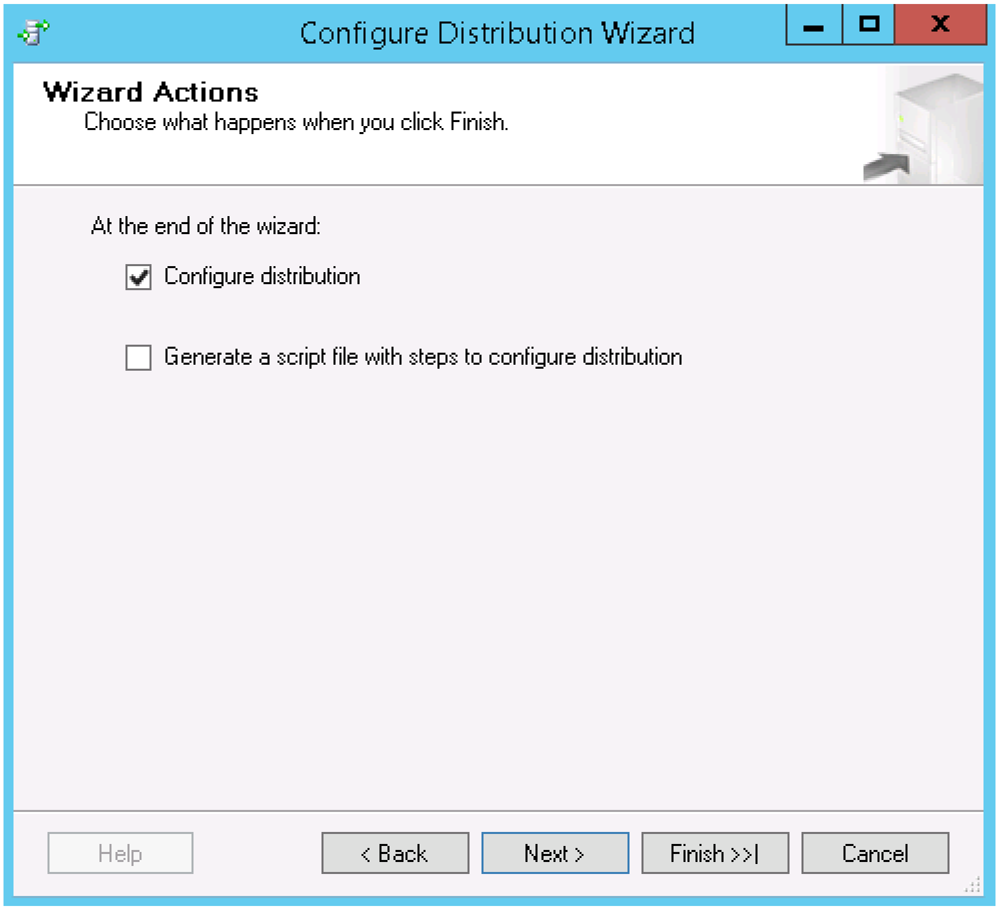
Specify the password used to connect to the Distributor to perform administrative functions.

Specify the password required for 'CDCDEV17' to establish an administrative link with the Distributor, 'CDCDEV20'.

Password: *********

Confirm password: *********

In the Wizard Action dialog accept the defaults and click Finish.



Configure Distribution Wizard

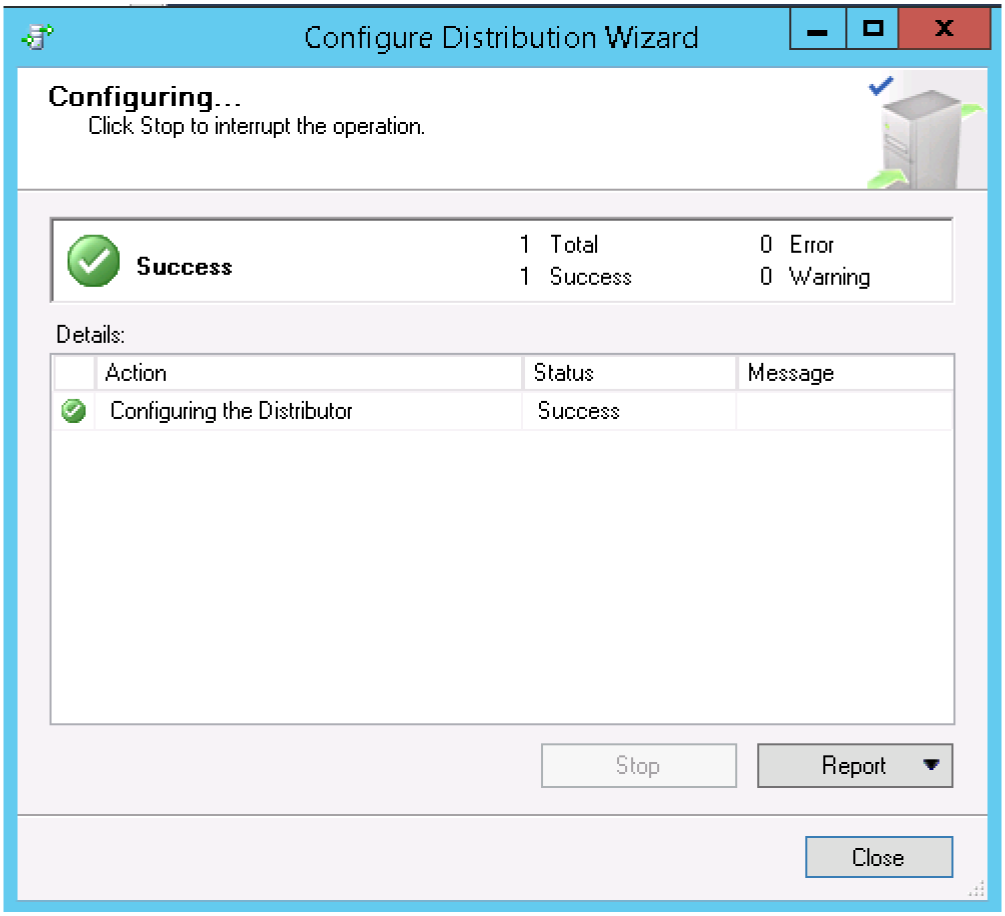
Wizard Actions

Choose what happens when you click Finish.

At the end of the wizard:

☑ Configure distribution

□ Generate a script file with steps to configure distribution



Configure Distribution Wizard

Configuring...

Click Stop to interrupt the operation.

Success

1 Total
1 Success
0 Error
0 Warning

Details:

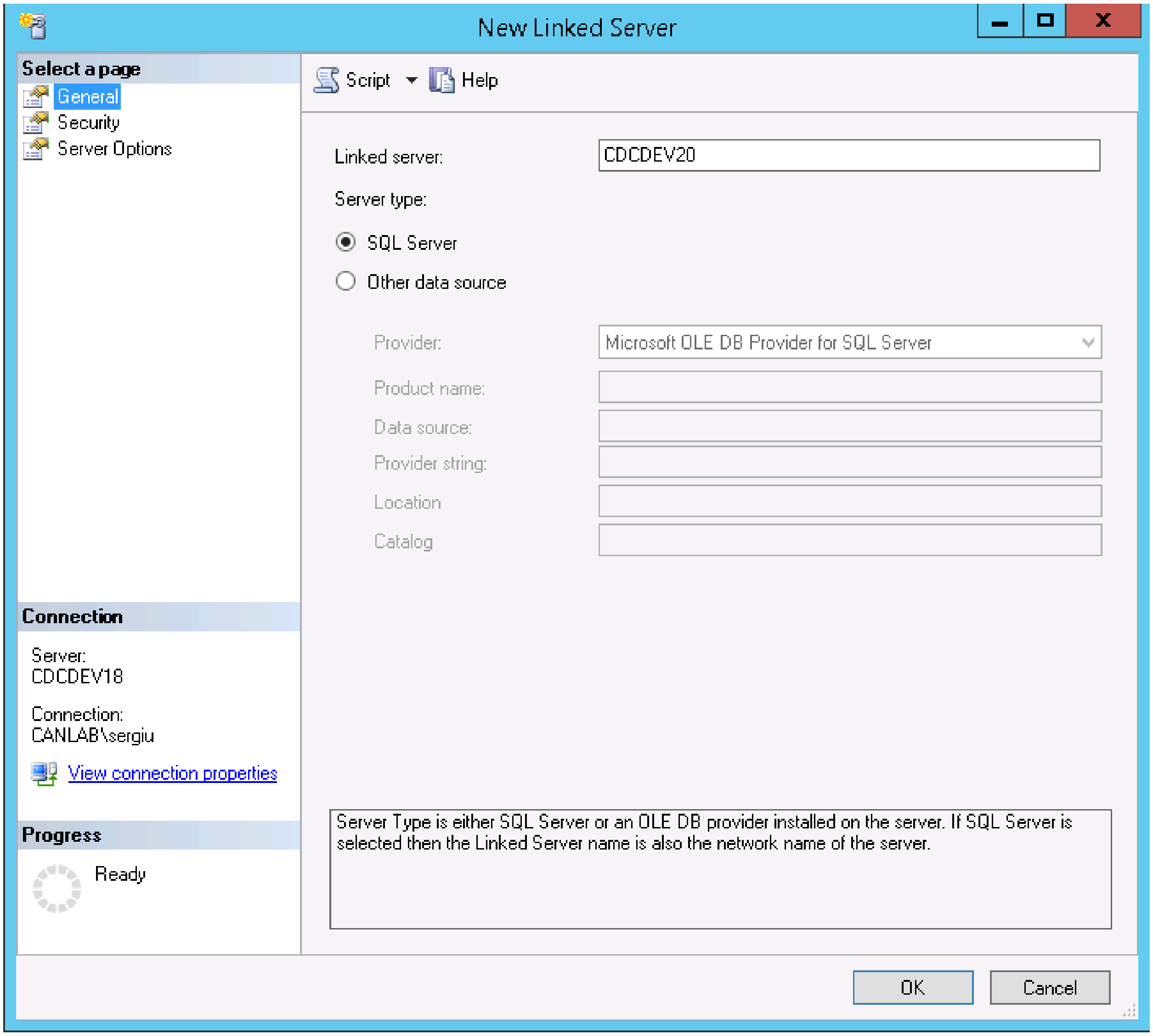
	Action	Status	Message
✓	Configuring the Distributor	Success	

Repeat the steps to configure the remote distributor on the rest of the nodes that could become publishers: CDCDEV18 because it is a SYNCHRONOUS replica. CDCDEV19 does not need this because it is configured as an ASYNCHRONOUS replica.

3.3 Configure all replicas as Replication publishers

For each of the possible publishers (CDCDEV17, CDCDEV18 synchronous replicas only), create a linked server with the name of the remote distributor.

- Right click Server Objects -> Linked Servers and select Add Linked Server
- Type in the linked server name: CDCDEV20.
- Choose the SQL Server type
- Configure security for the linked server
- On the server options enable RPC Out
- Click Ok



Linked server: CDCDEV20

Server type:

● SQL Server

○ Other data source

Provider: Microsoft OLE DB Provider for SQL Server

Product name:

Data source:

Provider string:

Location

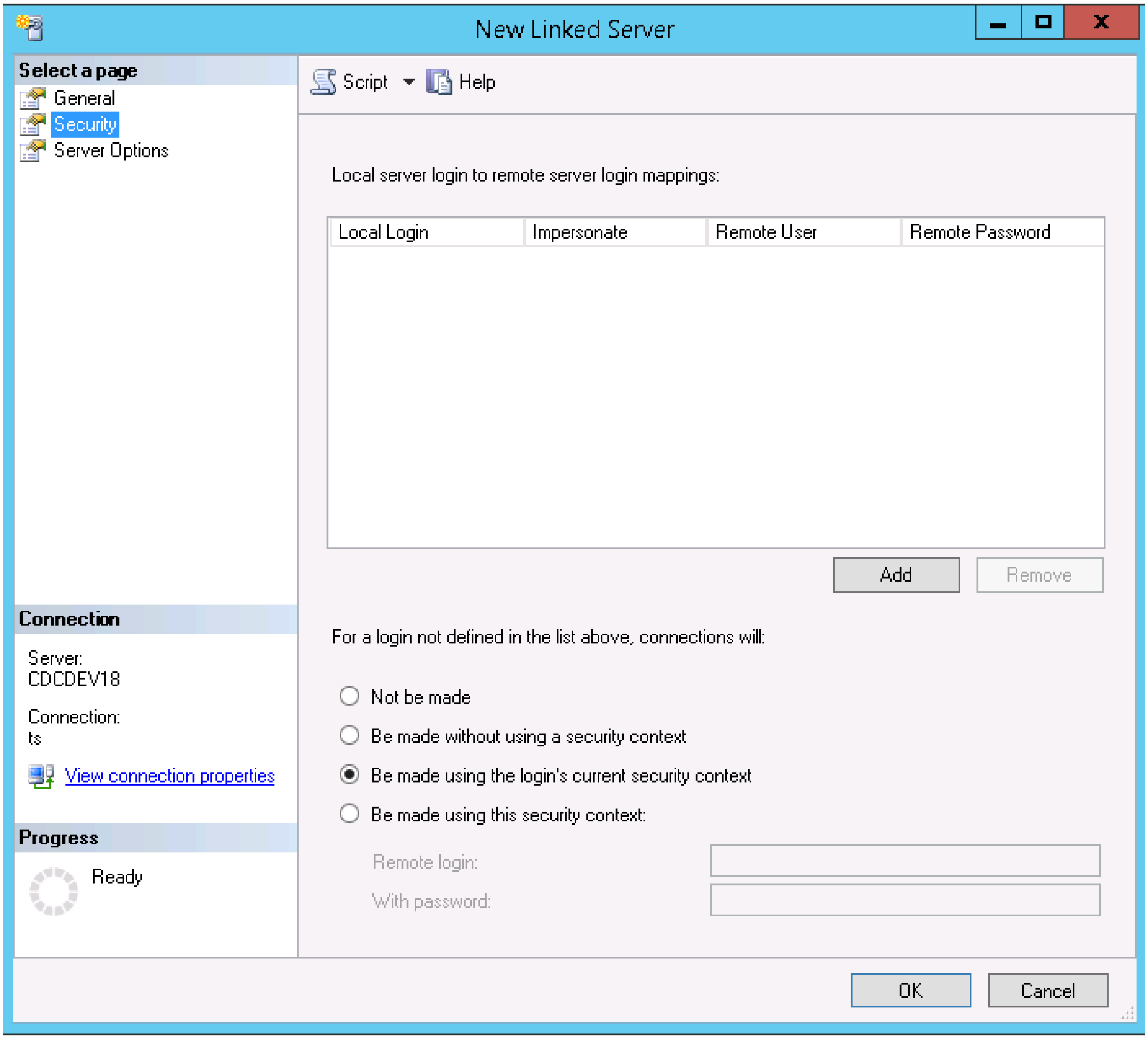
Catalog

Server Type is either SQL Server or an OLE DB provider installed on the server. If SQL Server is selected then the Linked Server name is also the network name of the server.

Server: CDCDEV18

Connection: CANLAB\sergiu

Ready



Connection

Server:
CDCDEV18

Connection:
ts

Local server login to remote server login mappings:

Local Login	Impersonate	Remote User	Remote Password

For a login not defined in the list above, connections will:

○ Not be made

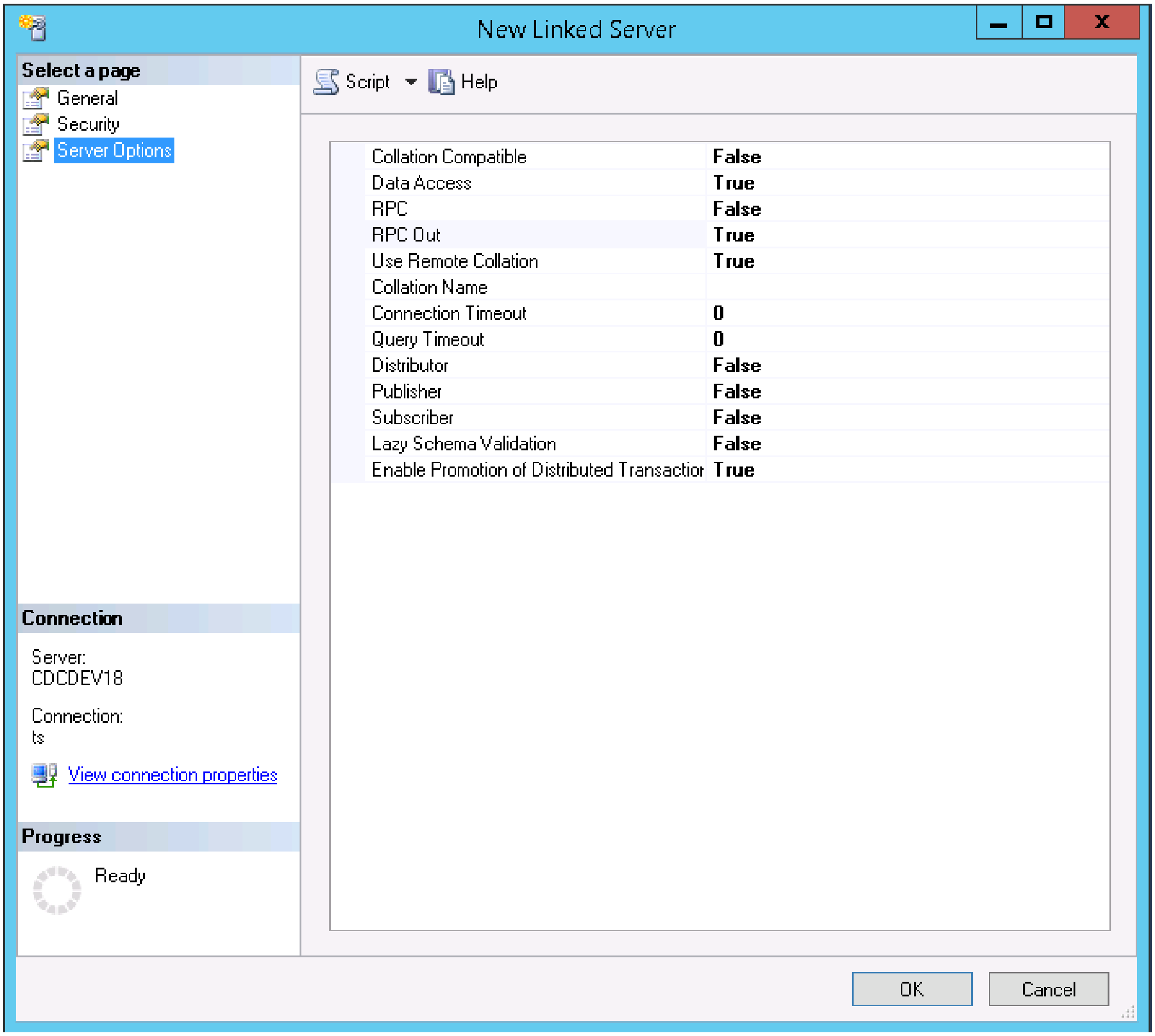
○ Be made without using a security context

● Be made using the login's current security context

○ Be made using this security context:

Remote login:

With password:



Collation Compatible	False
Data Access	True
RPC	False
RPC Out	True
Use Remote Collation	True
Collation Name	
Connection Timeout	0
Query Timeout	0
Distributor	False
Publisher	False
Subscriber	False
Lazy Schema Validation	False
Enable Promotion of Distributed Transactior	True

Server:
CDCDEV18

Connection:
ts

Alternatively, use stored procedures to configure the linked server. For additional information, please refer to Microsoft documentation on configuring Linked servers.

3.4 Redirect the original publisher to the AlwaysOn Listener Name

At the distributor, in the distribution database, run the stored procedure sp_redirect_publisher to associate the original publisher and the published database with the availability group listener name of the availability group.

```
USE distribution;
GO
EXEC sys.sp_redirect_publisher
@original_publisher = 'CDCDEV17',
   @publisher_db = 'TestDb',
   @redirected_publisher = 'cdcsql2014lsn1,5025';
```

Review Microsoft documentation on maintenance steps for the redirected publisher. Some of possible documents to review are:

https://msdn.microsoft.com/en-us/library/hh710048.aspx
https://msdn.microsoft.com/en-us/library/hh403414.aspx

3.5 Run the Replication Validation Stored Procedure to Verify the Configuration

At the distributor, in the distribution database, run the stored procedure sp_validate_replica_hosts_as_publishers to verify that all replica hosts are now configured to serve as publishers for the published database.

```
 USE distribution;
DECLARE @redirected_publisher sysname;
EXEC sys.sp_validate_replica_hosts_as_publishers
@original_publisher = 'CDCDEV17',
@publisher_db = 'TestDb',
@redirected_publisher = @redirected_publisher output;


select @redirected_publisher
```

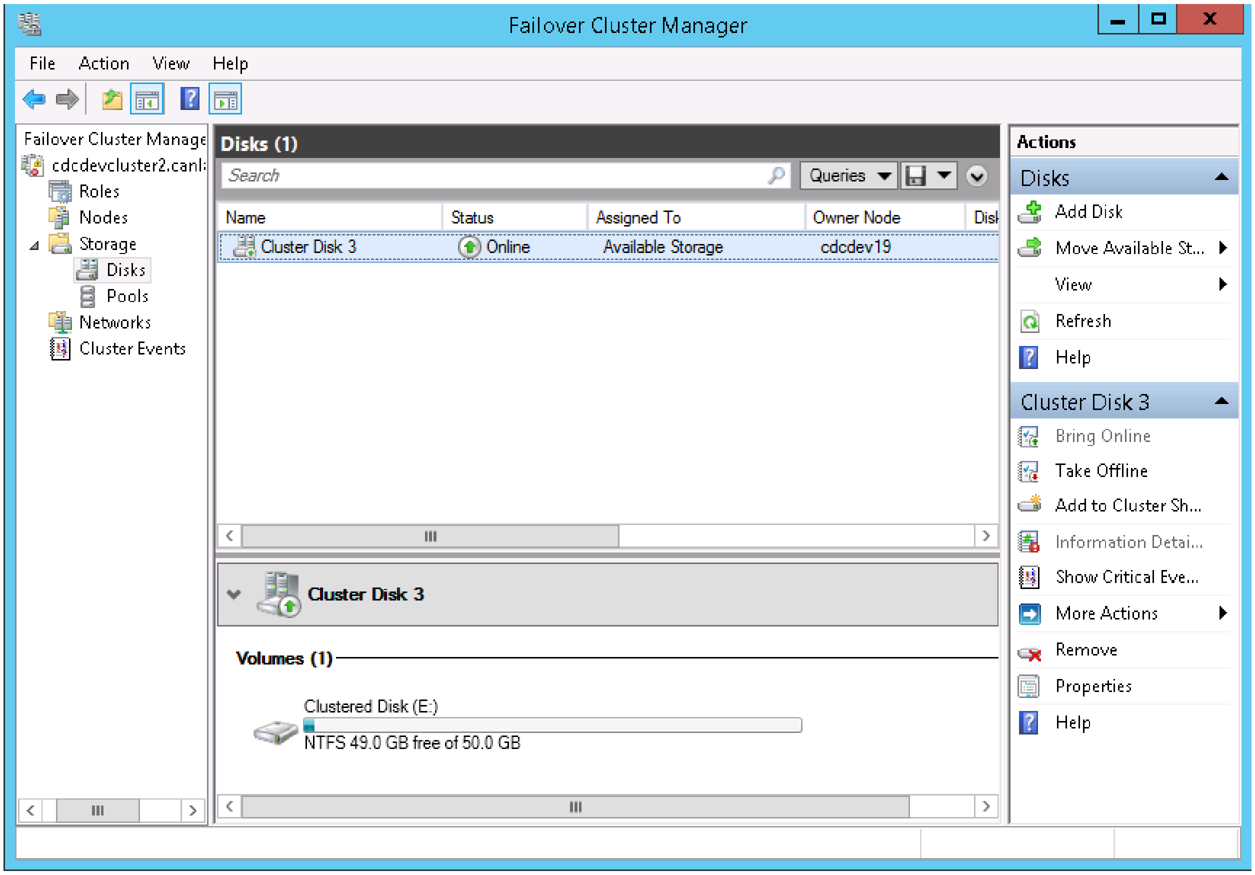
Review Microsoft documentation on maintaining the redirected original publisher. Review Microsoft documentation on maintenance steps for the redirected publisher. Some of possible documents to review are:

https://msdn.microsoft.com/en-us/library/hh710048.aspx
https://msdn.microsoft.com/en-us/library/hh403414.aspx

4 Add symmetrical shared clustered disk reachable from all AlwaysOn nodes

In order to continue with the replication preparation, shared clustered disk that will hold the IIDR binaries and metadata has to be defined as a dependent resource in the AlwaysOn Role that was created in the Windows Failover Cluster for the AlwaysOn deployment. These steps are similar to the steps that are required to be performed when adding shared clustered disk for the Microsoft SQL Server Clustering solution: https://msdn.microsoft.com/en-us/library/ms179530.aspx. More details on how to add Shared clustered disk can be found at:
http://blogs.msdn.com/b/clustering/archive/2012/06/02/10314262.aspx

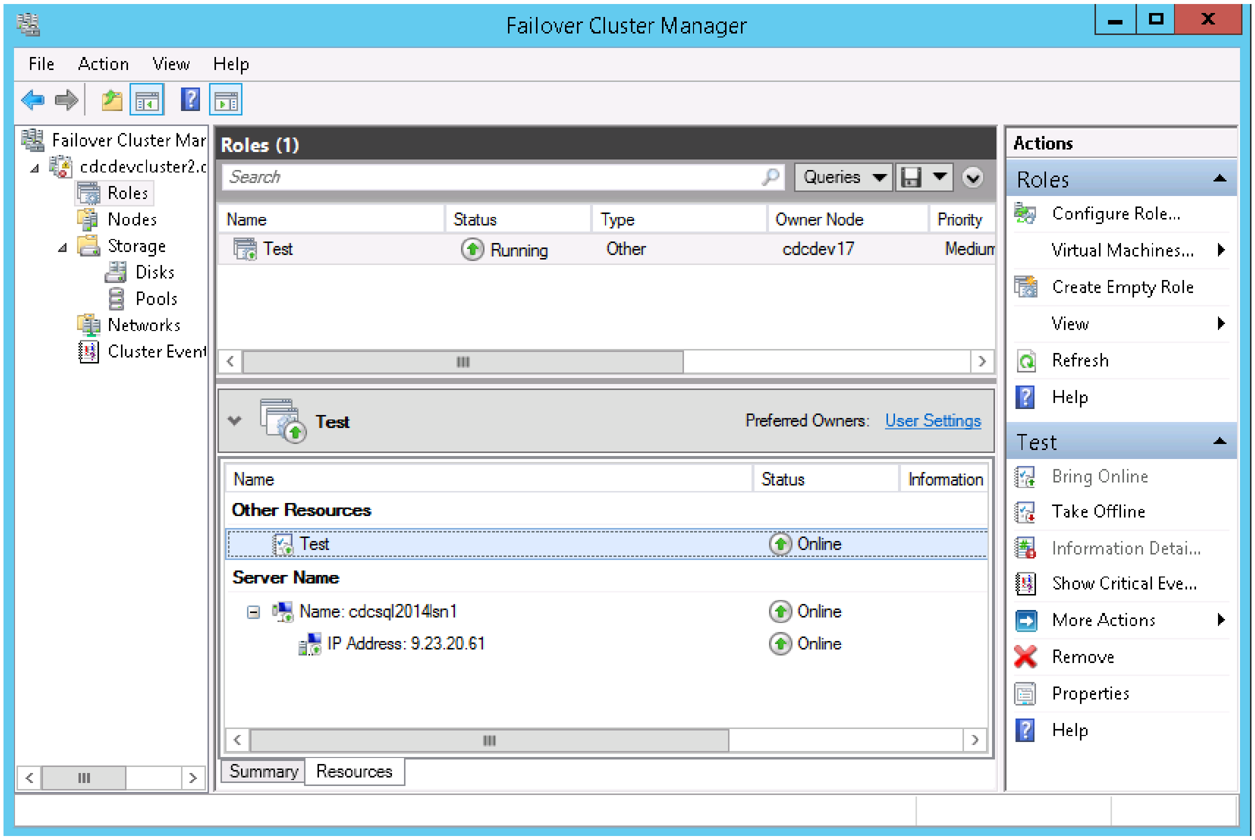
Add your clustered disk to the Storage -> Disks.



Name	Status	Assigned To	Owner Node	Disk
Cluster Disk 3	Online	Available Storage	cdcdev19	

Clustered Disk (E:)
NTFS 49.0 GB free of 50.0 GB

Open the Failover Cluster Manager and find the Role that was defined by the AlwaysOn configuration.



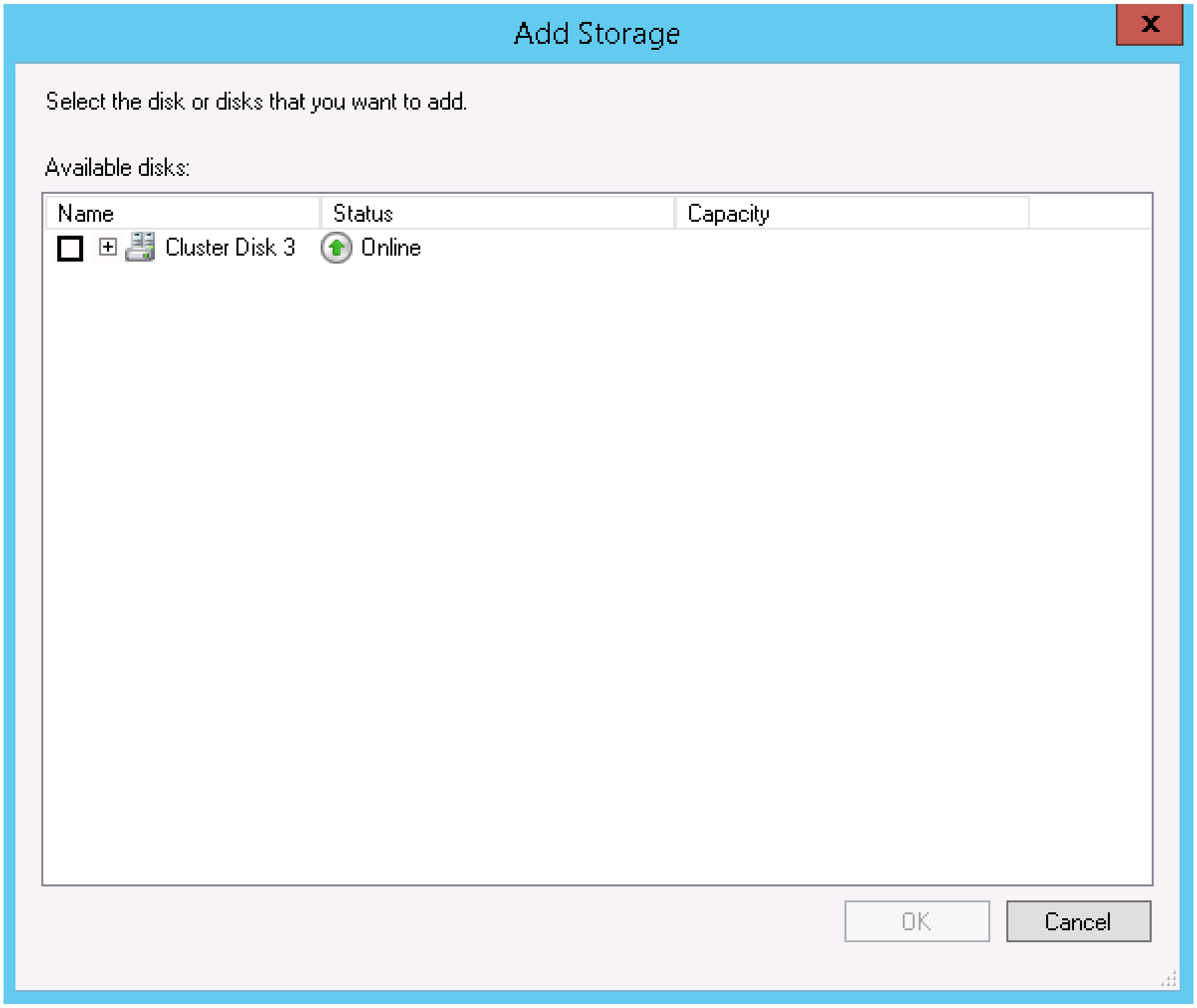
Name	Status	Type	Owner Node	Priority
Test	Running	Other	cdcdev17	Mediur

Test

Preferred Owners: User Settings

Name	Status	Information
Other Resources		
Test	Online	
Server Name		
Name: cdcsql2014lsn1	Online	
IP Address: 9.23.20.61	Online	

Right click the AlwaysOn role and select add storage.

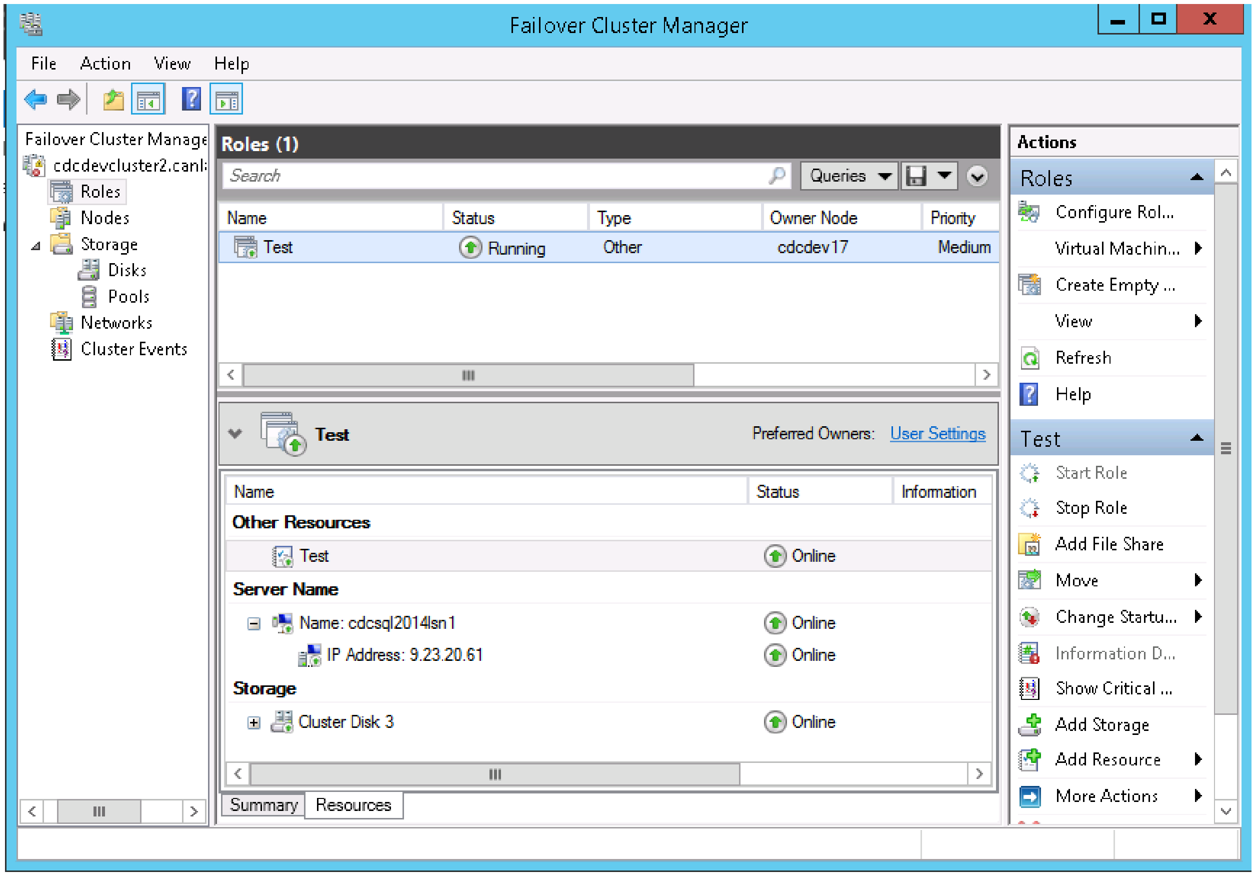


Select the disk or disks that you want to add.

Available disks:

Name	Status	Capacity
☐ Cluster Disk 3	Online	

Select the disk and click OK.



Name	Status	Type	Owner Node	Priority
Test	Running	Other	cdcdev17	Medium

Test

Name	Status	Information
Other Resources		
Test	Online	
Server Name		
Name: cdcsql2014lsn1	Online	
IP Address: 9.23.20.61	Online	
Storage		
Cluster Disk 3	Online	

5 Define a Microsoft Windows domain user that can be used to run the IIDR service

Given the distributed nature of the AlwaysOn deployment and the fact that IIDR will need to read logs from network share, you need to have a Windows domain user that has network resource access.

5.1 Transaction log backups shared storage

Identify shared storage to which the transaction logs can be backed up. The storage has to be symmetrically accessible from all AlwaysOn nodes.

The backup strategy is expanded in AlwaysOn deployments. One can choose to perform different types of backups on any number of nodes. The metadata about the backups is found in the msdb database that is present on the SQL Server instance deployed on the node that was responsible for taking the backup. The information about the transaction log backups is important to IIDR as it permits identifying which transaction log backups contain the data that needs to be processed next. Accessing the physical transaction log files is just as important for IIDR. Immediately after CDC identifies which files contain the data that needs to be processed next, CDC will try to access and read data from those transaction log backups. All backups taken from any Microsoft SQL Server instance need to be taken to a similar shared location. More information can be found at:

http://blogs.msdn.com/b/sqlgardner/archive/2012/07/18/sql-2012-alwayson-and-backups-part-1-offloading-the-work-to-a-replica.aspx

http://blogs.msdn.com/b/sqlgardner/archive/2012/07/21/sql-2012-alwayson-and-backups-part-2-configuring-backup-preferences-and-automating-backups.aspx

http://blogs.msdn.com/b/sqlgardner/archive/2012/08/28/sql-2012-alwayson-and-backups-part-3-restore.aspx

If the transaction log backup is already found in a shared network directory, make sure that it is readable by the IIDR process user.

If the backup directory is not a network drive, then:

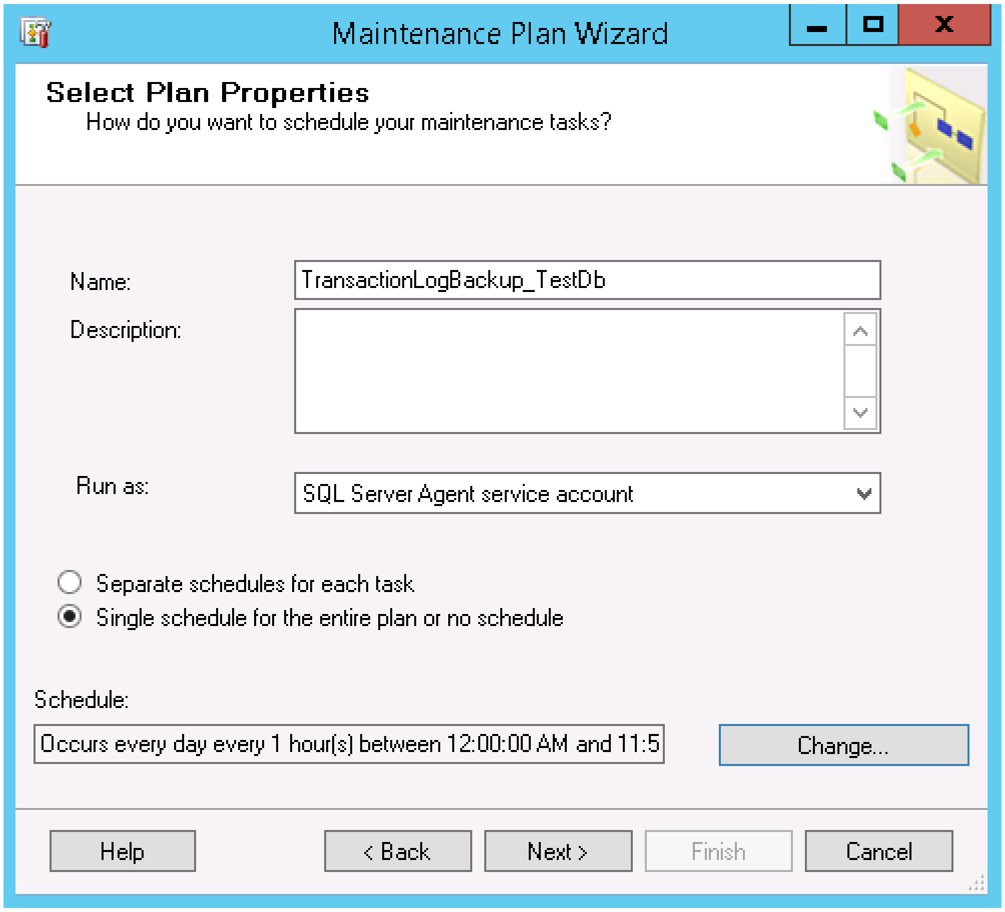
- Define a network directory (UNC path, or mapped drive), that is symmetrically accessible from all nodes
- Microsoft SQL Server service user has all the privileges that are needed by Microsoft SQL Server to create successful backups to this shared network directory. Pease see Microsoft documentation for details
- Ensure that IIDR service user has read access to this network directory

For the purpose of our test, we defined \\CDCDEV20\Backups with all the required access privileges.

6 Microsoft SQL Server transaction log maintenance plans

As mentioned in section 5.1, AlwaysOn expands on the Microsoft backup. The backups, including transaction log backups, can be offloaded to any replica. Given that the metadata for the backups is found in the msdb database and it cannot be part of an availability group, the metadata about the transaction log backups will only be found on the Microsoft SQL Server instance that created the backup. The next section will describe how to set up transaction log backup using a Microsoft maintenance plan.

- Right click on CDCDEV17 Management-> Maintenance plans and select “Maintenance Plan Wizard”. Click Next
- Provide an identifiable name for the plan. For ease of maintenance, you may want to keep the plan names consistent and unique across all the Microsoft SQL Server instance where they may be defined
- Define a schedule for the maintenance plan



Select Plan Properties

How do you want to schedule your maintenance tasks?

Name: TransactionLogBackup_TestDb

Description:

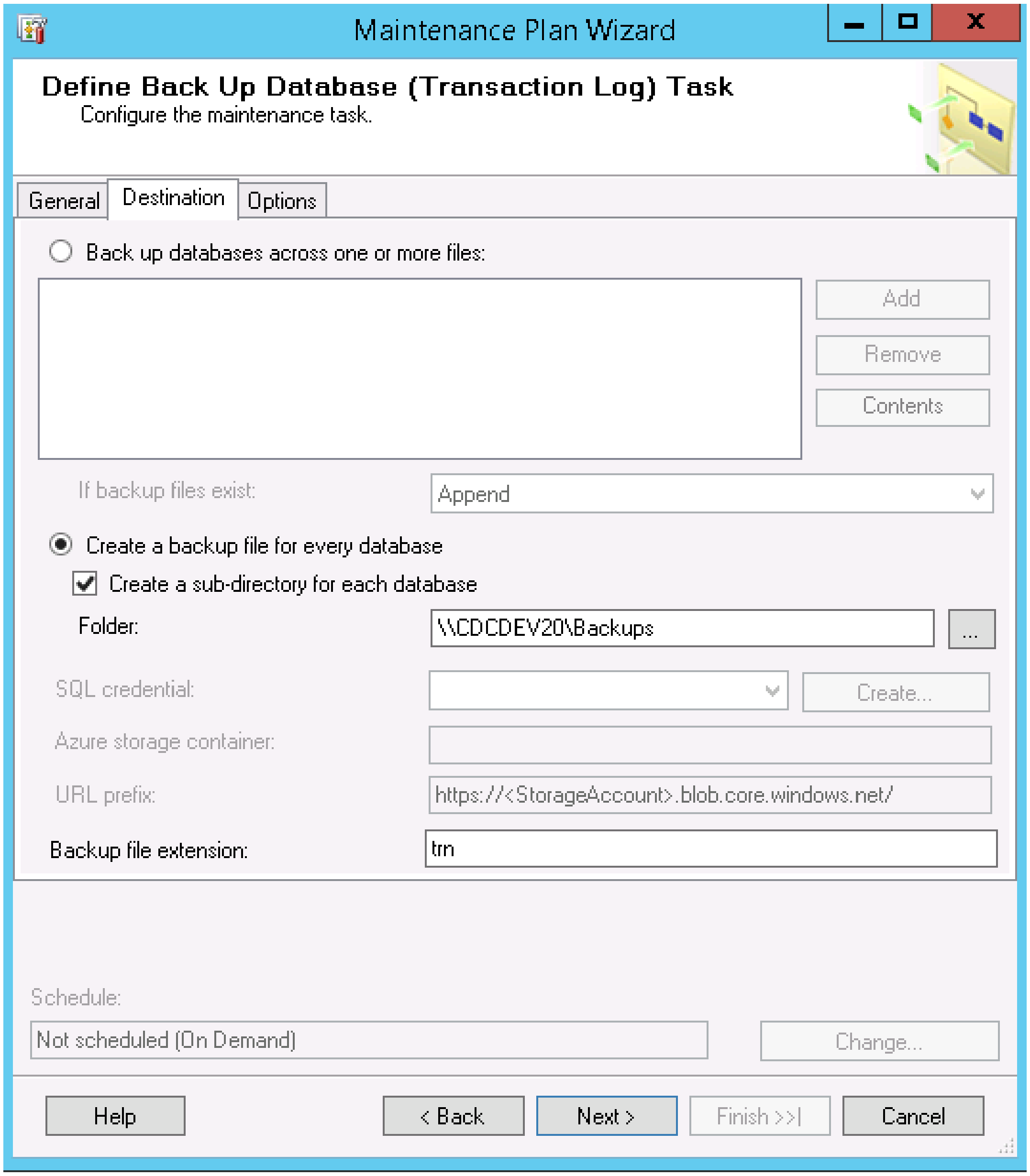
Run as: SQL Server Agent service account

○ Separate schedules for each task

● Single schedule for the entire plan or no schedule

Schedule: Occurs every day every 1 hour(s) between 12:00:00 AM and 11:5

- Click Next. Choose “Backup Database (Transaction Log)”. Click Next
- Choose the database that you want to be backed up
- On the Destination tab, type in the shared folder \\CDCDEV20\Backups where the backups need to be taken



Define Back Up Database (Transaction Log) Task

Configure the maintenance task.

○ Back up databases across one or more files:

If backup files exist: Append

● Create a backup file for every database

☑ Create a sub-directory for each database

Folder: \\CDCDEV20\Backups

SQL credential:

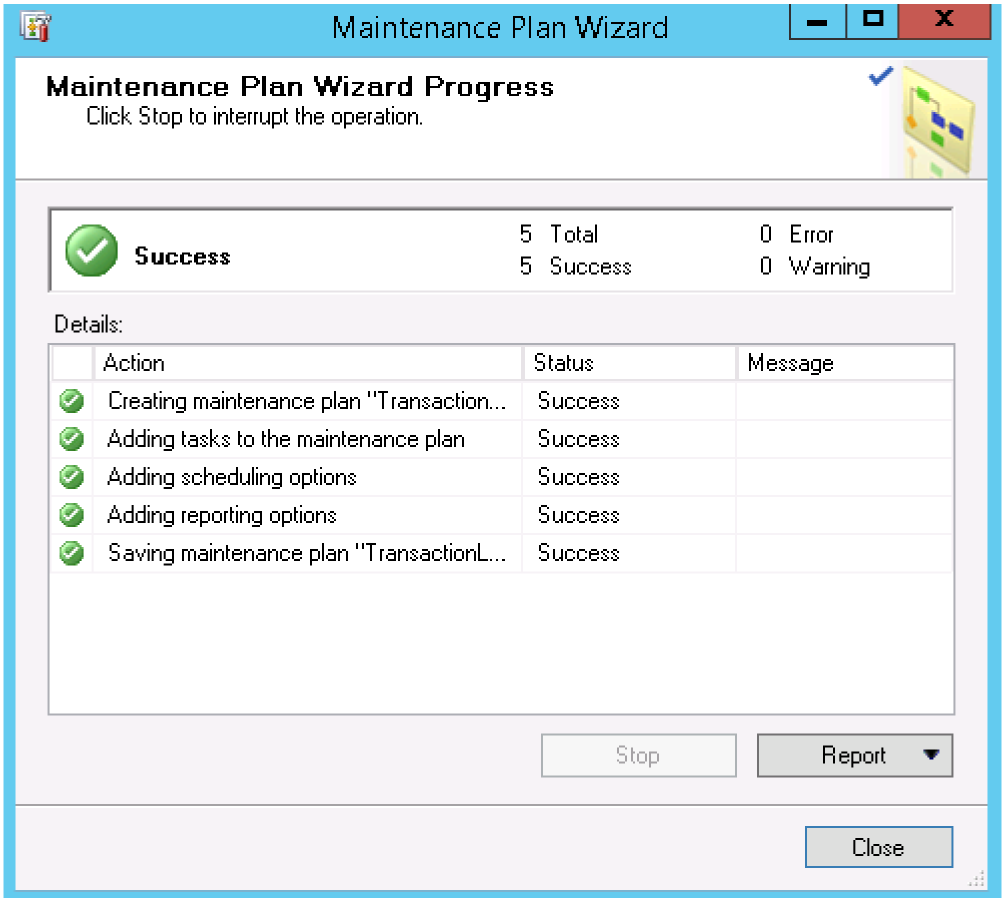
Azure storage container:

URL prefix: https://<StorageAccount>.blob.core.windows.net/

Backup file extension: trn

Schedule: Not scheduled (On Demand)

- Click Next. Choose a shared folder where you want the report to be written. You could choose the \\CDCDEV20\Backups as the location assuming you checked the “Create a sub-directory for each database” on the previous screen. IIDR will issue a warning if the directory that is supposed to contain transaction log backups contains files of other types when it discovers metadata about transaction logs based on the files found in that directory
- Click Next. Click Finish


Maintenance Plan Wizard
Maintenance Plan Wizard Progress
Click Stop to interrupt the operation.
Success
5 Total
5 Success
0 Error
0 Warning
Details:

	Action	Status	Message
✓	Creating maintenance plan "Transaction...	Success	
✓	Adding tasks to the maintenance plan	Success	
✓	Adding scheduling options	Success	
✓	Adding reporting options	Success	
✓	Saving maintenance plan "TransactionL...	Success	

- Repeat the Maintenance plan creation on any of nodes that you want to participate in maintaining the chain of transaction log backups

For more information on maintenance plans or AlwaysOn backup strategies, please see Microsoft SQL Server documentation.

7 IIDR Database authentication user

Create an IIDR user able to connect to the database on all nodes where it may fail over.

In order to have IIDR fail over smoothly from a primary node to a new primary node, it will need to be able to connect to the database with the same user name and password and privileges on any availability replica to which it could fail over and any node where a transaction log backup could be taken. Now you can define the user with which IIDR will connect to a Microsoft SQL Server instance. This user can be used for data replication, table metadata discovery, or for connecting to the msdb database in order to extract metadata about transaction log backups.

- On each Microsoft SQL Server instance, right click Security and select New Login… Choose the user name/password, and role for the IIDR user. For more information, see:

http://www-01.ibm.com/support/knowledgecenter/SSTRGZ_11.3.3/com.ibm.cdcdoc.cdcformssql.doc/concepts/creatingawindowsaccountinsqlserver.dita?lang=en

8 Deploy IIDR on the shared clustered disk

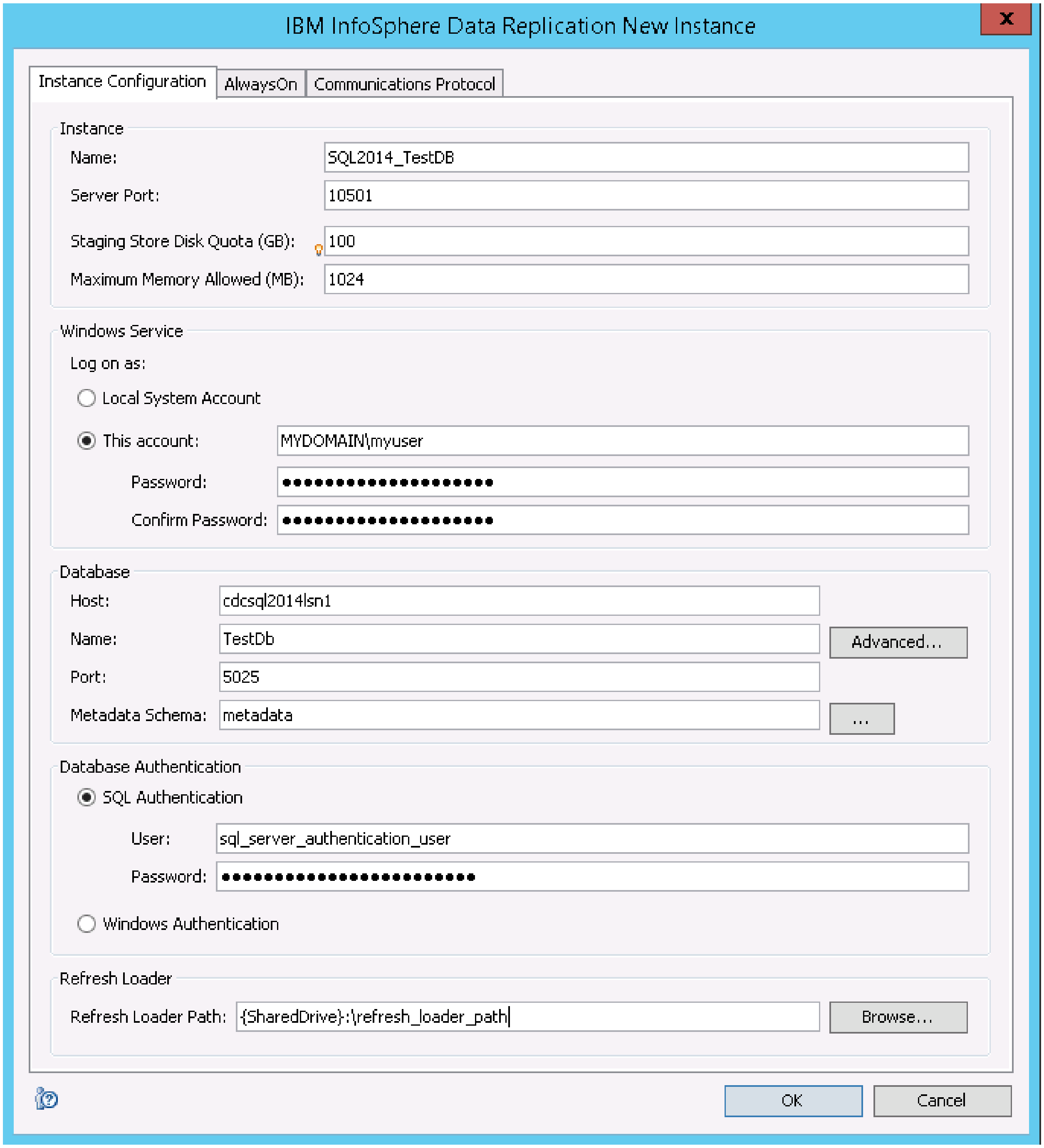
Now that we have:

- Prepared the AlwaysOn environment for replication
- Added symmetrical shared storage for IIDR metadata
- Have a windows domain user to run IIDR service under
- Identified shared storage to which the transaction logs can be backed up
- Defined maintenance plans to back up the transaction logs to the backup storage
- Created the IIDR user to connect to Microsoft SQL Server instances

We can proceed with deploying IIDR product on the AlwaysOn availability group that holds the current PRIMARY replica for the database. The node that holds the current PRIMARY replica should also be the owner of the shared symmetrical storage where IIIDR needs to be deployed. In our example, CDCDEV17 is the current primary replica and the owner of the primary replica. The deployment is similar to any deploying IIDR for Microsoft SQL Server cluster. The difference in deployment is that Microsoft SQL Server does not use the shared storage in this case to store the database and the logs. For more details see:
http://www-01.ibm.com/support/knowledgecenter/SSTRGZ_11.3.3/com.ibm.cdcdoc.cdcformssql.doc/concepts/configuringclustered.dita?lang=en

To deploy IIDR for Microsoft SQL Server:

- Choose the installation directory and the instance directory to be on the symmetrical shared storage.
- After deploying the product, one can create an instance. Launch the Configuration tool to create the IIDR instance that would replicate data from the database that is part of an AlwaysOn availability group.



IBM InfoSphere Data Replication New Instance

Instance Configuration | AlwaysOn | Communications Protocol

Instance

Name: SQL2014_TestDB

Server Port: 10501

Staging Store Disk Quota (GB): 100

Maximum Memory Allowed (MB): 1024

Windows Service

Log on as:

○ Local System Account

● This account: MYDOMAIN\myuser

Password: ●●●●●●●●●●●●●●●●●●●●

Confirm Password: ●●●●●●●●●●●●●●●●●●●●

Database

Host: cdcsql2014lsn1

Name: TestDb

Port: 5025

Metadata Schema: metadata

Database Authentication

● SQL Authentication

User: sql_server_authentication_user

Password: ●●●●●●●●●●●●●●●●●●●●●●●●

○ Windows Authentication

Refresh Loader

Refresh Loader Path: {SharedDrive}:\refresh_loader_path

The data that needs to be filled out in the configuration tool is not much different from creating an instance in non AlwaysOn environment. A few things to pay attention to:

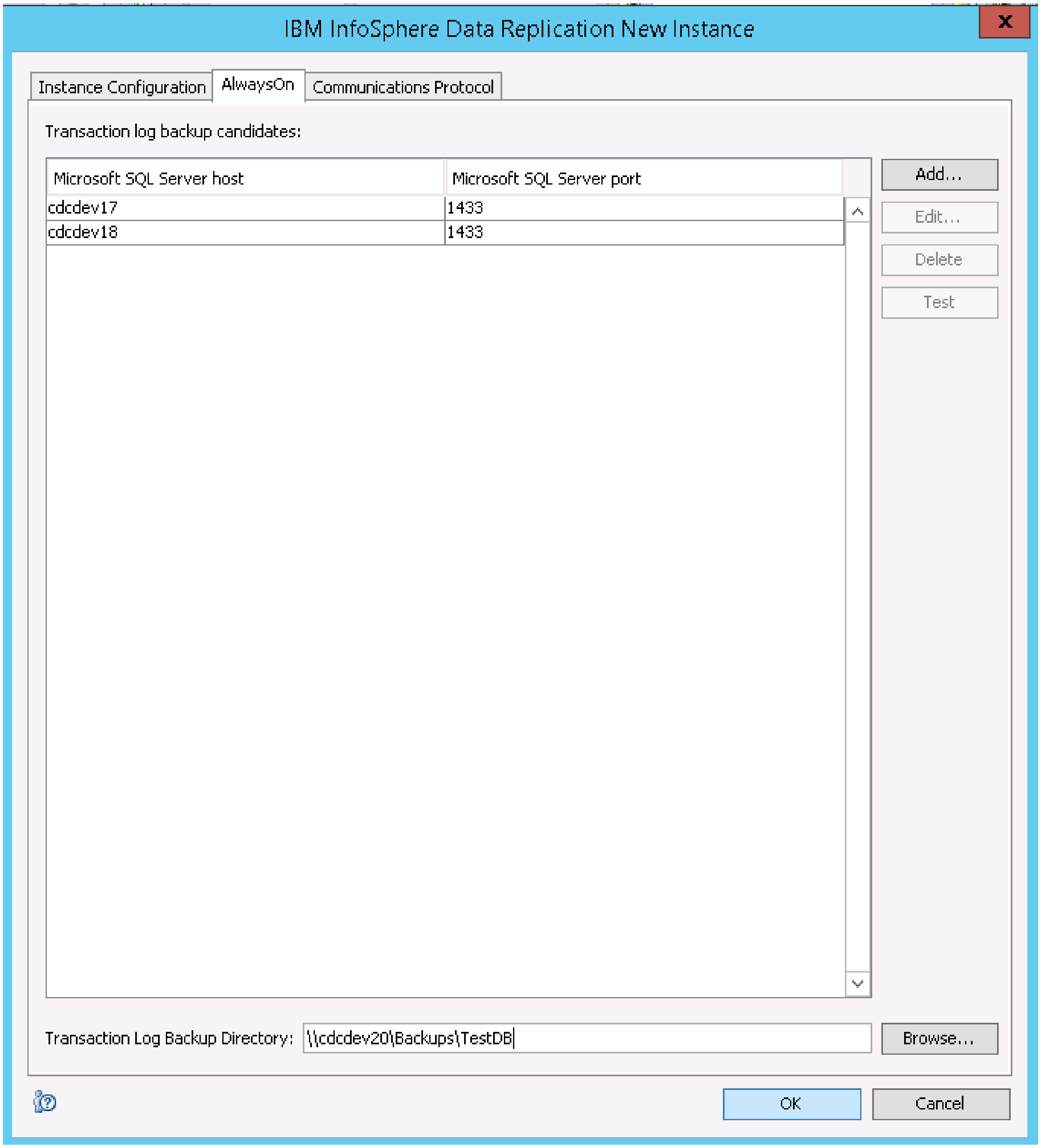
- For Log on As, choose This account and type in a Windows user account that has access to network resources.
- For host type the AlwaysOn availability group listener name that your database belongs to.
- For Port type the port number for the AlwaysOn availability group.

- Refresh Loader path needs to be a valid path on any node IIDR may fail over to. One option to ensure that is to use the same Shared symmetrical disk used for IIDR installation as the refresh_loader_path.

The transaction log backups can be taken by any Microsoft SQL Server instance in the AlwaysOn availability group cluster. As such the metadata about the files may be present in the msdb database on different Microsoft SQL Server instances. In order to determine which transaction log backup files need to be processed, IIDR queries the msdb database on all the nodes from the list until it finds information about the transaction log backup that needs to be replicated.

However, in case of a failure, the Microsoft SQL Server instance where the backup was taken may be down and as such the msdb database will be unreachable. In order to work around this possible scenario, the directory where the transaction log backups are taken also needs to be provided.

Navigate to the AlwaysOn tab:



IBM InfoSphere Data Replication New Instance

Transaction log backup candidates:

Microsoft SQL Server host	Microsoft SQL Server port
cdcdev17	1433
cdcdev18	1433

Transaction Log Backup Directory: \\cdcdev20\Backups\TestDB

- Provide a list of Microsoft SQL Server instances that could serve as transaction log backup candidates
  - IIDR will use the user name and the password from the instance configuration to connect to the transaction log backup candidates at the provided hosts and ports to extract metadata about transaction log backups
- Provide a folder where all the transaction log backups are taken. IIDR will use this location to extract the transaction log backup metadata when the transaction log backup candidate

where the transaction log backup was taken is not reachable. A few things to pay attention to:

- The folder has to be symmetrically reachable from all nodes where IIDR could fail over
- The folder has to exist
- The folder has to be readable by the IIDR process user
- The folder needs to contain only transaction log backup files
- Do not mix transaction log backups from multiple databases in the same folder
- The folder should not contain subdirectories. The subdirectories will not be processed
- A large number of files in this directory can slow down IIDR responsiveness. A warning will be issued when over 1000 files are found in the directory

9 Create IIDR service on secondary synchronous replicas

Do not start the instance after it is created. Similar services that were created on the current node that is responsible for the active PRIMARY replica need to be created on each SECONDARY replica node where IIDR may fail over. Navigate to IIDR installation directory and find the command dmcreateclusterservice.exe and invoke the command for each replica to which IIDR may fail over.

dmcreateclusterservice.exe -I *<IIDR instance_name> <secondary_replica_node>*

For the test example that we have, we will invoke the command below. IIDR can only fail over to SYNCHRONOUS availability mode replicas and the availability group Test is configured to have only CDCDEV18 as a synchronous availability mode replica.

dmcreateclusterservice.exe –I SQL2014_TestDb CDCDEV18

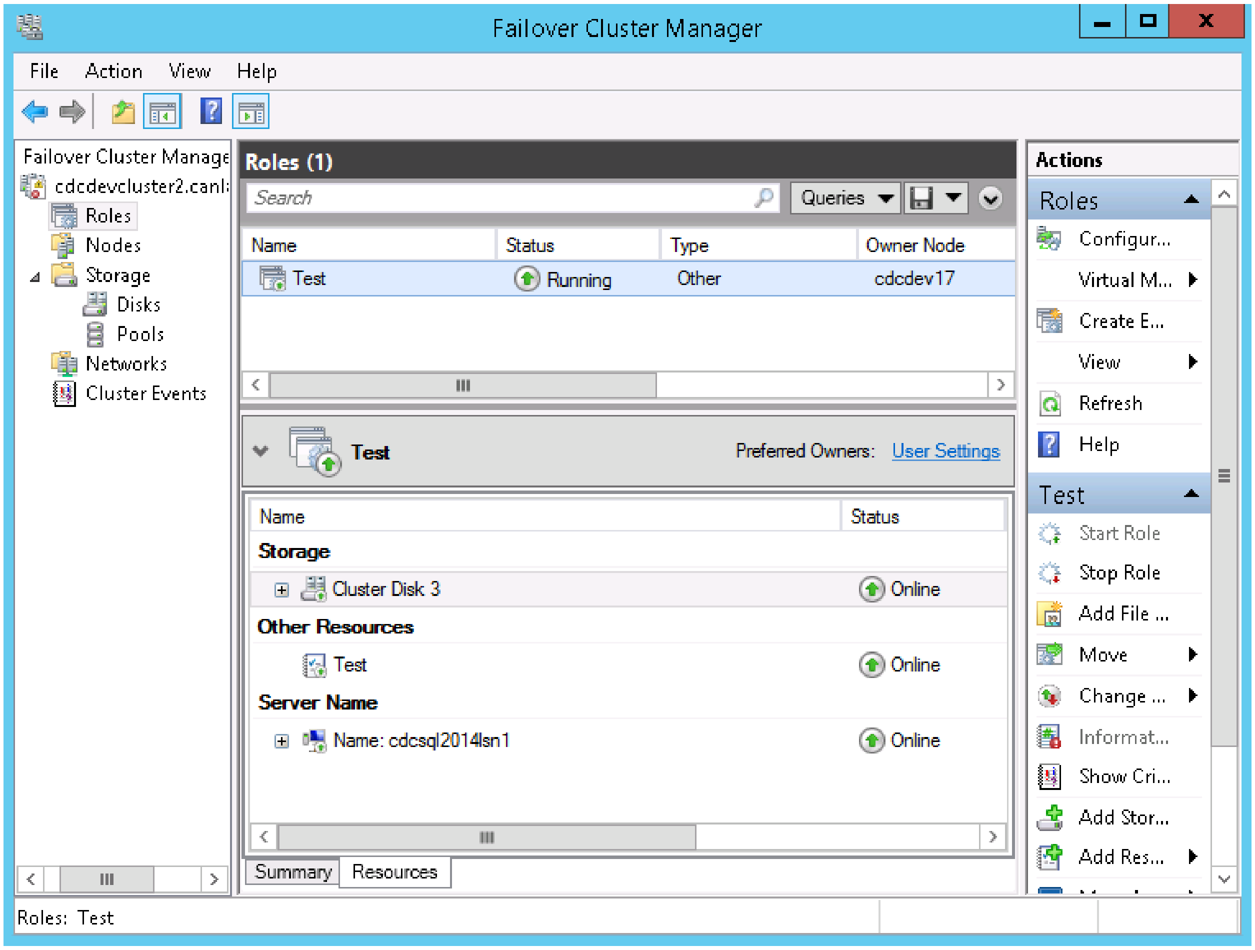
This command will create a service “IBM InfoSphere Data Replication [SQL2014_TestDb]” on the cdcdev18 replica.

Note: After a fail over the CDC service may fail to start. When you start the service manually, it may inform you that there was a logon error. If you edit the user name or the password and save in the Properties -> Log On tab, you may get a confirmation that Log On as a service privilege was granted to the user.

You can also add the privileges manually:

https://technet.microsoft.com/en-us/library/cc794944(v=ws.10).aspx

- While being on the PRIMARY node (CDCDEV17), add the IIDR service as a dependent resource to the AlwaysOn availability group role that was created in the Windows Failover Clustering
  The next step includes adding the IIDR service as a dependent resource to the AlwaysOn availability group role that was created in Windows Failover Cluster at the time when the availability group was defined
- Open Windows Failover Cluster manager and navigate to the role created by the availability group



Roles (1)

Name	Status	Type	Owner Node
Test	Running	Other	cdcdev17

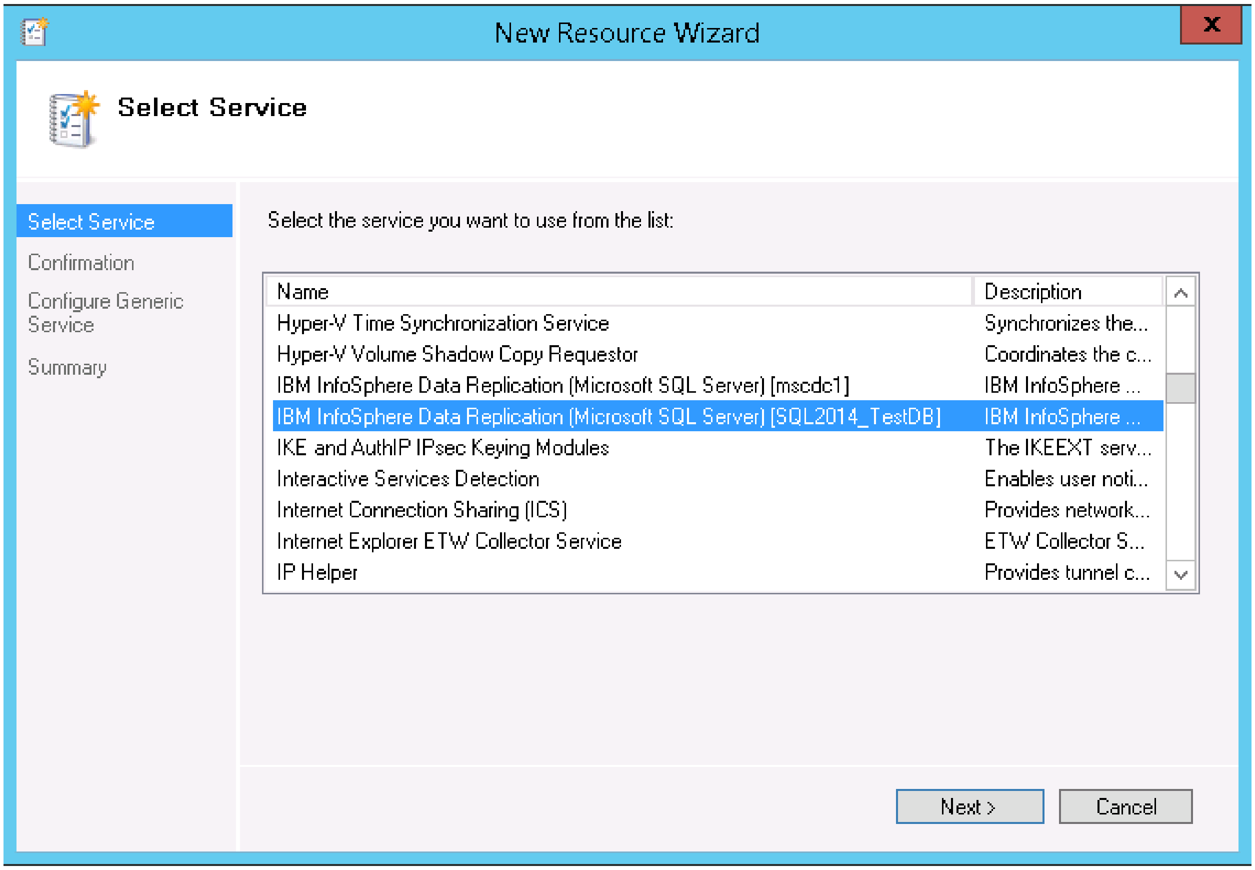
Test

Preferred Owners: User Settings

Name	Status
Storage	
Cluster Disk 3	Online
Other Resources	
Test	Online
Server Name	
Name: cdcsql2014lsn1	Online

Roles: Test

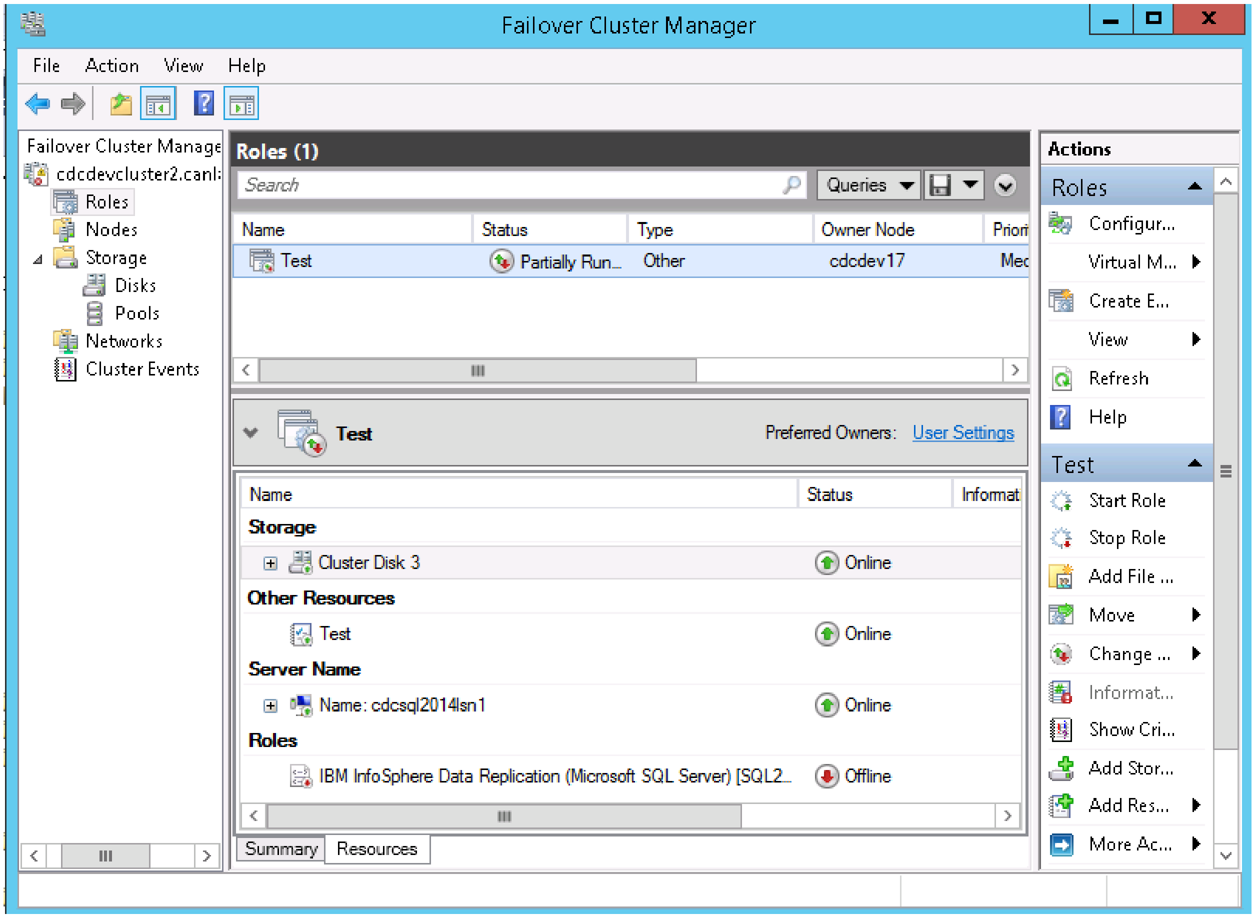
- Right click the Test role and select Add Resource -> Generic Service. Select the service “IBM InfoSphere Data Replication [SQL2014_TestDb]” and click Next



Select the service you want to use from the list:

Name	Description
Hyper-V Time Synchronization Service	Synchronizes the...
Hyper-V Volume Shadow Copy Requestor	Coordinates the c...
IBM InfoSphere Data Replication (Microsoft SQL Server) [mscdc1]	IBM InfoSphere ...
IBM InfoSphere Data Replication (Microsoft SQL Server) [SQL2014_TestDB]	IBM InfoSphere ...
IKE and AuthIP IPsec Keying Modules	The IKEEXT serv...
Interactive Services Detection	Enables user noti...
Internet Connection Sharing (ICS)	Provides network...
Internet Explorer ETW Collector Service	ETW Collector S...
IP Helper	Provides tunnel c...

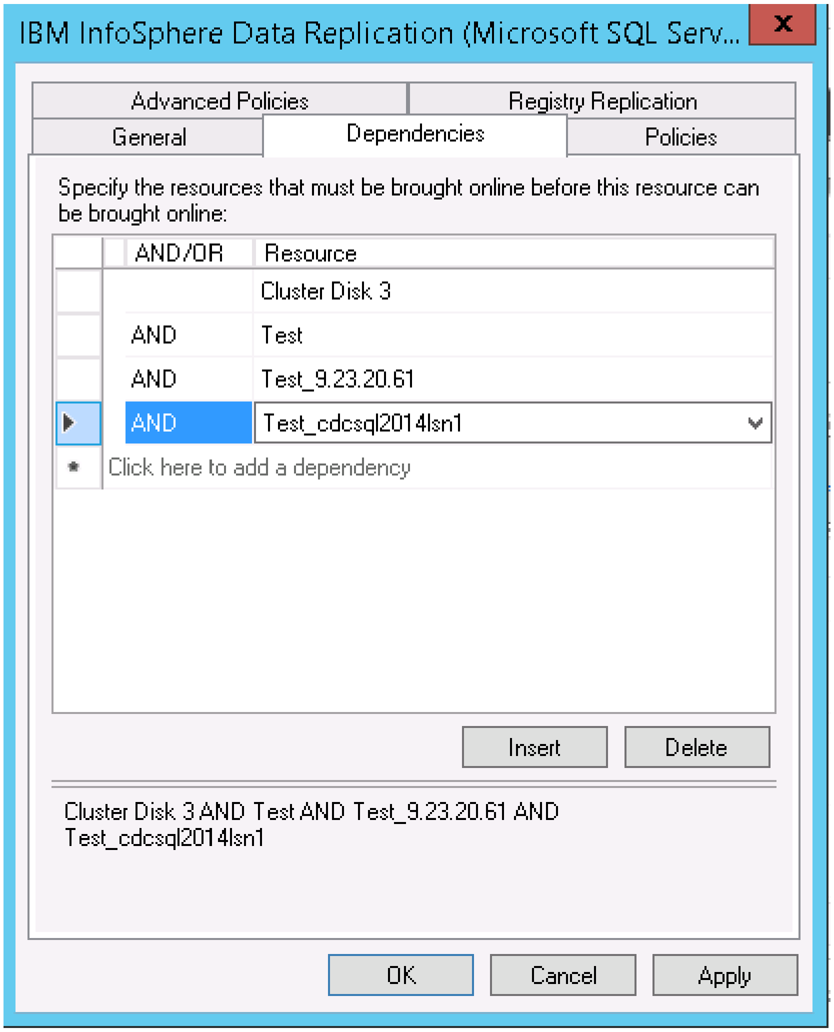
Click Next in Confirmation dialog. Click Finish in the Summary dialog.



Name	Status	Type	Owner Node	Priori
Test	Partially Run...	Other	cdcdev17	Med

Name	Status	Informat
Storage		
Cluster Disk 3	Online	
Other Resources		
Test	Online	
Server Name		
Name: cdcsql2014lsn1	Online	
Roles		
IBM InfoSphere Data Replication (Microsoft SQL Server) [SQL2...	Offline	

Right click the new resource added to the availability group role and select Properties. Navigate to the Dependencies tab and add the rest of the resources to the list of dependencies. This will prevent the IIDR resource from being started prior to having the rest of the resources initialized and started after a failover.

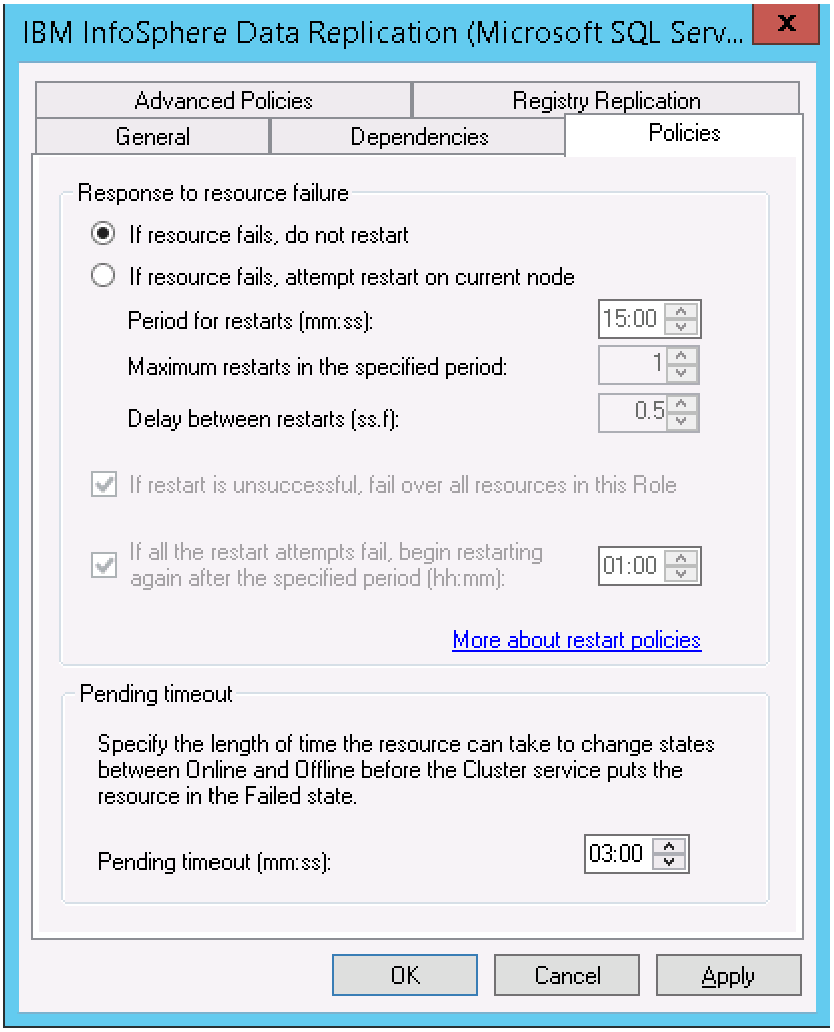


Specify the resources that must be brought online before this resource can be brought online:

AND/OR	Resource
	Cluster Disk 3
AND	Test
AND	Test_9.23.20.61
AND	Test_cdcsql2014lsn1
Click here to add a dependency	

Cluster Disk 3 AND Test AND Test_9.23.20.61 AND Test_cdcsql2014lsn1

Navigate to the Policies tab and choose the policy that you would want to be applied when the service fails or is manually stopped. In most cases, users may not want to fail over the entire availability group in case IIDR fails or is manually stopped. If this is the case, select “If Resource Fails, do no restart”. For more information about resource properties, consult Microsoft documentation. Click OK.



IBM InfoSphere Data Replication (Microsoft SQL Serv...

Response to resource failure

● If resource fails, do not restart

○ If resource fails, attempt restart on current node

Period for restarts (mm:ss): 15:00

Maximum restarts in the specified period: 1

Delay between restarts (ss.f): 0.5

☑ If restart is unsuccessful, fail over all resources in this Role

☑ If all the restart attempts fail, begin restarting again after the specified period (hh:mm): 01:00

More about restart policies

Pending timeout

Specify the length of time the resource can take to change states between Online and Offline before the Cluster service puts the resource in the Failed state.

Pending timeout (mm:ss): 03:00

Right click the service and select “Bring Online”

10 Configure the new Datastore in Management Console

When creating a new Datastore in Management Console use the AlwaysOn availability group listener name as the host name for the new Datastore. In our example, use cdcsql2014lsn1 as the host name for the new Datastore. The rest of the configuration is similar to defining a new Datastore. For more information, please refer to IIDR documentation on configuring a new Datastore.

11 Test the deployment

In order to test the deployment one would need to:

- Create a test source table in the database
- Create a test subscription in Management Console and create a table mapping between the newly created source test table and a target table
- Insert a couple of rows in the source test table
- Start mirroring
- Check that the table has been successfully refreshed and is mirroring
- Insert a few more rows. Test that replication has been successful
- Start Windows Failover Cluster Manager and navigate to the AlwaysOn availability group role
- Right click on the role and select Move -> Select Node. Select the Synchronous replica node (CDCDEV18)
- All resources will fail over and IIDR will be started
- The subscriptions will not be automatically started. The user will have to manually start the subscriptions or use persistent subscriptions that permits restarting subscriptions after a recoverable error
- Start the subscription. Insert a few more rows. Check that data has been replicated to the target

12 Persistent subscriptions

Persistent subscriptions permit subscription restart after a recoverable error. This IIDR feature is used by default in AlwaysOn deployments in order to automatically restart the subscriptions after a failover. In order to disable persistent subscriptions open IBM Management Console, connect to the datastore, right click on the subscription, select properties, then Advanced Settings and uncheck the "Mark Subscription as persistent" checkbox.

12.1 Considerations around automatic subscription restart for persistent subscriptions

- Persistent subscriptions functionality is initiated by the source engine. Not all sources support persistent subscriptions

- Persistent subscriptions attempt to restart a subscription at fixed intervals. By default, this fixed interval is set to 5 minutes. If you wish to change the automatic restart interval you need to change the system property mirror_auto_restart_interval_minutes, e.g. mirror_auto_restart_interval_minutes = 3

- Persistent subscriptions only try to restart a subscription if it was stopped due to a recoverable error

13 Considerations when transitioning the database to an AlwaysOn availability group

An existing IIDR deployment cannot be upgraded to support AlwaysOn. A new IIDR deployment is needed in order to support AlwaysOn. One needs to follow the instructions in this document in order to deploy a new IIDR installation that would support AlwaysOn.

If you currently have IIDR deployed and your database is not AlwaysOn, the following steps need to be executed in order to add the database to an AlwaysOn availability group and to deploy IIDR that supports such a database deployment.

- Establish a quiesce point for the database. This means that the database cannot be accessible to either DML or DDL transactions. If a quiesce point cannot be established a refresh of all tables will be needed
- Replicate all the data up to the quiesce point – i.e. to the head of log
- Upgrade your existing IIDR instance to the latest version that supports AlwaysOn. Start the instance. This will upgrade the IIDR metadata to the latest product version
- Upgrade Management Console and Access Server to the latest version
- Add the database to an AlwaysOn availability group
- Deploy a new IIDR installation on the shared clustered disk and follow all the steps required for an AlwaysOn IIDR deployment
- Start both instances
- Use Management Console to either export/import subscriptions or promote subscriptions from the old non AlwaysOn deployment to the AlwaysOn deployment
- If a quiesce point has been established at the beginning and all data has been replicated to the head of log prior to upgrade, you can Mark Table Capture Point for Mirroring for all tables in all subscriptions in the AlwaysOn instance. This action will tell IIDR to start reading from the current head of log. If a quiesce point could not be established at the beginning, then all tables have to be refreshed
- A good rule of thumb is test and document the specific steps needed for your solution in a test environment prior to making changes to production environment

14 Considerations when transitioning the database out of an AlwaysOn availability group

One may decide to continue to use IIDR replication and no longer have their database as part of an availability group. In order to achieve this one needs to deploy a new IIDR instance configured with the non AlwaysOn database. Here are the steps that need to be executed when transitioning from a database that is part of an AlwaysOn availability group to the same database that is not part of an availability group.

14.1Using an existing IIDR installation on Clustered Disk

The following steps need to be executed when one wants to transition the database out of an availability group and change IIDR to use an installation residing on local disk.

- Establish a quiesce point for the database. This means that the database cannot be accessible to DML or DDL transactions. If a quiesce point cannot be established a refresh of all tables will be needed
- Replicate all the data up to the quiesce point – i.e. to the head of log Remove the database from the AlwaysOn availability group
- Create a new IIDR instance with the database that is not part of an availability group.
- You can decide you want to continue to use the remote distributor or define a local distributor for Microsoft Replication. Make any necessary adjustments at this time
- Start both instances
- Use Management Console to either export/import subscriptions or promote subscriptions from the old AlwaysOn instance to the non AlwaysOn instance
- Remove the IIDR service of the AlwaysOn instance from the availability group
- If a quiesce point has been established at the beginning and all data has been replicated to the head of log prior to upgrade, you can Mark Table Capture Point for Mirroring for all tables in all subscriptions in the AlwaysOn instance. This action will tell IIDR to start reading from the current head of log. If a quiesce point could not be established at the beginning, then all tables have to be refreshed
- A good rule of thumb is test and document the specific steps needed for your solution in a test environment prior to making changes to production environment

14.2Using a new IIDR installation on Local Disk

The following steps need to be executed when one wants to transition the database out of an availability group and move the CDC installation from the clustered shared disk to local disk.

- Establish a quiesce point for the database. This means that the database cannot be accessible to DML or DDL transactions. If a quiesce point cannot be established a refresh of all tables will be needed
- Replicate all the data up to the quiesce point – i.e. to the head of log Upgrade your existing AlwaysOn IIDR instance to the latest version of the product that you want to deploy on local drive. Start the AlwaysOn instance. This will upgrade the IIDR metadata to the latest product version
- Upgrade Management Console and Access Server to the latest version

- Remove the database from an AlwaysOn availability group
- Deploy a new IIDR installation on the local disk
- You can decide you want to continue to use the remote distributor or define a local distributor for Microsoft Replication. Make any necessary adjustments at this time
- Start both instances
- Use Management Console to either export/import subscriptions or promote subscriptions from the old AlwaysOn deployment to the new non AlwaysOn deployment
- If a quiesce point has been established at the beginning and all data has been replicated to the head of log prior to upgrade, you can Mark Table Capture Point for Mirroring for all tables in all subscriptions in the AlwaysOn instance. This action will tell IIDR to start reading from the current head of log. If a quiesce point could not be established at the beginning, then all tables have to be refreshed.
- A good rule of thumb is test and document the specific steps needed for your solution in a test environment prior to making changes to production environment.

15 Resuming replication after automatic or manual failover to a synchronous replica

As you probably have already tested, after a failover to a new PRIMARY replica, IIDR fails together with the availability group to the new node. Assuming the subscriptions have been marked persistent, the replication will also automatically restart and continue from where it left over on the previous PRIMARY replica. Subscriptions are marked as persistent by default, when IIDR is configured with a database contained in an AlwaysOn availability group.

16 Resuming replication after forced failover to an asynchronous replica

IIDR only supports replication from a synchronous replica. IIDR does not support failing over to an asynchronous replica. In addition, if you failover back to the previous PRIMARY replica, IIDR will not be able to continue mirroring, because IIDR does not support failover from an ASYNCHRONOUS replica.

In order to recover from such a situation, you will need to either define a new set of synchronous replicas or fail over to the original PRIMARY replica, ensure that Microsoft Replication continues to be properly configured for the environment and initiate Refresh/Mirror for all tables.