
Installing and Configuring a SQL Server 2012 Failover Cluster

Edwin M Sarmiento

Applies to:

- SQL Server 2012
- SQL Server 2014

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.

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

You may use this document for your internal, reference purposes. You may **NOT** copy, alter or reuse this document or any portions of it for other than the original purpose without the written consent of the author.

© 2014 Copyright. All rights reserved.

Table of Contents

Introduction	1
Assumptions.....	1
Cluster Configuration Details.....	2
Connecting to the iSCSI Storage	3
Initializing and Formatting the Disks.....	6
Installing the .NET Framework 3.5	11
Windows Failover Cluster Feature Installation	12
Windows Server Failover Clustering Configuration	14
SQL Server 2012 Failover Cluster Instance Installation - Install Primary (Initial) Cluster Node	20
SQL Server 2012 Failover Cluster Instance Installation - Install Secondary (Failover) Cluster Node	33
About The Author	36

Introduction

The goal of this documentation is to provide a step-by-step procedure in configuring a 2-node Windows Server 2012 Failover Cluster that will host a SQL Server 2012 Failover Clustered Instance.

Assumptions

When using this guide, a few assumptions have been made

- Windows Server 2012 is installed on each server that you will cluster and join to the domain
- The disks (LUNs) that you want to use in the cluster are exposed to the servers that you will cluster and are configured appropriately as per [StarWind's High Availability Best Practices documentation](#)
- You have decided which quorum model will be used by the failover cluster. This document will use the disk witness as the quorum model.

Cluster Configuration Details

Prior to building your cluster, it is important to have all the details in place to make sure that installation and configuration will go smoothly. The following details will be used to build and configure a SQL Server 2012 Failover Clustered Instance.

Active Directory Domain Name: TESTDOMAIN.local

StarWind Server iSCSI SAN IP address: 10.0.0.200

Cluster Nodes: WS-CLUSTER3 & WS-CLUSTER4

Cluster Disks: Q:\, E:\, F:\ & H:\

Windows Server Failover Cluster Name and IP Address: WINCLUSTER2 & 172.16.0.211

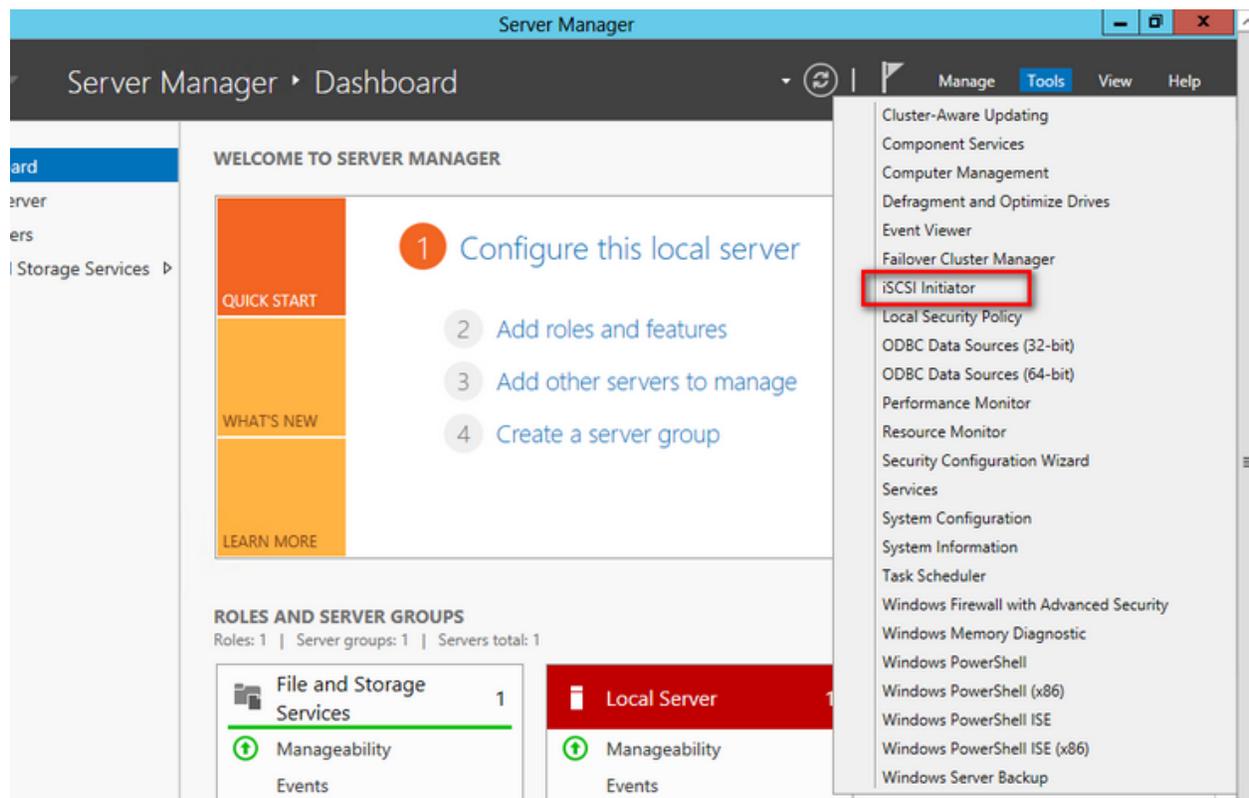
SQL Server Failover Cluster Name and IP Address: SQLCLUSTER2 & 172.16.0.212

SQL Server Service Account: TESTDOMAIN\sqlservice

Connecting to the iSCSI Storage

Windows Server 2012 comes with iSCSI Initiator software that enables connection of a Windows host to an external iSCSI storage array like StarWind Software's iSCSI SAN using network adapters. You can launch the tool from the Server Manager dashboard, under Tools and select iSCSI Initiator.

NOTE: These steps have to be performed on both of the servers that will act as nodes in your failover cluster.

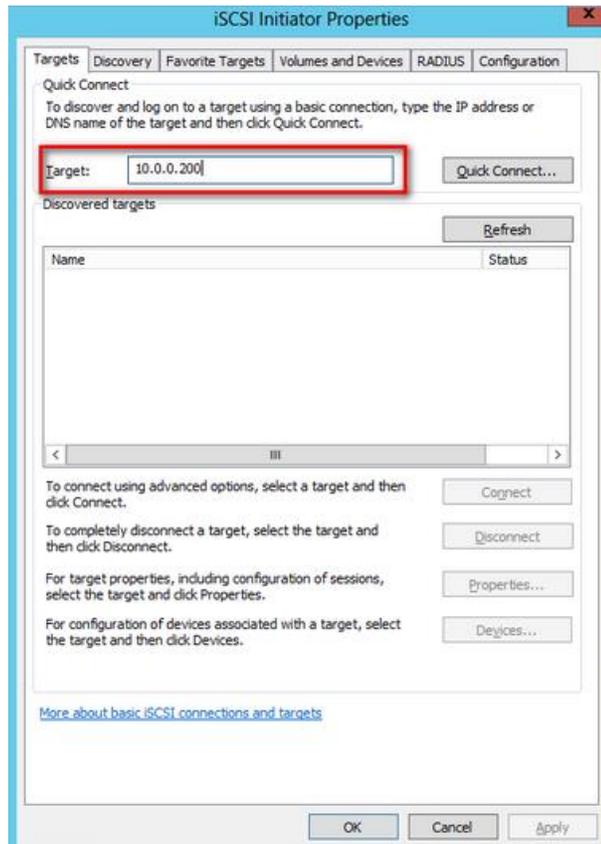


You will get a message saying that the Microsoft iSCSI service is not running. Simply click **Yes** to continue.

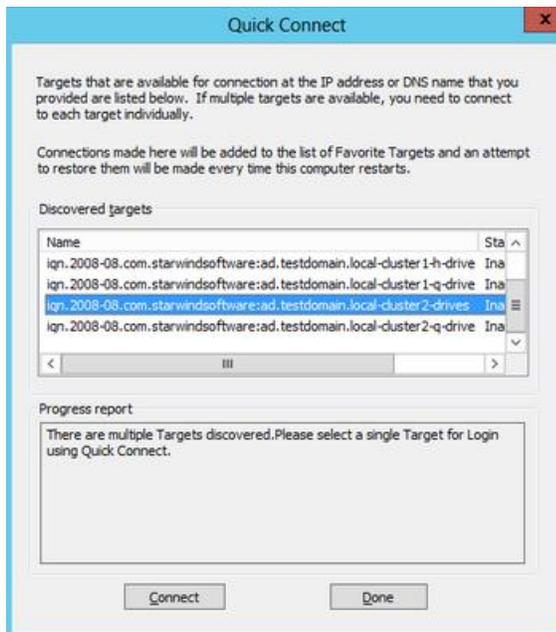


To connect to the iSCSI target:

1. In the **iSCSI Initiator Properties** page, under the **Targets** tab, enter the IP address of the StarWind server iSCSI SAN on the **Target** field. Click **Quick Connect**. You should see a list of the iSCSI Targets that have been defined on your iSCSI SAN.



2. In the **Quick Connect** dialog box, select the target that you will use and click **Connect**. Click **Done** to go back to the **iSCSI Initiator Properties** page.



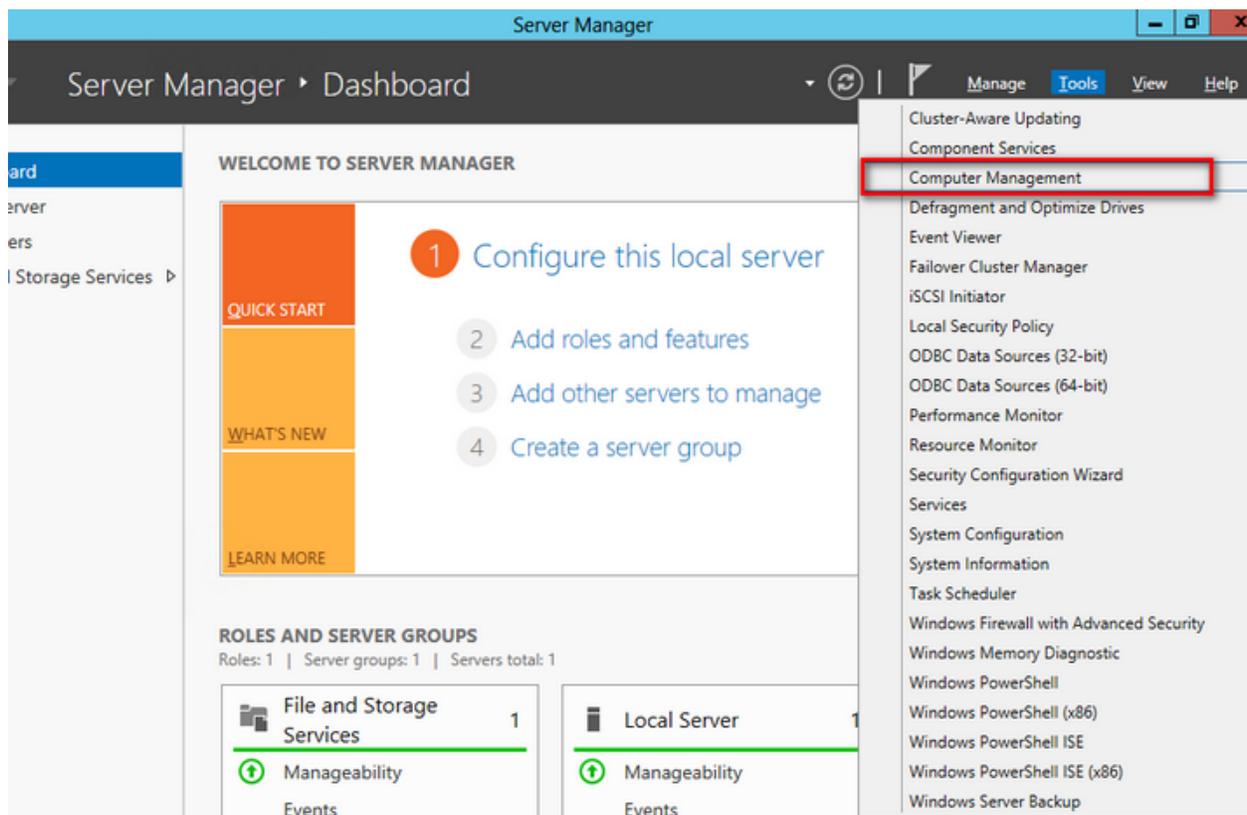
3. Click **OK** to close the **iSCSI Initiator Properties** page.

Once the targets have been defined using the iSCSI Initiator tool, you can now bring the disks online, initialize them, and create new volumes using the **Disk Management** console.

Initializing and Formatting the Disks

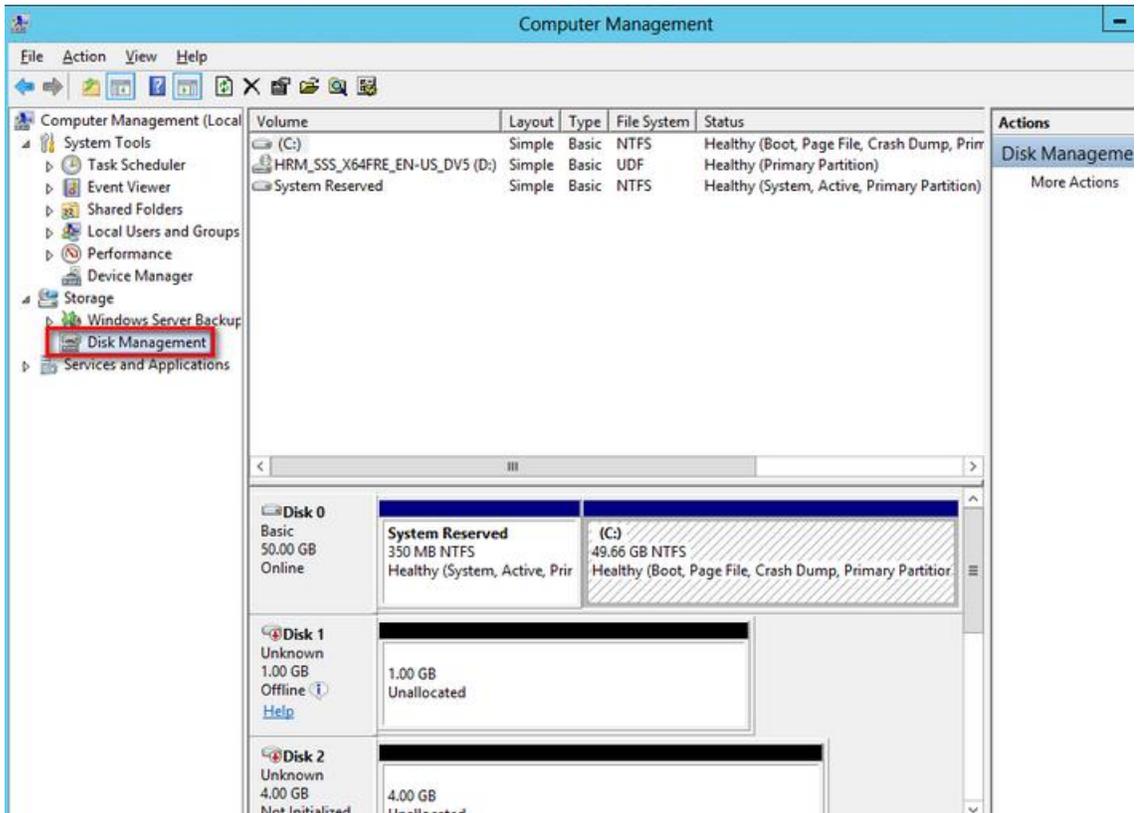
After configuring the servers to connect to the iSCSI target, we can proceed to initialize and format the disks. We will use the **Disk Management** console to perform this task. You can launch the tool from the **Server Manager** dashboard, under **Tools** and select **Computer Management**.

NOTE: These steps have to be performed on both of the servers that will act as nodes in your failover cluster.

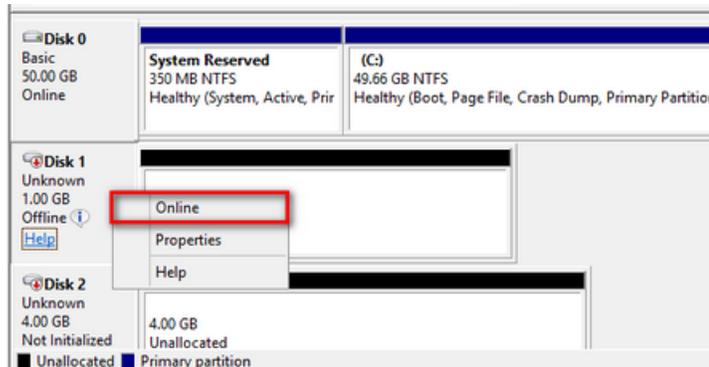


To initialize and format the drives:

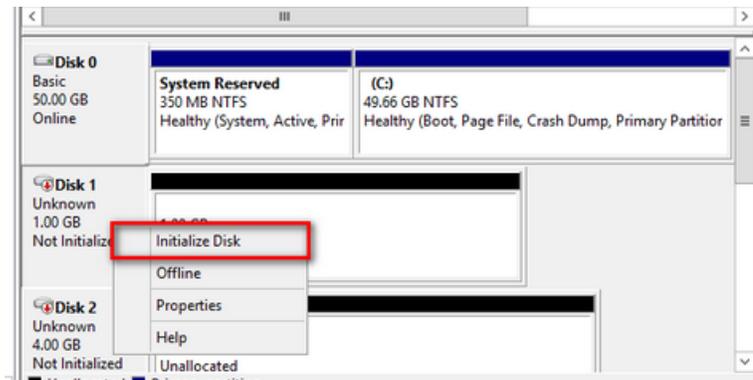
1. From within **Computer Management**, expand the **Storage** section and select **Disk Management**.



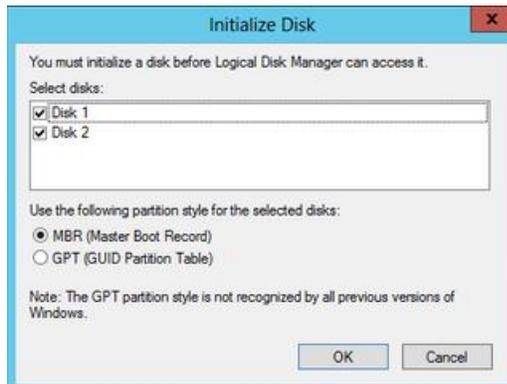
- Right-click any of the disks that you want to configure and select **Online**. Once the disk is brought online, it is now marked as **Not Initialized**.



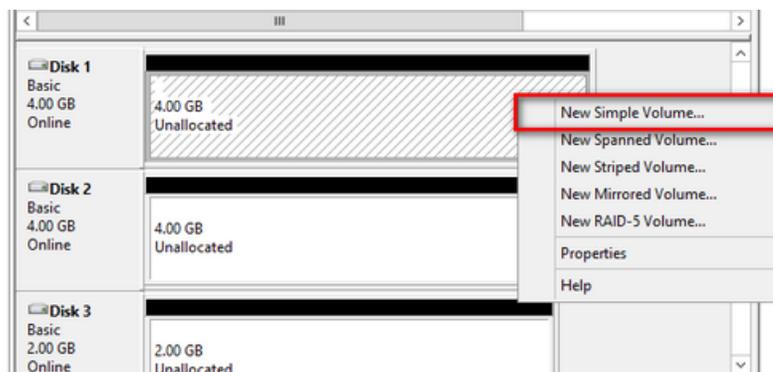
- To initialize, right-click on the disk and select **Initialize Disk**. The Initialize Disk dialog box will appear.



4. In the **Initialize Disk** dialog box, make sure that the correct disk is selected for initialization and then choose whether to initialize the disk using the MBR or GPT partition styles. The most common partition style for disks used in SQL Server instances is **MBR**, which is selected by default. Click **OK** to select the MBR partition style.



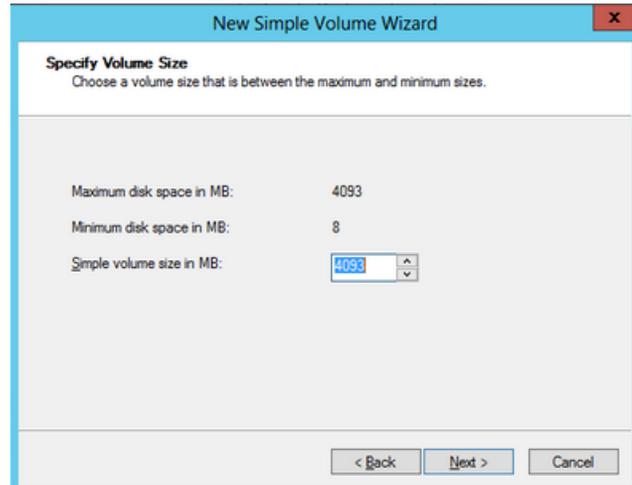
5. To create a disk partition, right-click on the unallocated space and select **New Simple Volume**.



6. In the **Welcome to the New Simple Volume Wizard** dialog box, click **Next**.



7. In the **Specify Volume Size** dialog box, enter the volume size and click **Next**.



8. In the **Assign Drive Letter or Path** dialog box, specify the drive letter you would like to use and click **Next**.



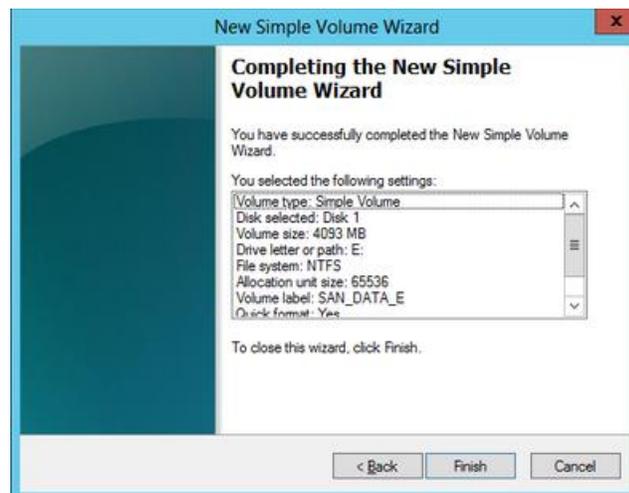
9. In the **Format Partition** dialog box,

- Make sure that the file system selected is **NTFS**.
- To follow Microsoft best practices on allocation unit size, select **64K**.
- In the **Volume label**: text box, enter the appropriate name.

Click **Next**



10. In the **Completing the New Simple Volume Wizard** dialog box, review the settings you have made and click **Finish**.



11. Repeat these steps on all of the disks that you want to configure as part of your Windows Server Failover Cluster.

Installing the .NET Framework 3.5

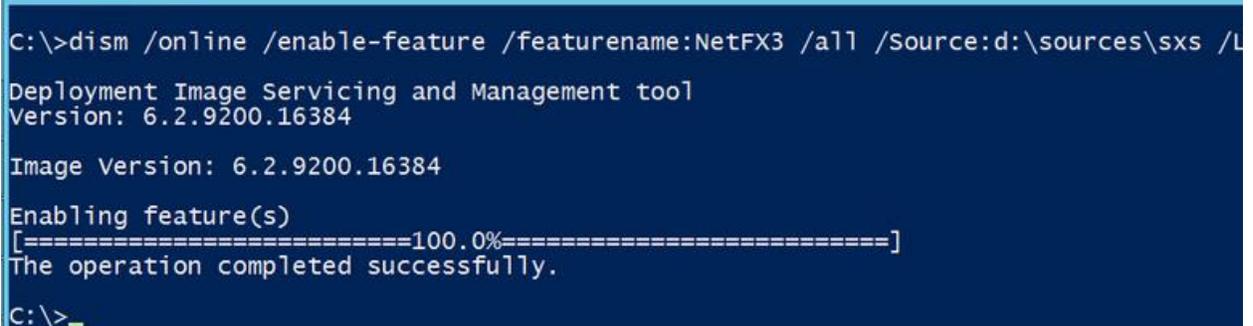
SQL Server 2012 requires both the Microsoft .NET Framework 3.5.1 and 4.0 as part of its prerequisite software. The .NET Framework 3.5.1 package is already a part of the Windows Server 2012 installation media while the .NET Framework 4.0 is included in the SQL Server 2012 installation media. However, there are issues when installing the .NET Framework 3.5.1 on Windows Server 2012 machines when using **the Server Manager** dashboard. [Microsoft KB article 2734872](#) describes the issues you will encounter when installing the .NET Framework 3.5.1 on a Windows 8 or Windows Server 2012 machine.

To install the Microsoft .NET Framework 3.5.1 on a Windows Server 2012, we will be using the [Deployment Image Servicing and Management \(DISM\) Command-Line](#) tool.

NOTE: These steps have to be performed on both of the servers that will act as nodes in your failover cluster.

To install the Microsoft .NET Framework 3.5.1, open a command-line and type the command below.

```
dism /online /enable-feature /featurename:NetFX3 /all /Source:d:\sources\sxs /LimitAccess
```



```
C:\>dism /online /enable-feature /featurename:NetFX3 /all /Source:d:\sources\sxs /L
Deployment Image Servicing and Management tool
Version: 6.2.9200.16384

Image Version: 6.2.9200.16384

Enabling feature(s)
[=====100.0%=====]
The operation completed successfully.

C:\>
```

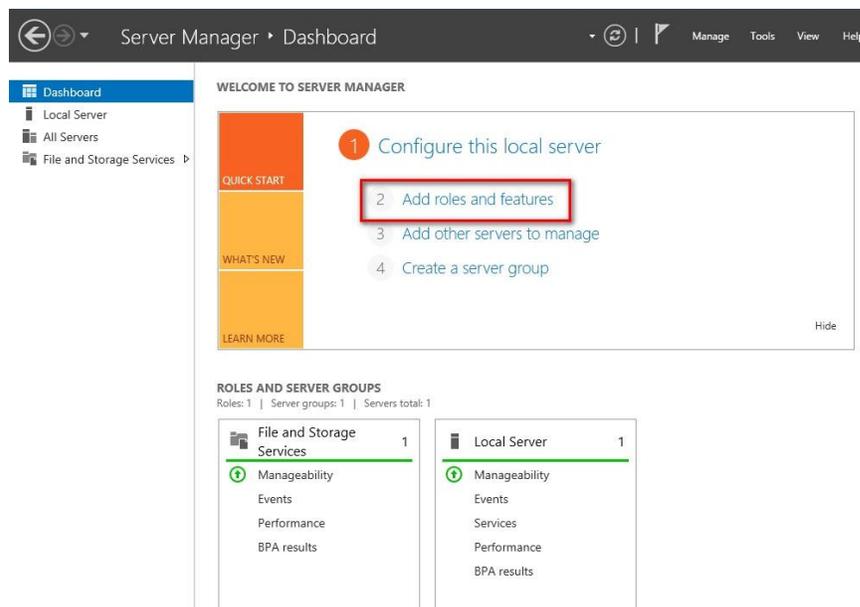
Windows Failover Cluster Feature Installation

Windows Server Failover Clustering feature provides high availability and scalability in many server workloads. SQL Server 2012 takes advantage of this feature and its capabilities to support a failover clustered instance and the new SQL Server 2012 AlwaysOn Availability Group feature. Discussion of the new SQL Server 2012 AlwaysOn Availability Group feature is beyond the scope of this document. We will only be focusing on SQL Server failover clustered instances.

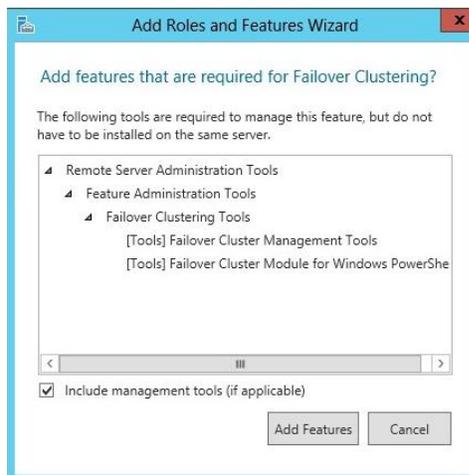
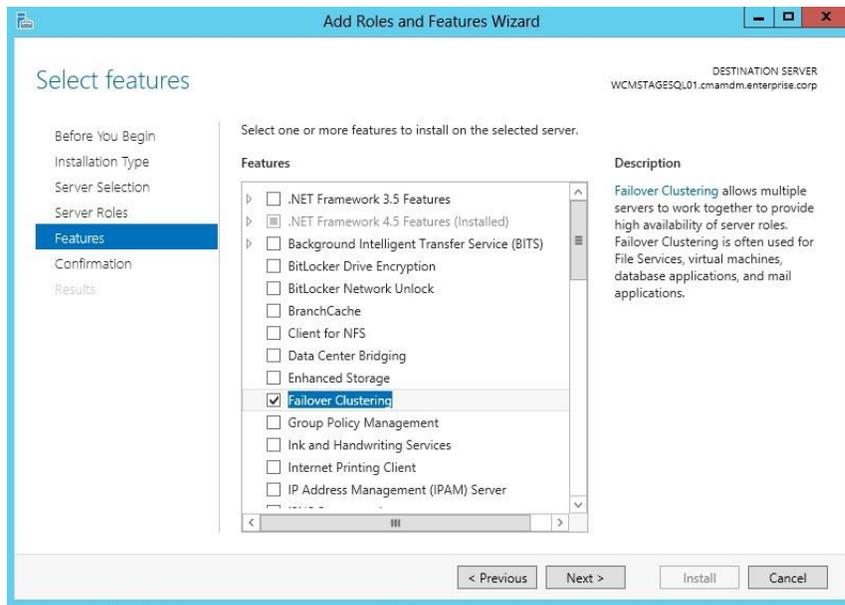
NOTE: These steps have to be performed on both of the servers that will act as nodes in your failover cluster.

To add the Failover Clustering feature:

1. Open the **Server Manager** dashboard and select **Add roles and features**. This will launch the **Add Roles Features Wizard**



2. Click **Next** until you reach the **Select Features** dialog box. Select the **Failover Clustering** checkbox. When prompted with the **Add features that are required for Failover Clustering** dialog box, click **Add Features**. Click **Next**.



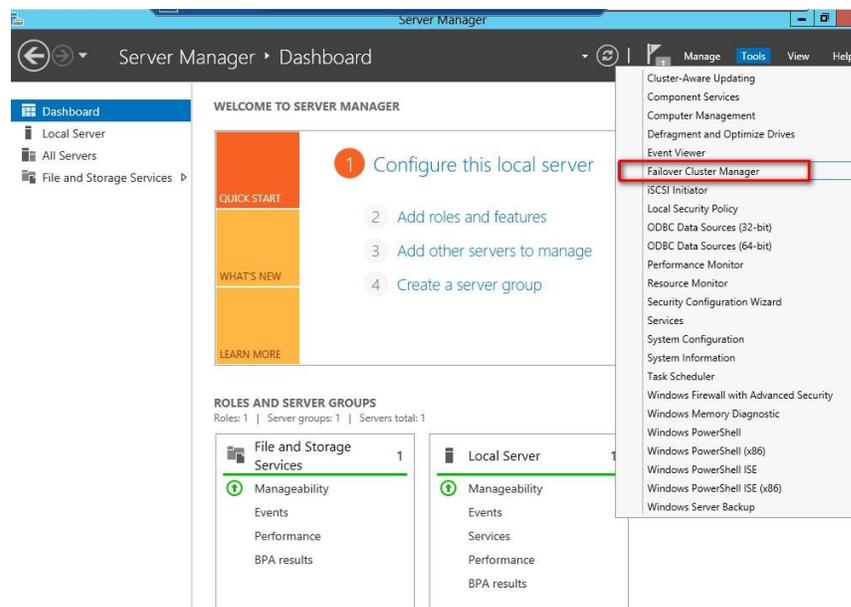
3. Click **Install** to install the Failover Clustering feature.

Windows Server Failover Clustering Configuration

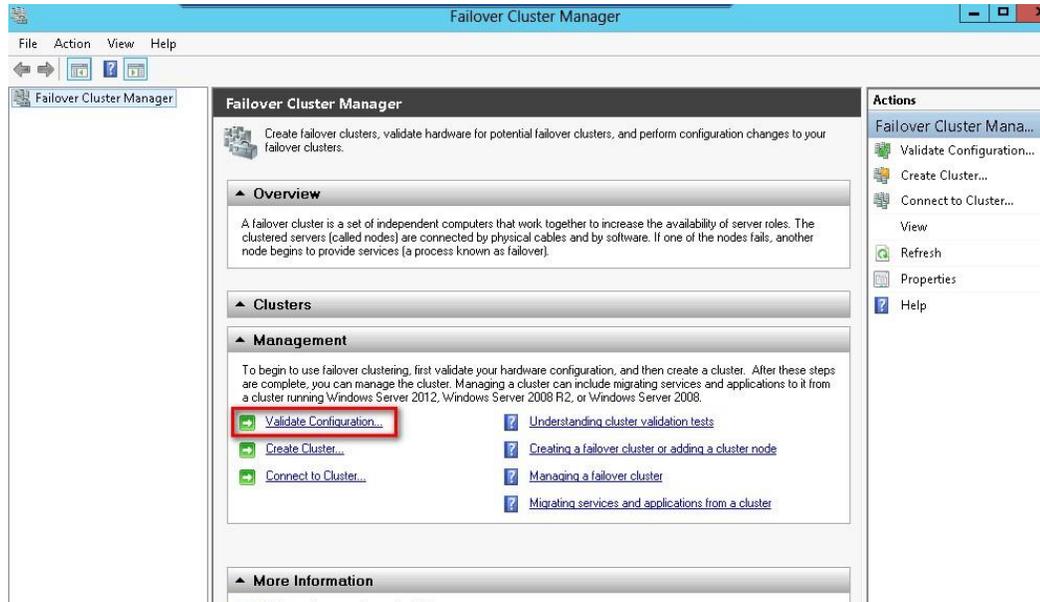
Once the Windows Server Failover Clustering Feature has been installed, we can proceed with configuring a Windows Server Failover Cluster. We will start with running the Failover Cluster Validation Wizard to make sure that our cluster configuration will be officially supported by Microsoft.

To configure Windows Failover Clustering,

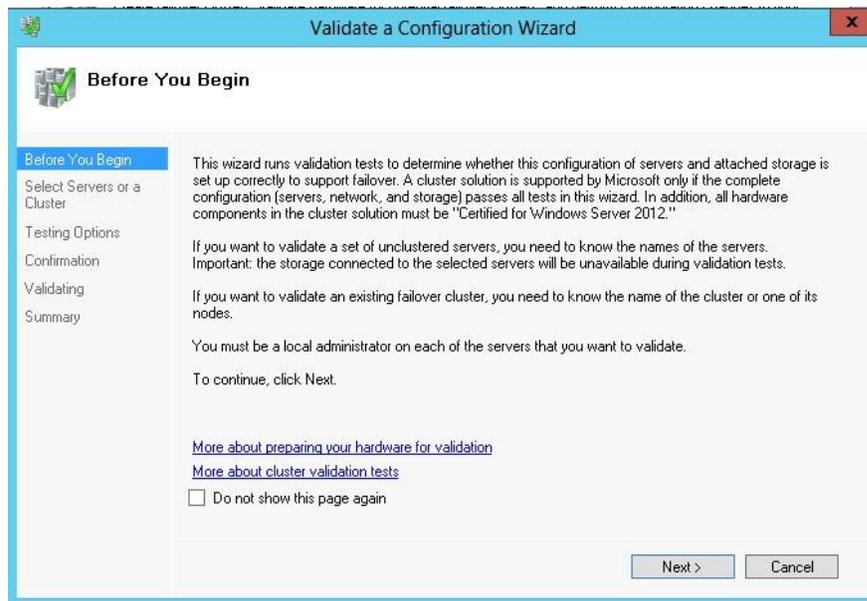
1. Launch **Failover Cluster Manager** from within the **Server Manager** console.



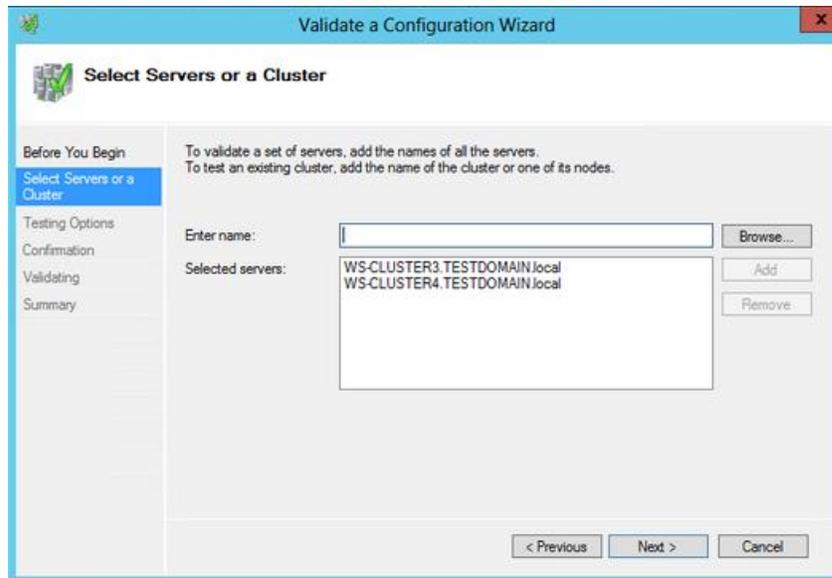
2. Within **Failover Cluster Manager**, click the **Validate Configuration...** link.



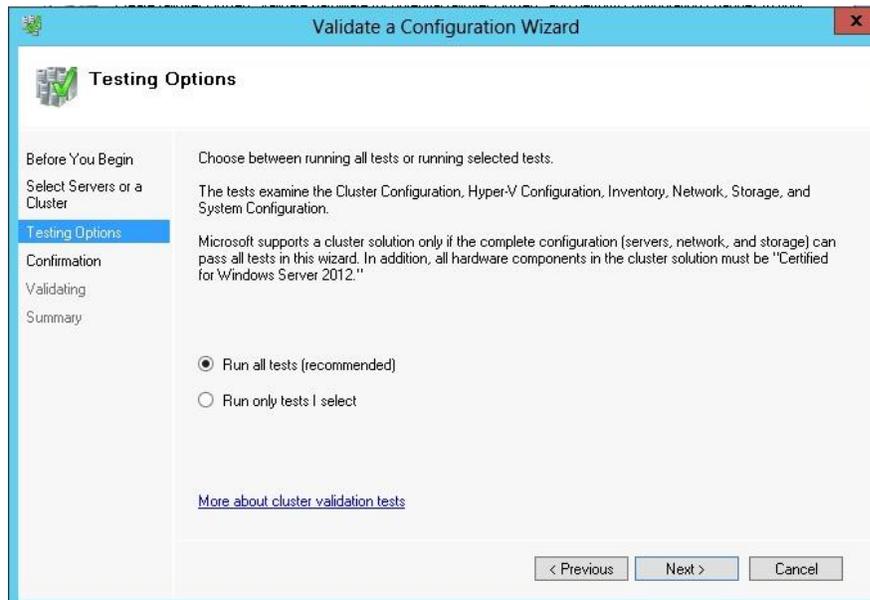
3. In the **Validate a Configuration Wizard** dialog box, click **Next**.



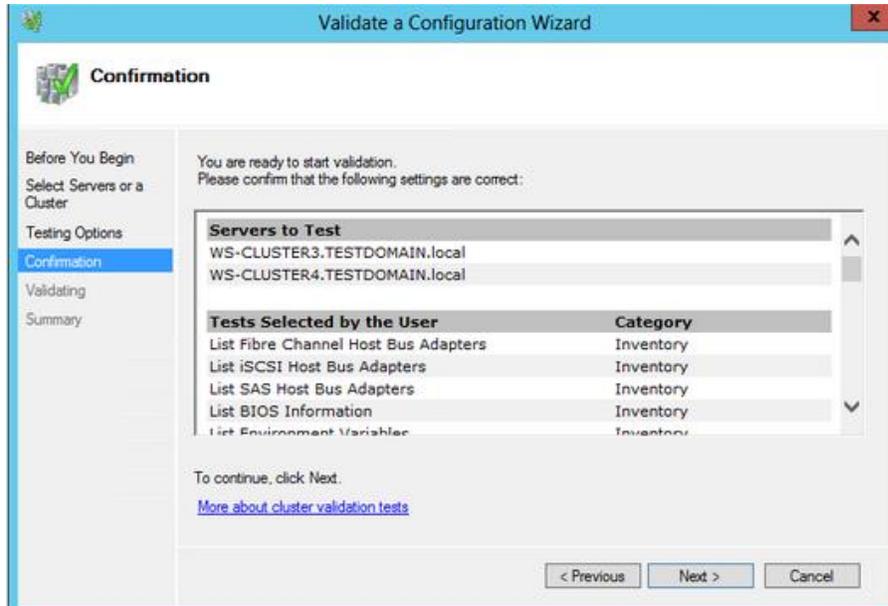
4. In the **Select Servers or a Cluster** dialog box, add the following server names – **WS-CLUSTER3** and **WS-CLUSTER4**. Click **Next**.



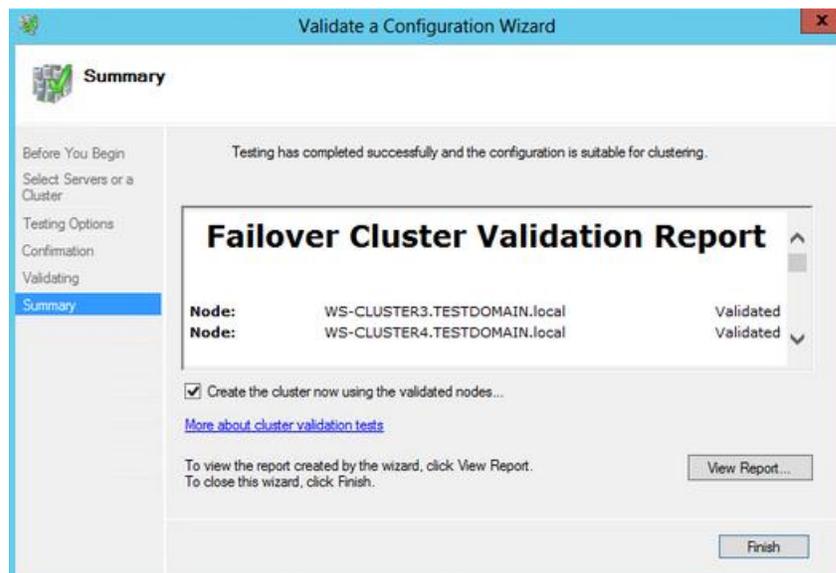
5. In the **Testing Options** dialog box, make sure that the option **Run all tests (recommended)** is selected. Click **Next**.



6. In the **Confirmation** dialog box, click **Next**.

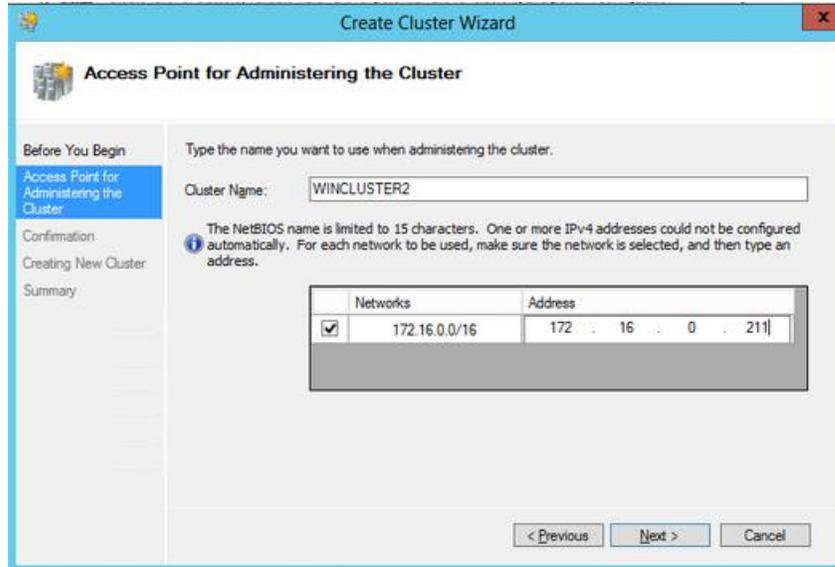


7. In the **Summary** dialog box, click **Finish** to create the Windows Server Failover Cluster.

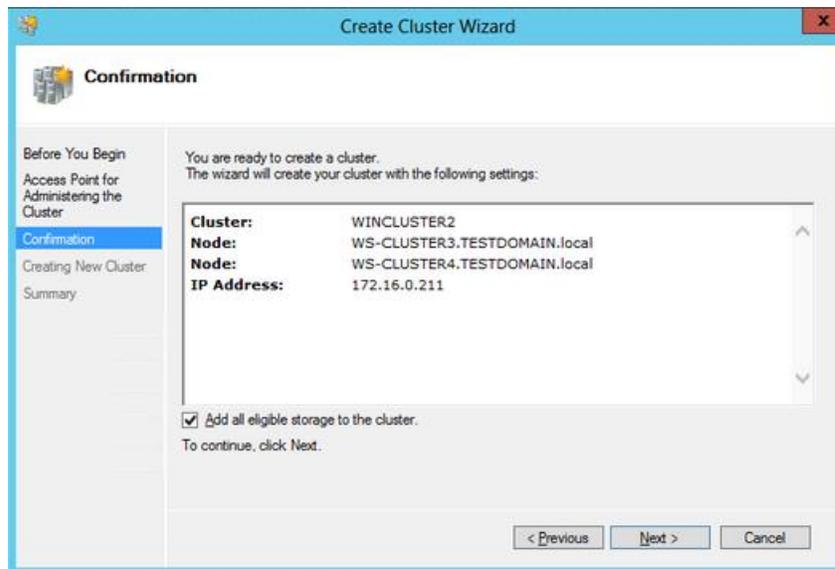


8. In the **Access Point for Administering the Cluster** dialog box, enter the following details

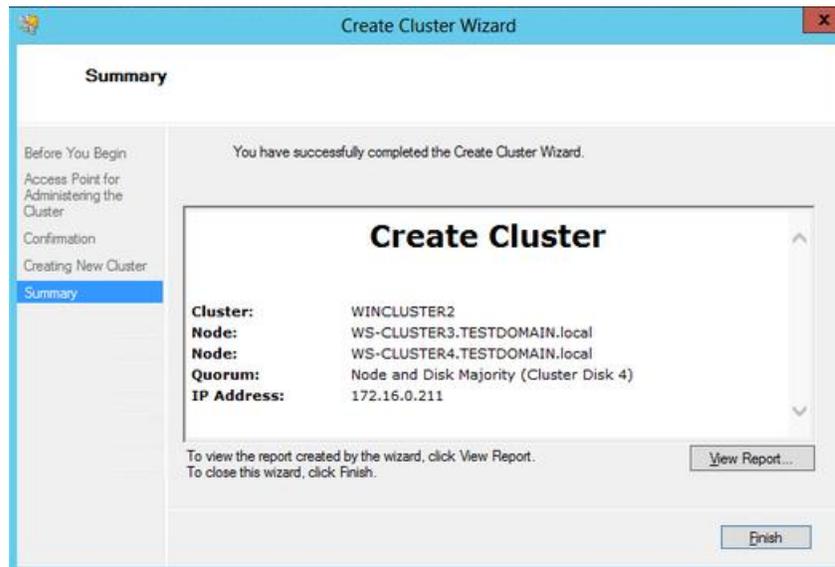
- **Cluster Name:** WINCLUSTER2
- **Address:** 172.16.0.211



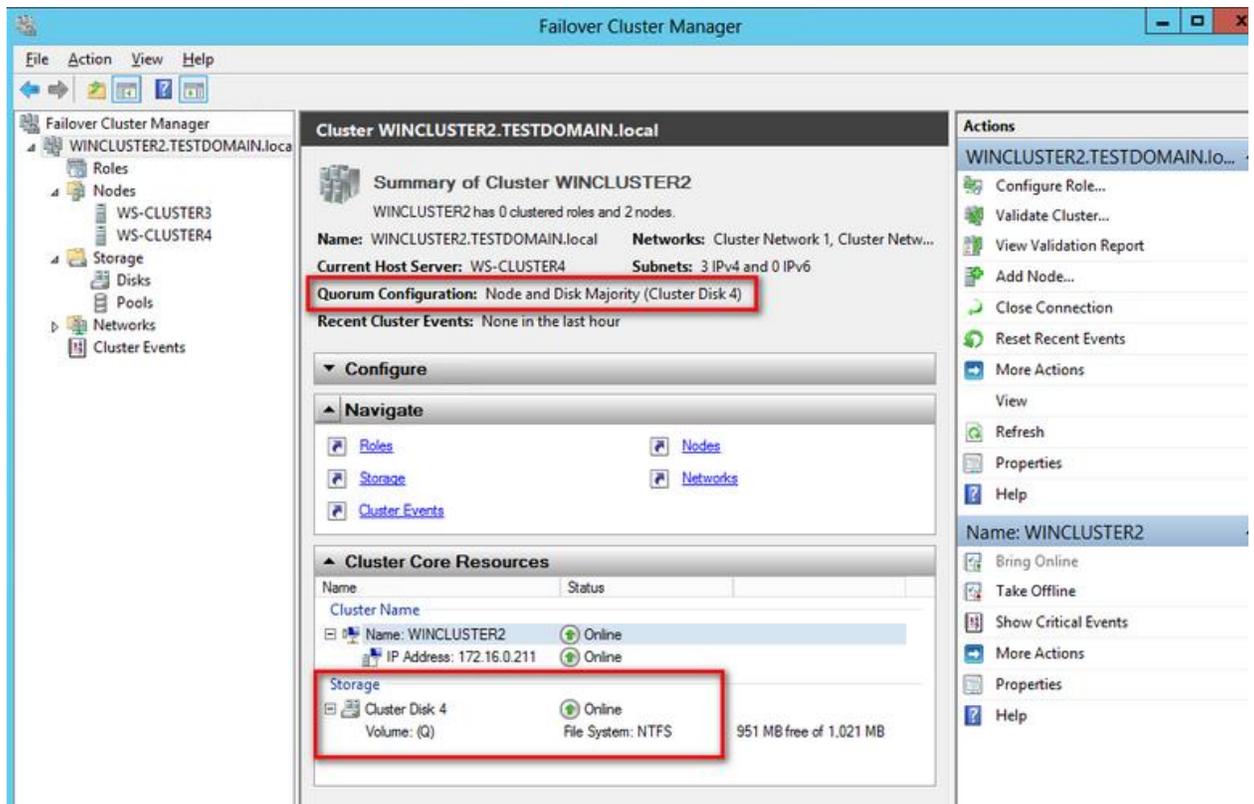
9. In the **Confirmation** dialog box, click **Next**. This will create the Windows Server Failover Cluster using the servers as nodes of the cluster, add DNS and Active Directory entries for the cluster hostname.



10. In the **Summary** dialog box, verify that the configuration is successful.



11. Verify that the cluster quorum configuration is using **Node and Disk Majority**, using the appropriate drive that you configured as the quorum disk.



SQL Server 2012 Failover Cluster Instance Installation - Install Primary (Initial) Cluster Node

Installing a SQL Server 2012 Failover Clustered Instance requires that you have the appropriate permissions in your Active Directory domain and that you have local Administrator rights to the machines that will act as nodes in your failover cluster. Configuring the appropriate permissions in Active Directory is beyond the scope of this document. For more information, refer to the document [Failover Cluster Step-by-Step Guide: Configuring Accounts in Active Directory](#).

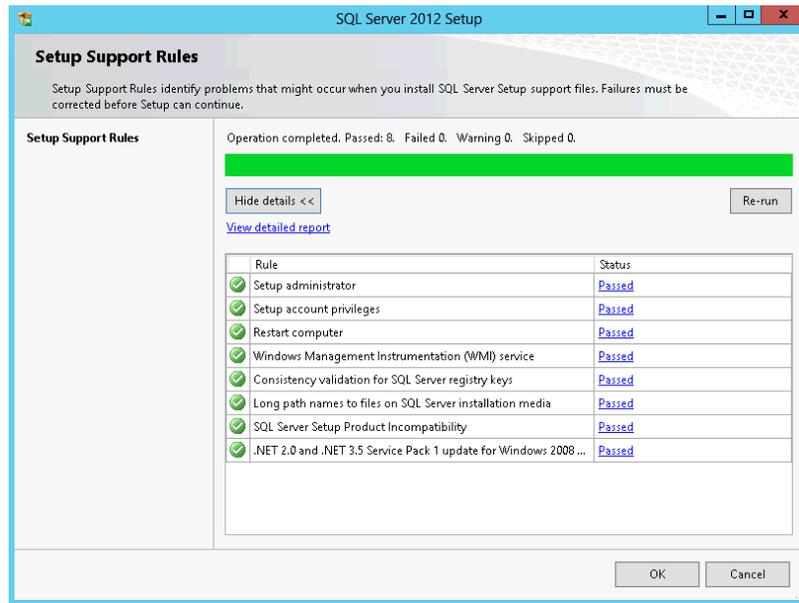
There are two options to install SQL Server 2012 Failover Clustered Instance. The first one is by using the integrated failover cluster install with the Add Node option and the second one is the Advanced/Enterprise installation option. The process outlined below will take into account the first option and will be installing a DEFAULT instance.

To install a SQL Server 2012 failover clustered instance,

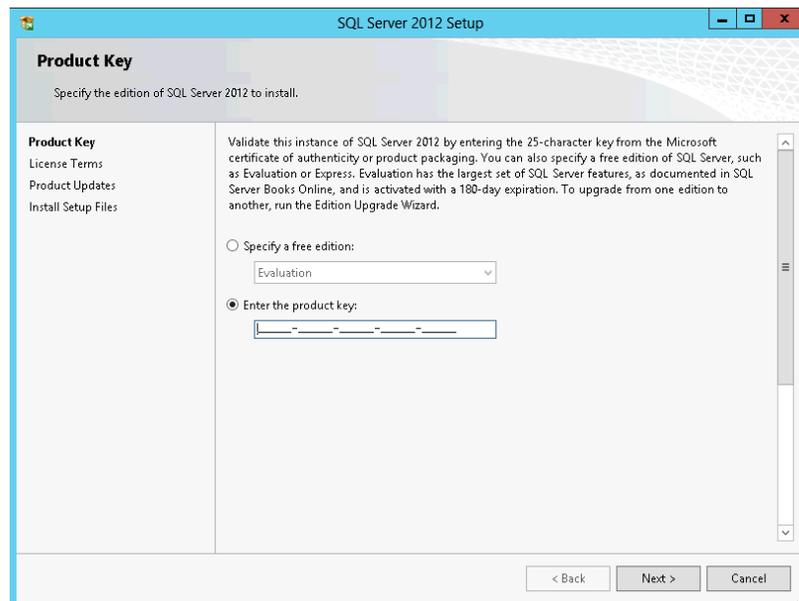
1. Run **setup.exe** from the installation media to launch **SQL Server Installation Center**. Click on the **Installation** link on the left-hand side.
2. Click on the **New SQL Server Failover Cluster Installation** link. This will run the SQL Server 2012 Setup wizard.



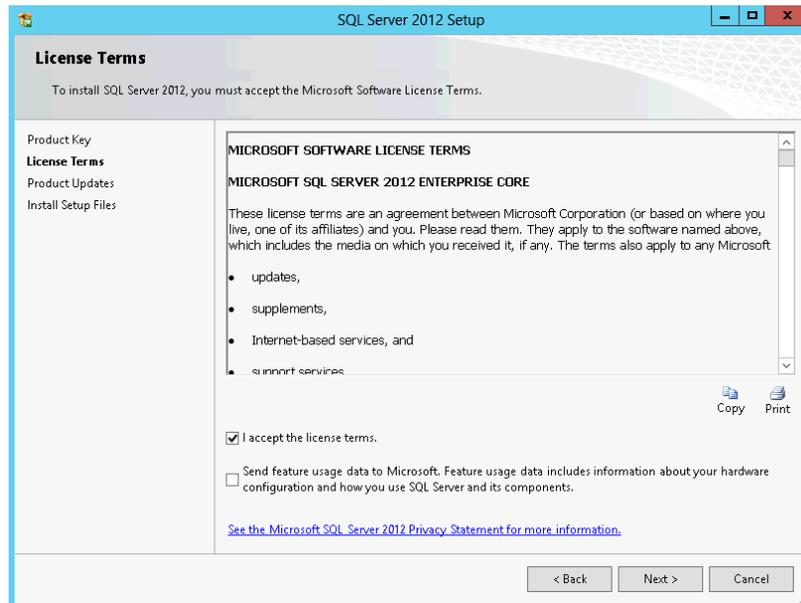
3. In the **Setup Support Rules** dialog box, validate that the checks return successful results and click **Next**.



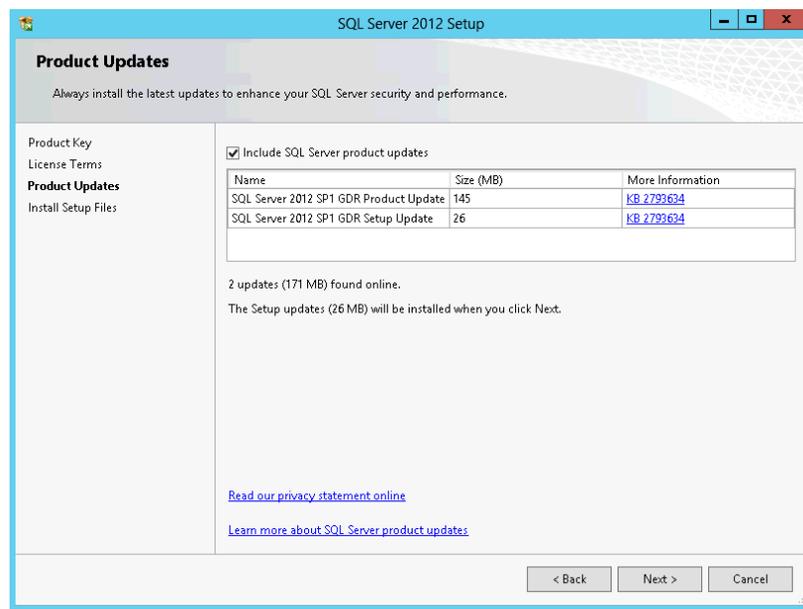
4. In the **Product Key** dialog box, enter the product key that came with your installation media and click **Next**.



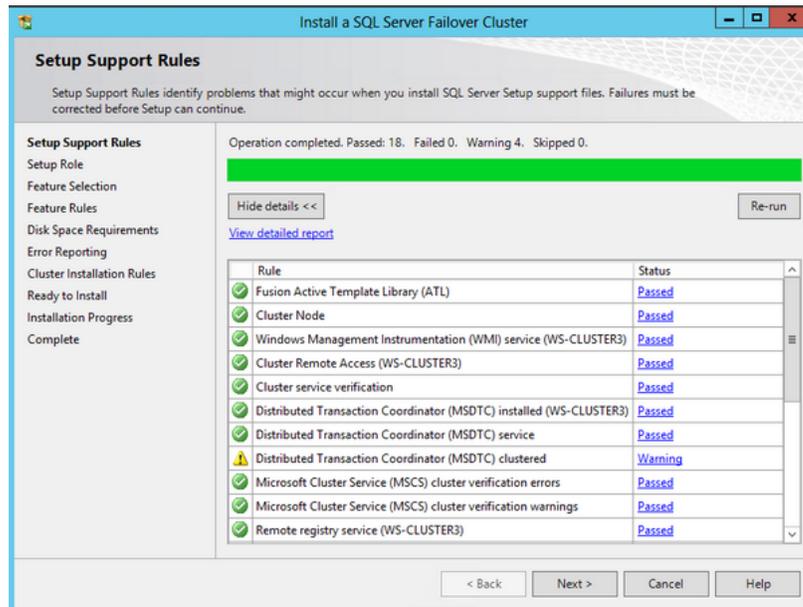
5. In the **License Terms** dialog box, click the I accept the license terms check box and click **Next**.



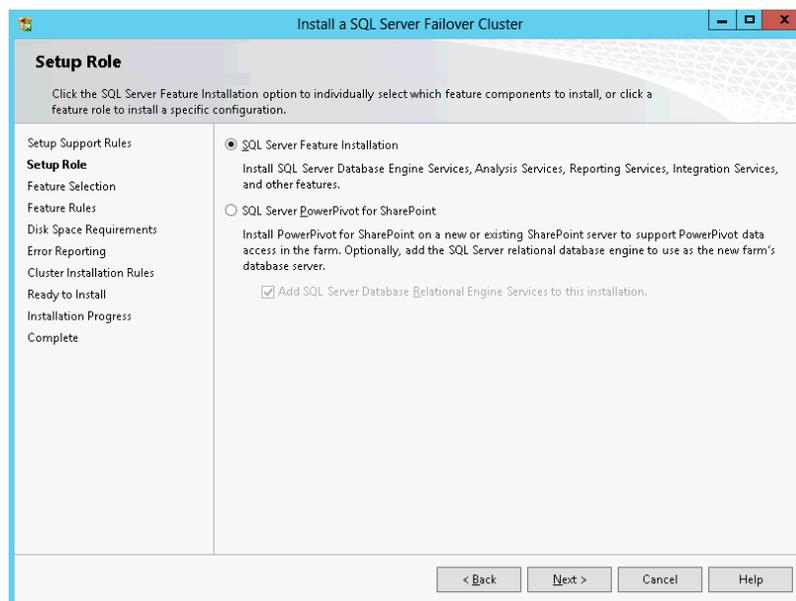
6. In the **Product Updates** dialog box, you have the option to include SQL Server product updates like service packs and cumulative updates in the installation process. If your servers are connected to the Internet, the installation media will connect to Windows Update and check for available SQL Server 2012 updates. You can opt to install the detected updates as part of the installation process. Click **Next**.



7. In the **Setup Support Rules** dialog box, validate that the checks return successful results. If the checks returned a few warnings, make sure you fix them before proceeding with the installation. Click **Next**.

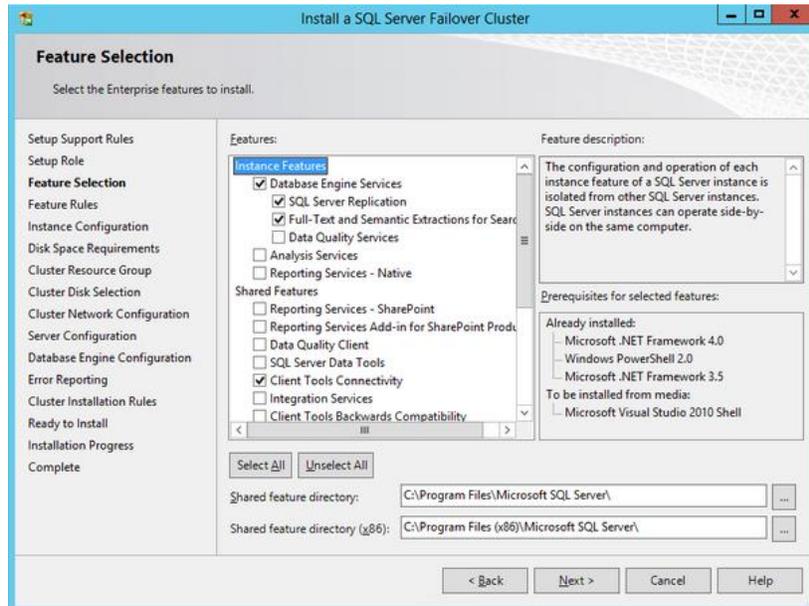


- In the **Setup Role** dialog box, select the **SQL Server Feature Installation** option and click **Next**.

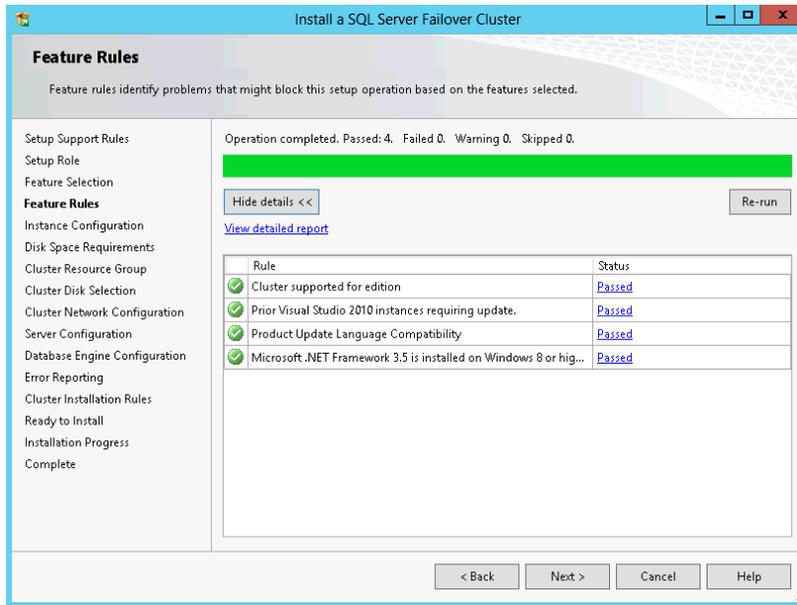


- In the **Feature Selection** dialog box, select the following components – **Database Engine Services**, **SQL Server Replication**, **Client Tools Connectivity** and **Management Tools**. Click **Next**.

NOTE: Data Quality Services is now a requirement when installing the Database Engine Services on a Failover Cluster when the installation media includes SQL Server 2012 Service Pack 1 as per Microsoft KB article 2674817.



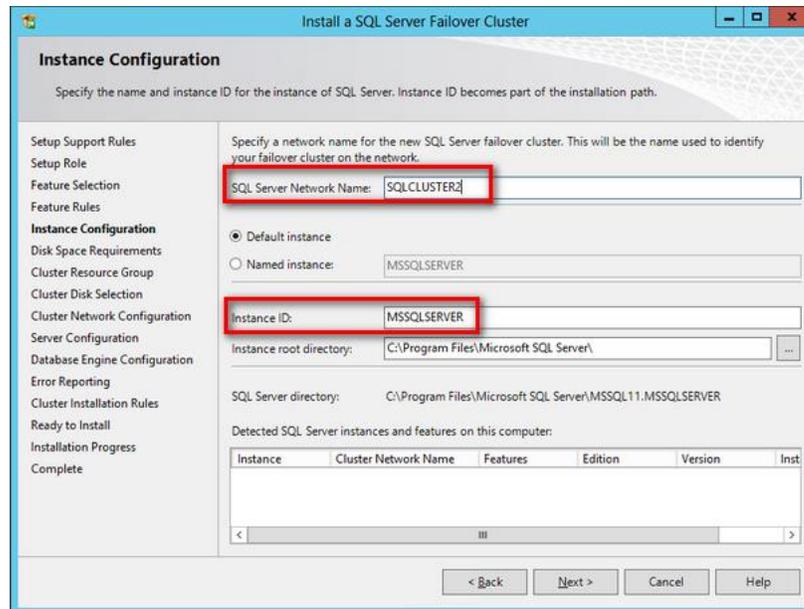
10. In the **Feature Rules** dialog box, verify that all the rules have passed. If the rules returned a few warnings, make sure you fix them before proceeding with the installation. Click **Next**.



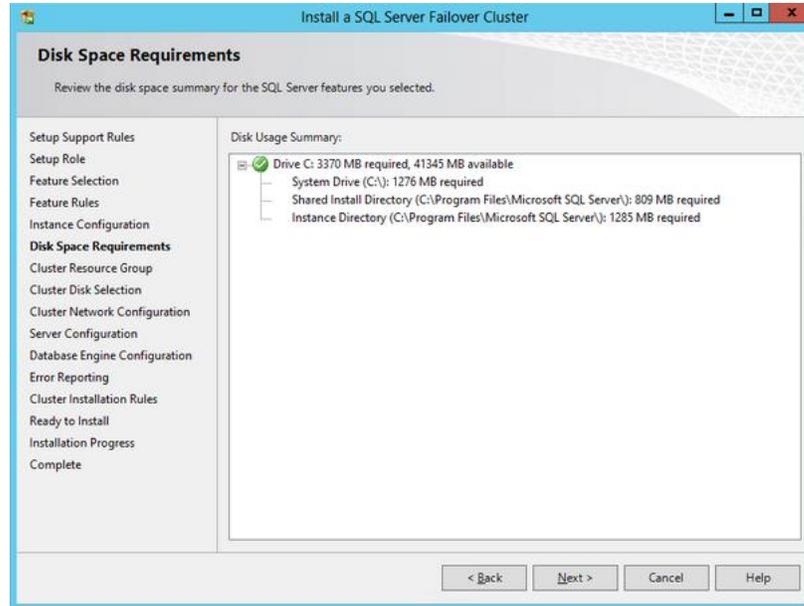
11. In the **Instance Configuration** dialog box, enter the following details:

- **SQL Server Network Name:** SQLCLUSTER2
- **Instance ID:** MSSQLSERVER

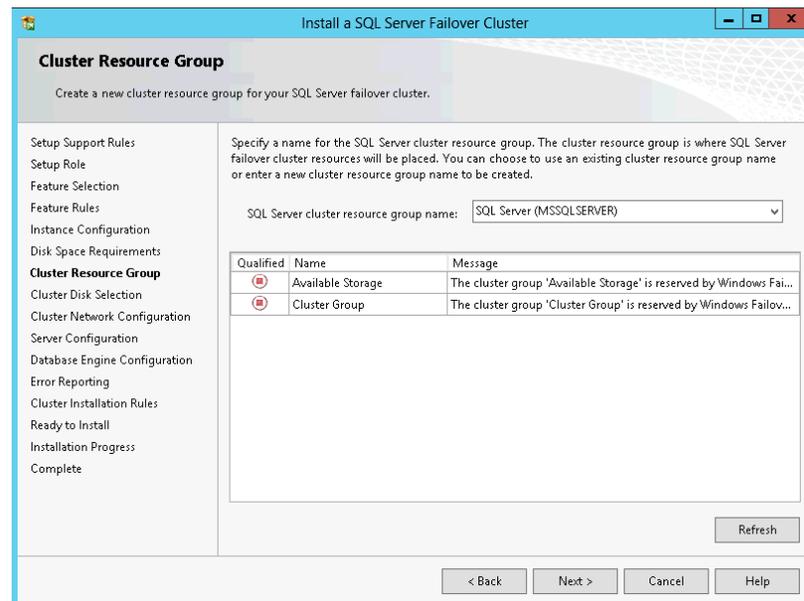
Click **Next**.



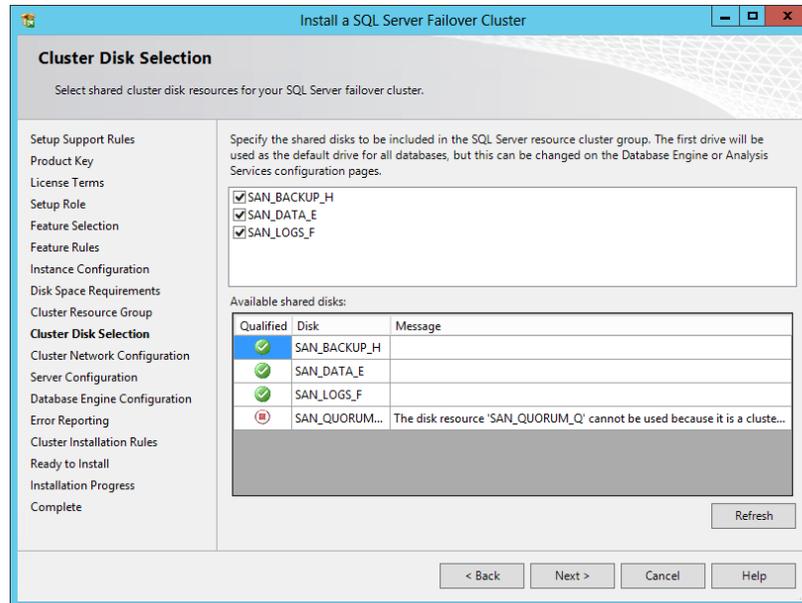
12. In the **Disk Space Requirements** dialog box, check that you have enough space on your local disks to install the SQL Server 2012 binaries. Click **Next**.



13. In the **Cluster Resource Group** dialog box, check the resources available on your Windows Server Failover Cluster. This tells you that a new Resource Group will be created on your cluster for the SQL Server instance. To specify the SQL Server cluster resource group name, you can either use the drop-down box to specify an existing group to use or type the name of a new group to create it. Accept all the defaults and click **Next**.

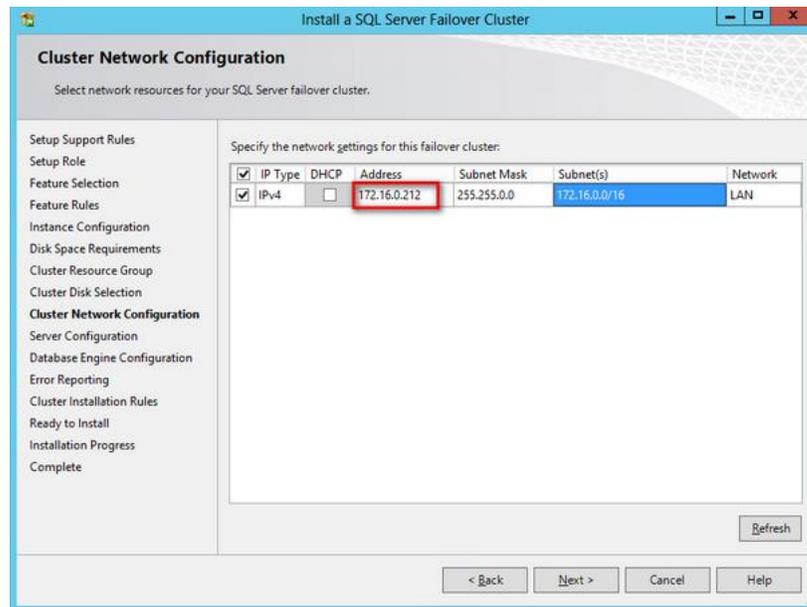


14. In the **Cluster Disk Selection** dialog box, select the available disk groups that are on the cluster for SQL Server 2012 to use. Click **Next**.



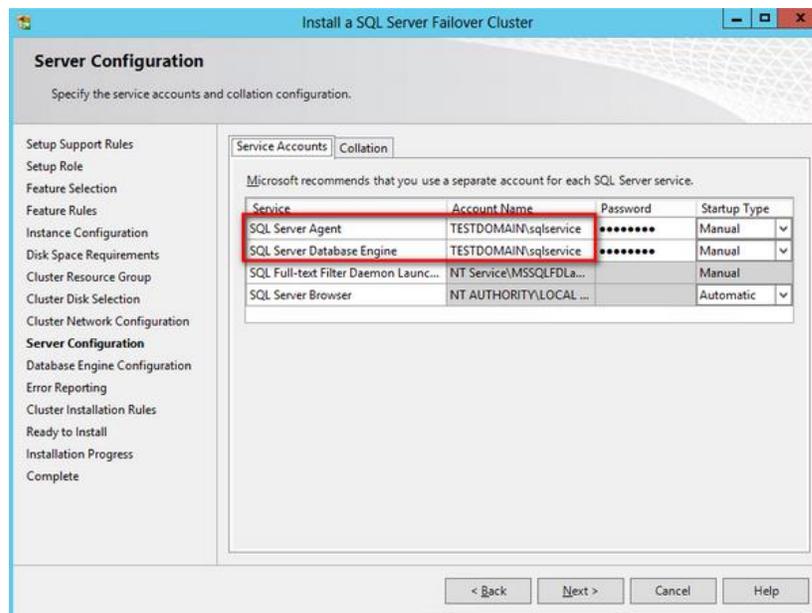
15. In the **Cluster Network Configuration** dialog box, enter the virtual IP address that your SQL Server 2012 failover cluster will use.

- **IP Type:** IPv4
- **Address:** 172.16.0.212



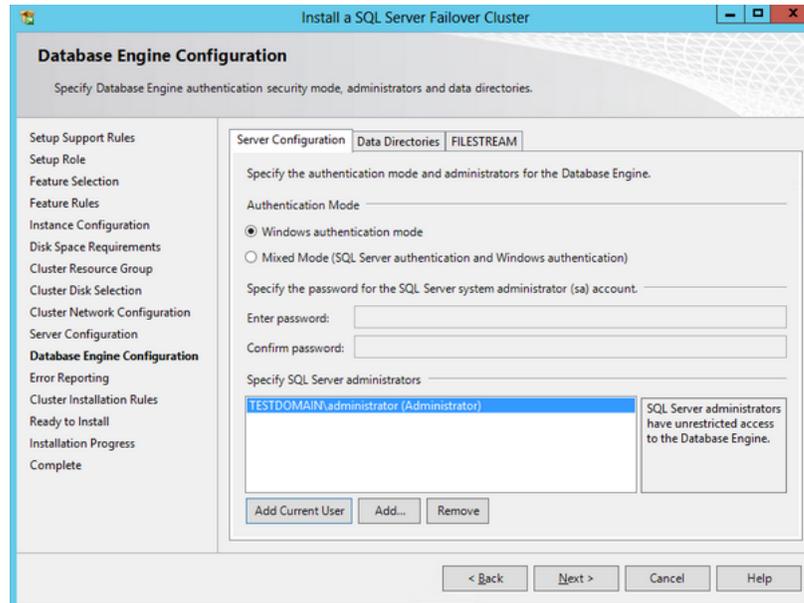
16. In the **Server Configuration** dialog box, use the following credentials for the SQL Server service accounts in the **Service Accounts** tab. Make sure that both the SQL Server Agent and SQL Server Database Engine services have a **Startup Type** of **Manual**. The Windows Server Failover Cluster will take care of stopping and starting the service.

- **SQL Server Agent:** TESTDOMAIN\sqlservice
- **SQL Server Database Engine:** TESTDOMAIN\sqlservice



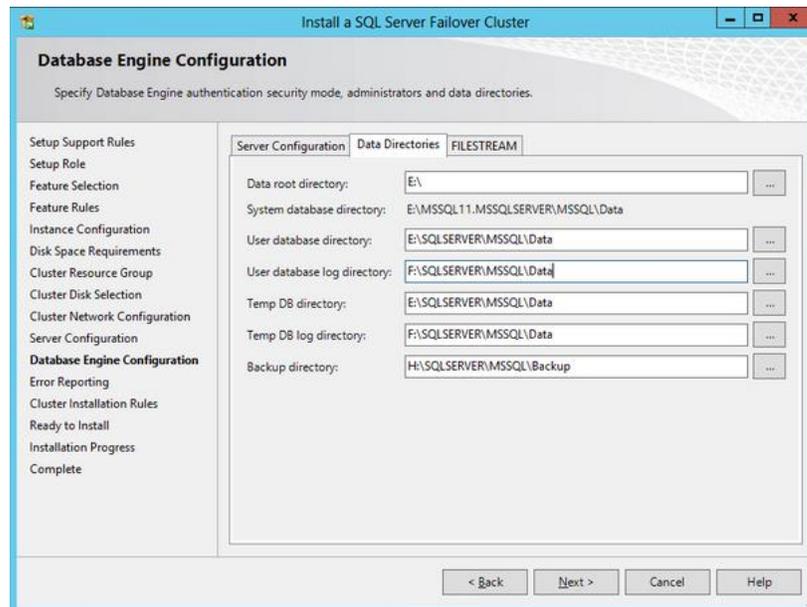
Click **Next**.

17. In the **Database Engine Configuration** dialog box, select **Windows Authentication Mode** in the **Server Authentication** tab. If you want to add the currently logged on user to be a part of the SQL Server administrators group, click the **Add Current User** button. Otherwise, you can add the appropriate domain accounts or security groups.

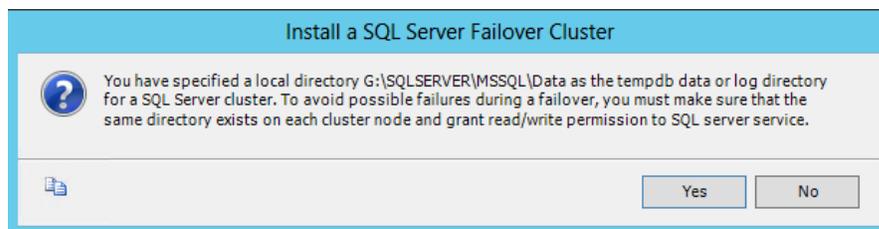


In the **Data Directories** tab, enter the following

- **Data root directory:** E:\
- **User database directory:** E:\SQLSERVER\MSSQL\Data
- **User database log directory:** F:\SQLSERVER\MSSQL\Data
- **Temp DB directory:** E:\SQLSERVER\MSSQL\Data
- **Temp DB log directory:** F:\SQLSERVER\MSSQL\Data
- **Backup directory:** H:\SQLSERVER\MSSQL\Backup

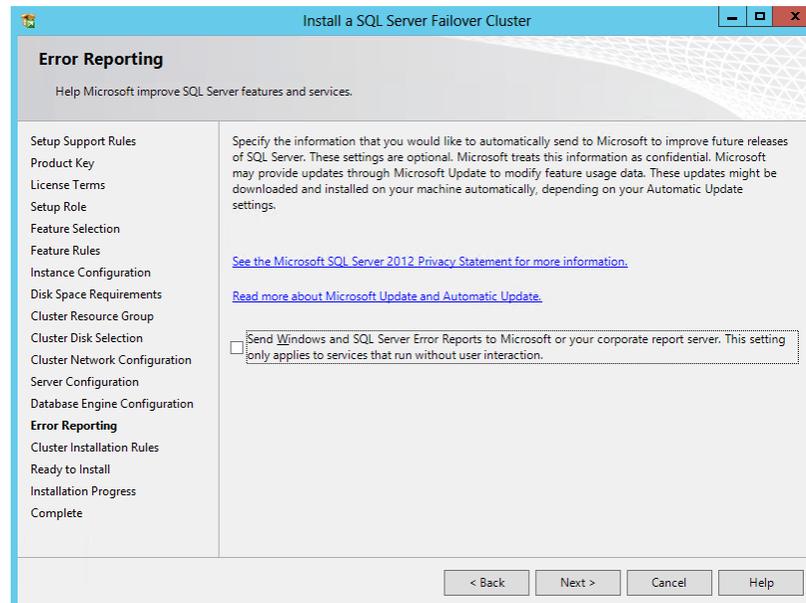


NOTE: New in SQL Server 2012 is the option to store the **tempdb** database on a local drive instead of a clustered drive. Should you decide to do so, you will get prompted to make sure that all of the nodes in the cluster contain the same directory structure and that the SQL Server service account has read/write permissions on those folders.

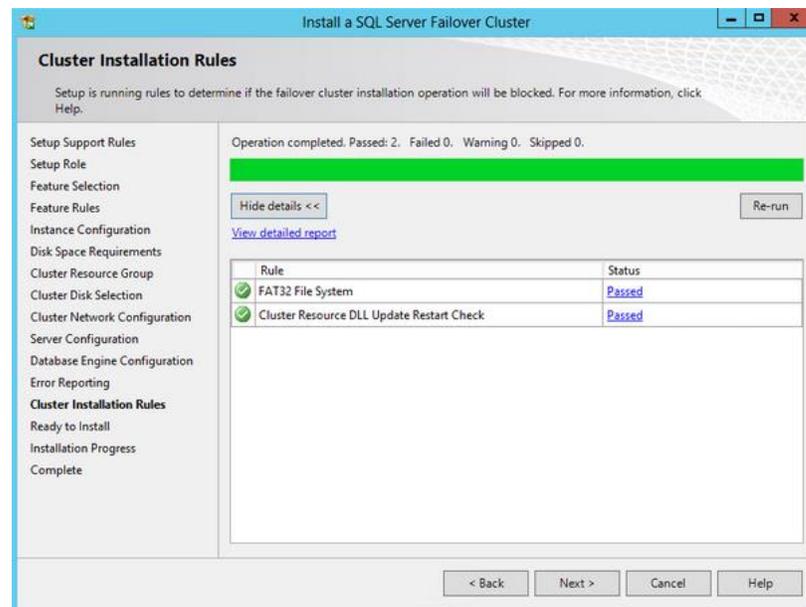


Click **Yes**. Then, click **Next**.

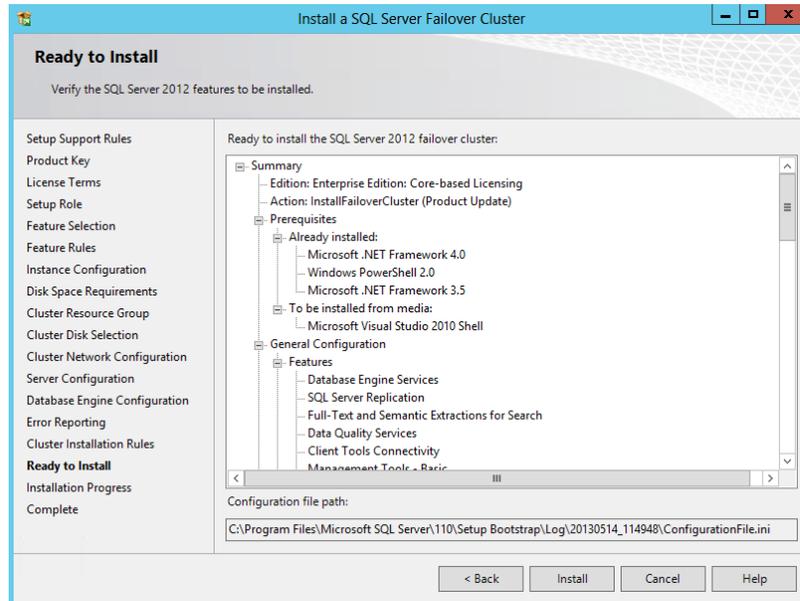
18. In the **Error and Usage Reporting** dialog box, click **Next**.



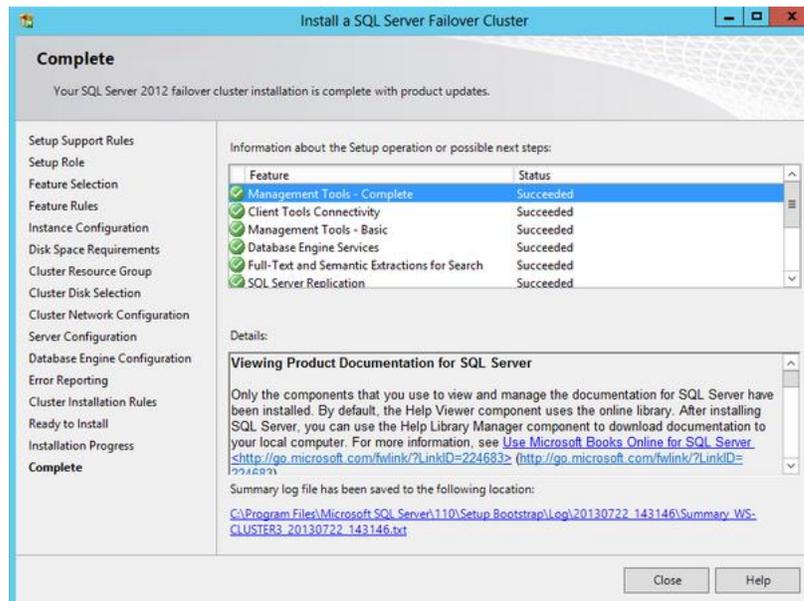
19. In the **Cluster Installation Rules** dialog box, verify that all checks are successful. Click **Next**.



20. In the **Ready to Install** dialog box, verify that all configurations are correct. Click **Next**.



21. In the **Complete** dialog box, click **Close**. This concludes the installation of a SQL Server 2012 Failover Clustered Instance.

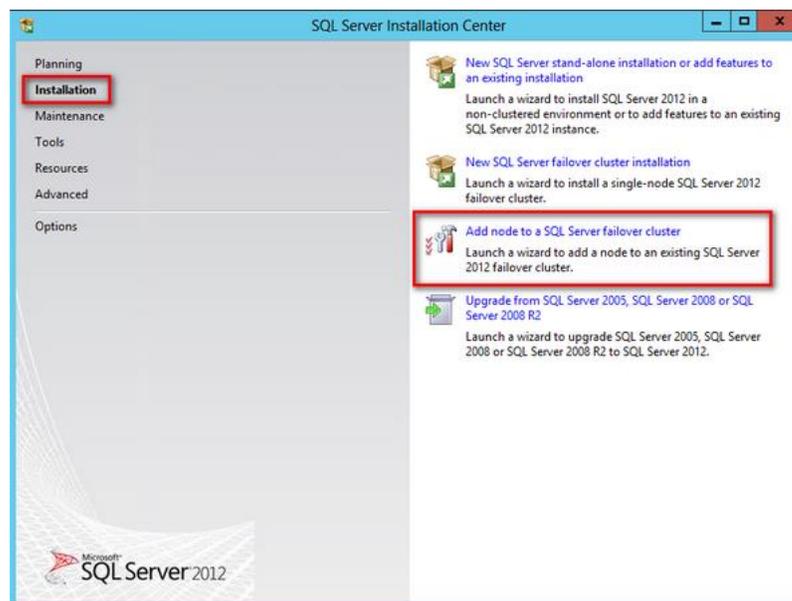


SQL Server 2012 Failover Cluster Instance Installation - Install Secondary (Failover) Cluster Node

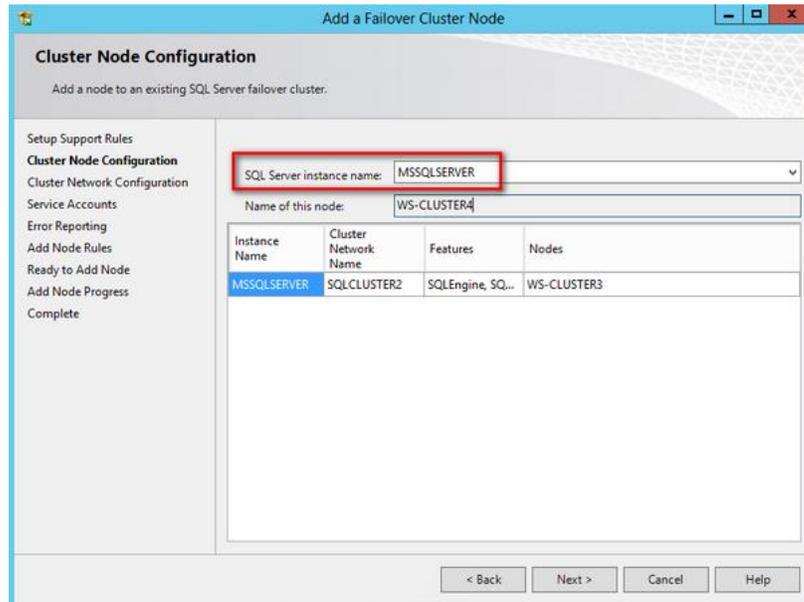
Now that you have a working failover cluster, we will make it highly available by adding nodes. Most of the steps outlined below are similar to the ones when installing a failover clustered instance. While all the steps are still included, some screenshots have been omitted for brevity.

To add a node to a SQL Server 2012 failover clustered instance,

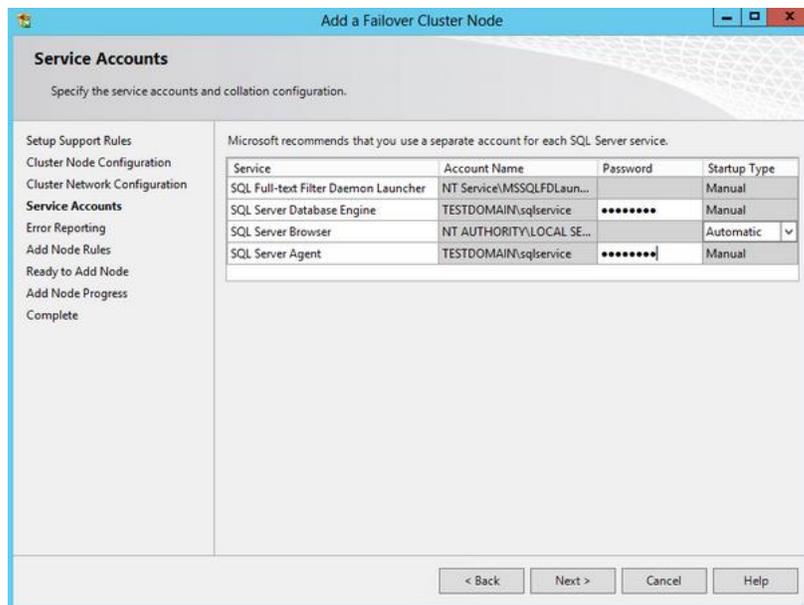
1. Run **setup.exe** from the installation media to launch **SQL Server Installation Center**. Click on the **Installation** link on the left-hand side.
2. Click on the **Add node to a SQL Server failover cluster** link. This will run the SQL Server 2012 Setup wizard.



3. In the **Setup Support Rules** dialog box, validate that the checks return successful results and click **OK**. Click **Next** until you get to the **Cluster Node Configuration** dialog box.
4. In the **Cluster Node Configuration** dialog box, validate that the information on the existing SQL Server 2012 cluster is correct. Click **Next**.



5. In the **Service Accounts** dialog box, verify that the information is the same as what you have used to configure the first node. Click **Next**.



NOTE: If you used different service accounts for the SQL Server services, make sure you provided the correct credentials. If you noticed, the order of the services in this dialog box differs from when you are installing a failover clustered instance.

6. In the **Error and Usage Reporting** dialog box, click **Next**.
7. In the **Add Node Rules** dialog box, verify that all checks are successful and click **Next**.
8. In the **Ready to Add Node** dialog box, verify that all configurations are correct and click **Install**.
9. In the **Complete** dialog box, click **Close**. This concludes adding a node to a SQL Server 2012 Failover Cluster.
10. Reboot both nodes after completing the installation just to be sure that there are no pending reboot operations that may possibly be flagged as a warning.

About The Author



Edwin M Sarmiento is a **Microsoft SQL Server MVP** and **Microsoft Certified Master** from Ottawa, Canada specializing in high availability, disaster recovery and system infrastructures running on the Microsoft server technology stack - ranging from Active Directory to SharePoint and anything in between. He is very passionate about technology but has interests in music, professional and organizational development, leadership and management matters when not working with databases. He lives up to his primary mission statement – *“To help people grow and develop their full potential as God has planned for them.”*

He wants the whole world to know that the FILIPINO is a world-class citizen and brings JESUS CHRIST to the world.

 <http://www.EdwinMSarmiento.com>

 edwin@EdwinMSarmiento.com

 [@EdwinMSarmiento](https://twitter.com/EdwinMSarmiento)

 <http://ca.linkedin.com/in/EdwinMSarmiento>

Did this document help you? Send feedback, comments and suggestions to:

edwin@EdwinMSarmiento.com

Interested in requesting the author for training, consulting, solutions architecture, whiteboard sessions or team mentoring, send an email to: edwin@EdwinMSarmiento.com