Technical Report

FlexPod with Microsoft SQL Server 2022 Clustering

High Availability with SQL Server Failover Cluster Instance

Kamini Singh, NetApp
July 2024 | TR-4995

In partnership with

Abstract

This document describes the design and implementation of a FlexPod® data center solution with Microsoft SQL Server Clustering which is designed for high availability of enterprise database systems. This solution covers best practices to deploy FlexPod with SQL Server Failover Cluster Instance to ensure the reliability and availability of various databases and applications.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Solution Overview

It is important that a datacenter solution embraces technology advancement in various areas, such as compute, network, and storage technologies to address rapidly changing requirements and challenges of IT organizations. The current industry trend in datacenter design is towards shared infrastructures. By using virtualization along with pre-validated IT platforms, enterprise customers have embarked on the journey to the cloud by moving away from application silos and toward shared infrastructure that can be quickly deployed, thereby increasing agility, and reducing costs.

The consolidation of IT applications, particularly databases, has generated considerable interest in the recent years. Being most widely adopted and deployed database platform over several years, Microsoft SQL Server databases have become the victim of a popularly known IT challenge “Database Sprawl.” Some of the challenges of SQL Server sprawl include underutilized Servers, high licensing costs, security concerns, management concerns, huge operational costs and so on. Therefore, SQL Server databases would be right candidate for migrating and consolidating on to a more robust, flexible, and resilient platform.

FlexPod® is a pre-validated and engineered datacenter solution designed to address rapidly changing needs of IT organizations. Cisco® and NetApp® have partnered to deliver FlexPod®, which uses best of breed compute, network, and storage components to serve as the foundation for a variety of enterprise workloads including databases, ERP, CRM, and Web applications, and so on. FlexPod systems incorporate resilient design across all layers of the infrastructure with no single point of failure. This is enabled by having redundant components and redundant connectivity between them.

NetApp storage systems harness the power of ONTAP® to simplify the data infrastructure from edge, core, and cloud with a common set of data services and 99.9999 percent availability. NetApp ONTAP 9 data management software enables you to modernize your infrastructure and transition to a cloud-ready data center. ONTAP 9 has set of features to simplify deployment and data management, accelerate and protect critical data, and make infrastructure future-ready across hybrid-cloud architectures.

A Windows Server Failover Cluster (WSFC) is a group of independent servers that work together to increase the availability of applications and services. SQL Server takes advantage of WSFC services and capabilities to support SQL Server Failover Cluster Instances. Windows Server Failover Clustering provides infrastructure features that support the high-availability and disaster recovery scenarios of hosted server applications such as Microsoft SQL Server and Microsoft Exchange. If a cluster node or service fails, the services that were hosted on that node can be automatically or manually transferred to another available node in a process known as failover.

A SQL Server failover cluster is also known as a High-availability cluster, as it provides redundancy for critical systems. The main concept behind failover clustering is to eliminate a single point of failure by including multiple network connections and shared data storage connected via SAN (Storage area network) or NAS (Network attached storage).

Each node in a cluster environment is monitored all the time via a network connection mechanism called the heartbeat. The cluster will automatically configure cluster heartbeat networking when you setup the cluster and add nodes to an existing cluster. A system must be able to overcome the situation called “split-brain” which occurs if all heartbeat links go down simultaneously. Then, all other nodes can conclude that one node is down and will try to restart the application on themselves. A failover cluster uses a quorum-based approach to monitor overall cluster health and maximize node-level fault tolerance.

SQL Server is the most prevalent workload for FlexPod's enterprise customer base. Ensuring the reliability and availability of SQL Server is paramount for businesses that rely on database systems for their critical operations. By validating FlexPod with SQL Server 2022 and focusing on high availability solutions like SQL Server Failover Cluster, this solution aims to help customers maintain continuous database availability, even in the event of node failures or outages. This solution showcases best practices for configuring SQL Server Failover Cluster Instance (FCI) on FlexPod. This validation will help

customers confidently deploy SQL Server 2022 on FlexPod, knowing that they have a resilient and efficient system that can handle their most demanding database needs.

Figure 1 shows a generic illustration of FlexPod with SQL Server Failover Clustering consisting of two nodes that are capable of performing failover operations. Both failover clustering nodes share the same NetApp ONTAP storage, which could be offered via AFF/FAS/ASA storage systems.

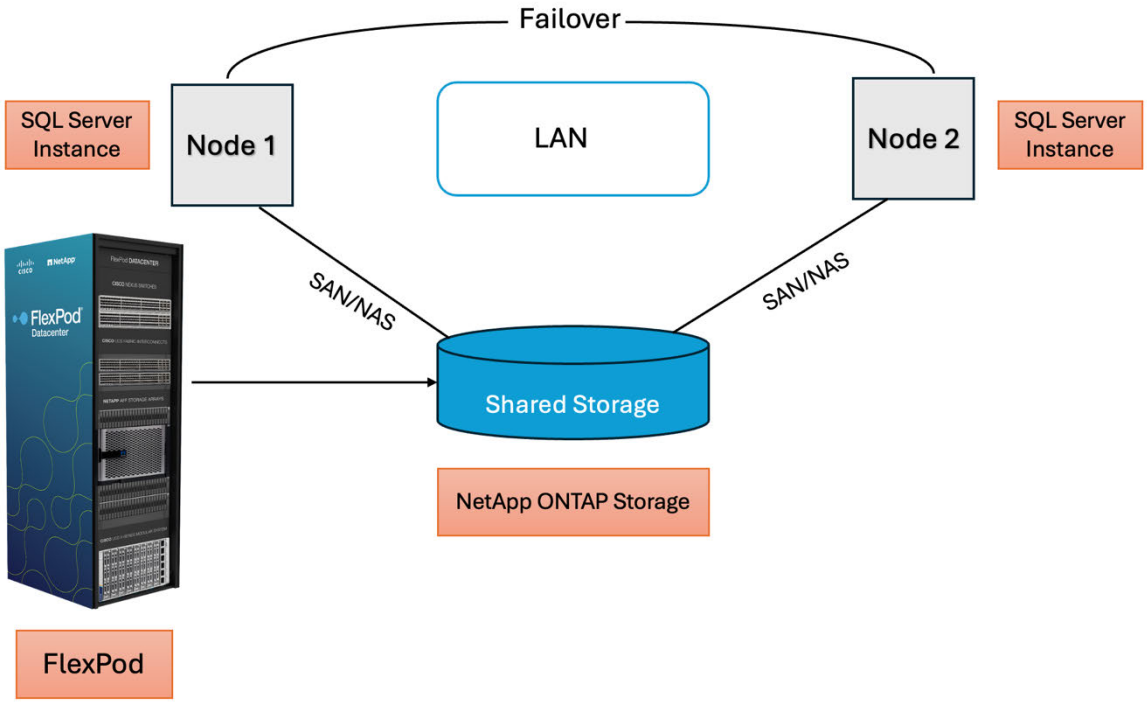


Figure 1 Generic Illustration of FlexPod with SQL Server Failover Clustering

In a SQL Server failover cluster, data needs to be on a shared storage. Since the data is shared among the nodes, the SQL Server failover cluster instance can be moved from one node to the other in case of any failure or maintenance. This solution can guarantee higher up-time and redundancy. Because there is only one storage space, regular SQL Server maintenance requirements are still needed. Also, if the shared storage isn't redundant, after a storage failure, the SQL Server database will be unavailable.

Audience

The intended audience for this document includes, but is not limited to, sales engineers, field consultants, database administrators, professional services, IT managers, partner engineers, and customers who want to take advantage of an infrastructure built to deliver IT efficiency and enable IT innovation. It is expected that the reader should have prior knowledge on FlexPod Systems and its components.

Solution Benefits

FlexPod systems integrated with SQL Server failover clustering offers the following benefits:

- Provides a reliable high availability solution for enterprise database systems. This reduces risk of data loss and service interruptions.
- Minimizes downtime and ensures business continuity. For example – implementing SQL Server failover cluster ensures minimum downtime during hardware failures or maintenance.
- Simplifies management of SQL Server environments.
- Distributing the workload across multiple SQL Server nodes optimizes performance. For example – A healthcare provider with high query volumes needs to balance the load across servers.
- Supports a broad array of storage solutions, including WSFC cluster disks (iSCSI, Fiber Channel, and so on) and server message block (SMB) shares.
- Zero reconfiguration of applications and clients during failovers.
- Throttled resource usage during failovers.

Solution Topology

This section describes the topology of the solution which was used for testing and validation.

FlexPod Topology

FlexPod® is a defined set of hardware and software that serves as an integrated foundation for both virtualized and non-virtualized solutions. VMware vSphere built on FlexPod includes NetApp All Flash FAS storage, Cisco Nexus networking, Cisco Unified Computing System, and VMware vSphere software in a single package. The design is flexible enough that the networking, computing, and storage resources can fit in one datacenter rack or be deployed according to a customer's datacenter design.

FlexPod Datacenter with Cisco UCS X-Series supports both IP and Fibre Channel (FC)—based storage access design. This document covers the IP-based solution using Cisco UCS M7 servers. For this solution, iSCSI configuration on Cisco UCS and NetApp AFF A400 is utilized to set up boot from SAN for the Compute Node. VMware ESXi hosts access the VM datastore volumes on NetApp using NFS and iSCSI. The physical connectivity details for IP-based design are explained below.

Figure 2 shows the FlexPod components and the network connections for a configuration with Cisco UCS 6536 Fabric Interconnects. This design can support end to end 100-Gbps Ethernet connections between the NetApp AFF A400 storage array and Cisco UCS X210c M7 compute nodes.

Each of the components can be scaled easily to support specific business requirements. For example, more (or different) servers or blade chassis can be deployed to increase computing capacity, additional storage controllers or disk shelves can be deployed to improve I/O capability and throughput, and special hardware or software features can be added to introduce new features.

The following components were used to validate and test the solution:

- One Cisco UCSX-9508 blade chassis with Cisco UCS 9108 25G IFM modules
- Two Cisco UCS X210c M7 compute nodes with the Cisco UCS VIC 15231
- Two Cisco UCS 6536 Fabric Interconnects
- Two Cisco Nexus 9336C-FX2 Switches

- One NetApp AFF A400 (high-availability pair) running clustered NetApp ONTAP with NVMe disk shelves

The physical topology for the IP-based FlexPod Datacenter is shown below.

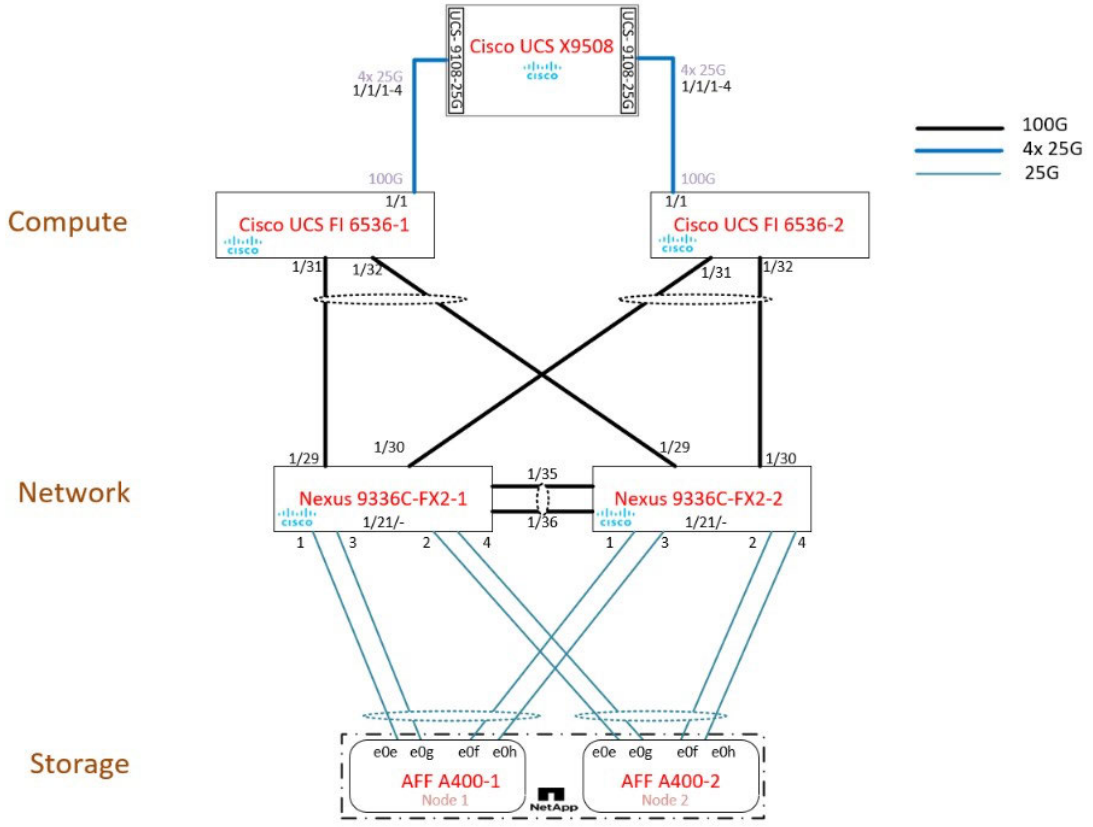


Figure 2 FlexPod Datacenter Physical Topology for iSCSI and NFS

To validate the IP-based storage access in this FlexPod configuration, the components are set up as follows:

- Cisco UCS 6536 Fabric Interconnects provide chassis and network connectivity.
- The Cisco UCS X9508 Chassis connects to fabric interconnects using Cisco UCSX-I-9108-25G intelligent fabric modules (IFMs), where four 25 Gigabit Ethernet ports are used on each IFM to connect to the appropriate FI. If additional bandwidth is required, all eight 25G ports can be utilized.
- Cisco UCSX-210c M7 Compute Nodes contain fifth-generation Cisco 15231 virtual interface cards (VICs) which are used for 100Gbps connectivity on each side of fabric interconnect.
- Cisco Nexus 9336C-FX2 Switches in Cisco NX-OS mode provide the switching fabric. Cisco UCS 6536 Fabric Interconnect 100-Gigabit Ethernet uplink ports connect to Cisco Nexus 9336C-FX2 Switches in a Virtual Port Channel (vPC) configuration.
- The NetApp AFF A400 controllers connect to the Cisco Nexus 9336C-FX2 Switches using four 25 GE ports from each controller configured as a vPC.

- VMware 8.0 ESXi software is installed on Cisco UCS X210c M7 Compute Nodes and servers to validate the infrastructure.

In this solution, VMware ESXi 8.0 virtual environment was tested and validated for deployment of Microsoft SQL Server 2022 databases on virtual machines running Microsoft Windows Server 2022 guest operating system. SQL Server virtual machines are configured to connect the NetApp AFF A400 storage LUNs directly using the in-guest Microsoft software iSCSI initiator. This approach bypasses the ESXi hypervisor VMFS storage layer for the LUNs that are used for storing SQL Server database files. This design approach provides better performance, simplifies management, enables efficient backup of data, and allows the association of storage QoS directly to objects hosting SQL Server data.

For detailed information about FlexPod design used in this solution, refer to the following link:

https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_sql2022_xseries.html - SolutionDesign

SQL Server Clustering Architecture

A Microsoft SQL Server Cluster is a collection of two or more physical or virtual servers with identical access to shared storage that provides the disk resources required to store the database files. These servers are referred to as “nodes”.

This section describes the SQL Server Failover Clustering architecture that was implemented and utilized for validation in this solution.

A Windows Failover Cluster uses shared storage – typically this shared storage is on a SAN. When a SQL Server Failover Cluster instance (FCI) is installed on the cluster, system and user databases are required to be on the shared storage. This allows the cluster to move the SQL instance to any server (or node) in the cluster whenever you request, or if one of the nodes is having a problem. There is only one copy of the data, but the network name and SQL Server service for the instance can be made active from any cluster node.

Figure3 illustrates SQL Server failover clustering architecture.

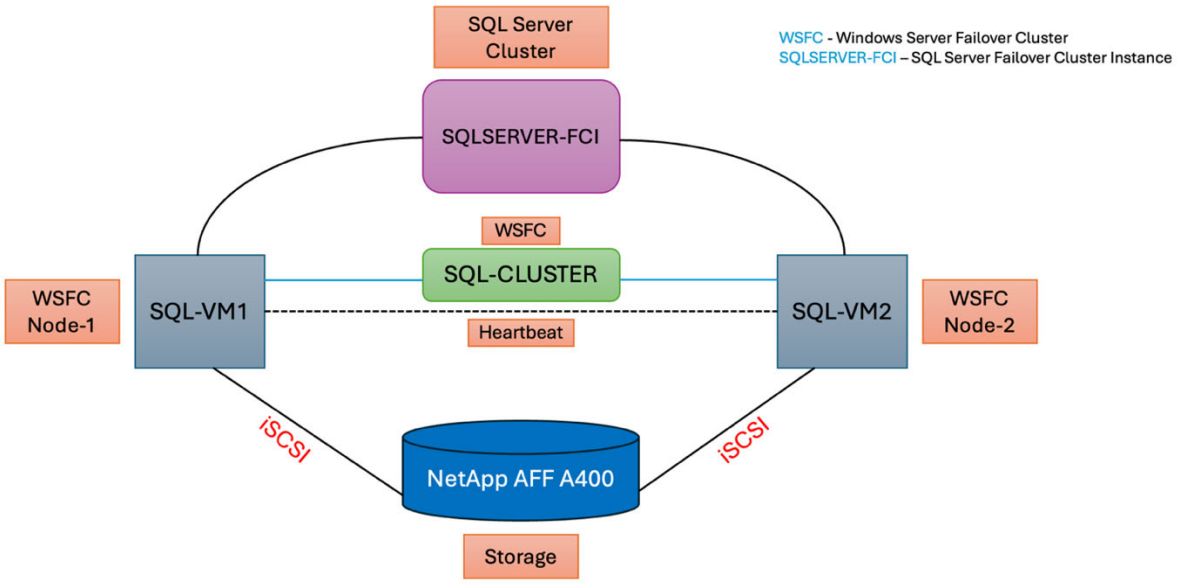


Figure 3 SQL Server Failover Clustering Architecture

Some key points specific to the above design and implementation are as follows:

- A Windows Server Failover Cluster (WSFC) named SQL-CLUSTER is deployed that consists of two servers (or nodes), named SQL-VM1 and SQL-VM2.
- A Failover Cluster Instance (FCI) is a single instance of SQL Server that is installed across Windows Server Failover Clustering nodes. In this solution, one SQL Server Failover Cluster instance is deployed, called SQLSERVER-FCI.
- Also, there is shared NetApp AFF A400 storage connected to the servers (nodes) via iSCSI. The storage volumes are connected to two nodes (VMs) while the SQL Server FCI is a logical entity which runs on one of these nodes. SQL Server FCI service access storage via the nodes.
- Each of the WSFC nodes talk to one another via a private network, sending a heartbeat signal between them. If one node does not communicate its heartbeat to the other node in the cluster, the secondary node will take ownership of any dependent services being run by the node that lost communication. This process is referred to as “failover”.
- When the server (or node) SQL-VM1 crashes, the failover cluster service in SQL-CLUSTER is aware of the situation through the heartbeat and automatically starts the SQL Server instance SQLSERVER-FCI on the SQL-VM2 server.

Solution Components

This section describes the components that were used and validated in this solution.

NetApp AFF A400

The NetApp AFF A400 offers full end-to-end NVMe support. The frontend FC-NVMe connectivity makes it possible to achieve optimal performance from an all-flash array for workloads that include artificial intelligence, machine learning, and real-time analytics as well as business-critical databases. The frontend NVMe-TCP connectivity enables you to take advantage of NVMe technology over existing ethernet infrastructure for faster host connectivity. On the back end, the NetApp AFF A400 supports both serial-attached SCSI (SAS) and NVMe-attached SSDs, offering the versatility for you to move up from your legacy A-Series systems and satisfying the increasing interest in NVMe-based storage.

The NetApp AFF A400 offers greater port availability, network connectivity, and expandability. The NetApp AFF A400 has 10 PCIe Gen3 slots per high availability pair. The NetApp AFF A400 offers 10GbE, 25GbE and 100GbE ports for IP based transport, and 16/32Gb ports for FC and FC-NVMe traffic. This model was created to keep up with changing business needs and performance and workload requirements by merging the latest technology for data acceleration and ultra-low latency in an end-to-end NVMe storage system.

Designed for enterprises that run all types of workloads in both SAN and NAS environments, the AFF A400 meets and exceeds all the demands, be it online transaction processing, virtualization, or file sharing. Running as fast as a high-end system that fits into a mid-range budget, it also stands out for its "NetApp signature" cloud integration, which enables easy cloud backup, tiering, caching and more, all built-in with the amazing ONTAP data management software.

For more information about the NetApp AFF A400 controllers, refer to the AFF A400 product page: https://www.netapp.com/data-storage/aff-a-series/aff-a400/.

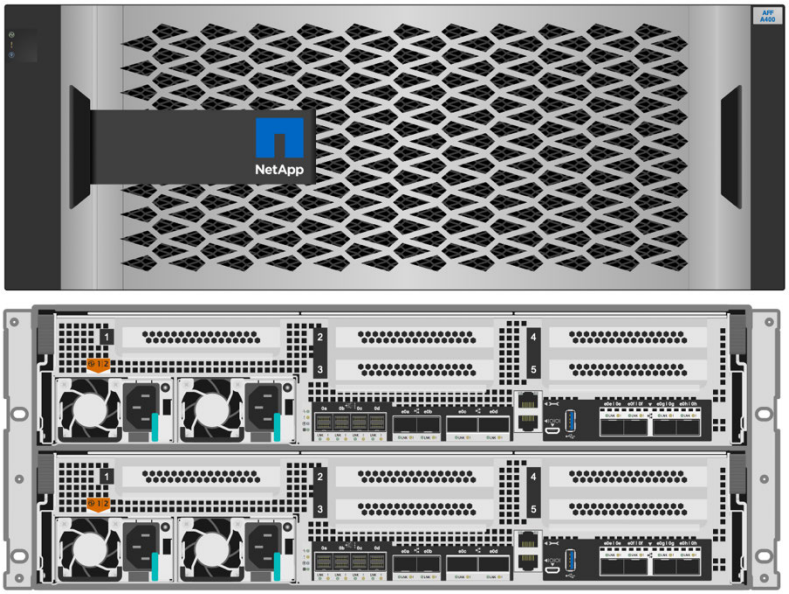


Figure 4 NetApp AFF A400 Front and Rear View

Cisco UCS X9508 Chassis

The Cisco UCS X-Series Modular System begins with the Cisco UCS X9508 Chassis, engineered to be adaptable and future-ready. The X-Series is a standards-based open system designed to be deployed and automated quickly in a hybrid cloud environment.

With a midplane-free design, I/O connectivity for the X9508 Chassis is accomplished with front-loading vertically oriented computing nodes that intersect with horizontally oriented I/O connectivity modules in the rear of the chassis. Cisco UCS X-Series is powered by Cisco Intersight, making it simple to deploy and manage at scale.

The Cisco UCS X9508 Chassis provides these features and benefits:

- The Seven-Rack-Unit (7RU) chassis has eight front-facing flexible slots. These can house a combination of computing nodes and a pool of future I/O resources, which may include Graphics Processing Unit (GPU) accelerators, disk storage, and non-volatile memory.
- X9508 Chassis supports two types of Intelligent Fabric Modules (IFMs). One is 9108 25G IFM and the other is 9108 100G IFM.
    - The 100G IFM has eight 100G ports and it provides up to 200 Gbps of unified fabric connectivity per computing node.
    - The 25G IFM has eight 25G ports and it provides up to 100 Gbps of unified fabric connectivity per computing node
- At the bottom of the chassis are slots ready to house future I/O modules that can flexibly connect the computing modules with I/O devices.
- Six 2800W Power Supply Units (PSUs) provide 54V power to the chassis with N, N+1, and N+N redundancy. A higher voltage allows efficient power delivery with less copper and reduced power loss.

- Efficient, 4x100mm, dual counter-rotating fans deliver industry-leading airflow and power efficiency.

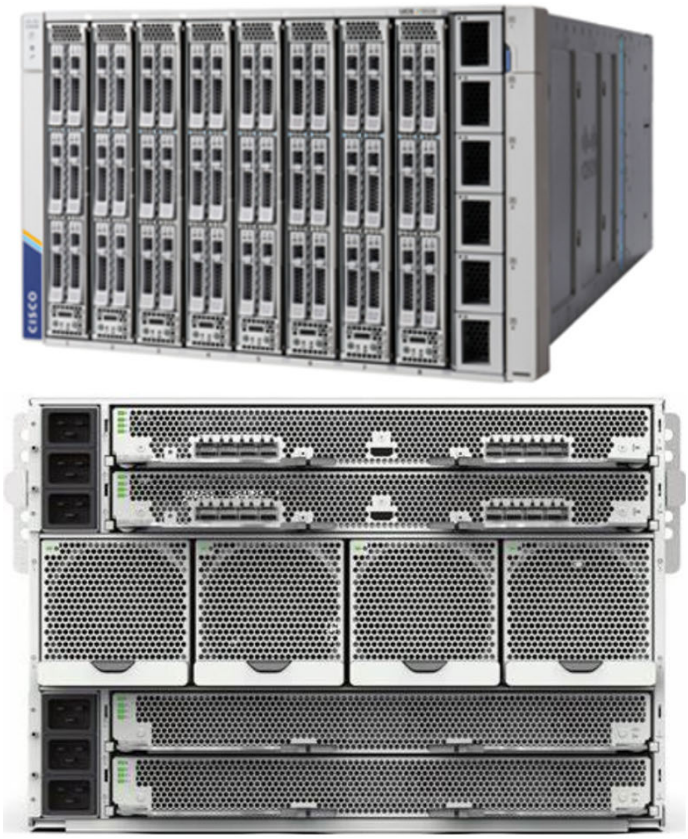


Figure 5 Cisco UCS X9508 Chassis Front and Back View

Cisco UCSX-I-9108 25G Intelligent Fabric Module

For the Cisco UCS X9508 Chassis, the network connectivity is provided by a pair of Cisco UCSX 9108-25G Intelligent Fabric Modules (IFMs). Like the fabric extenders used in the Cisco UCS 5108 Blade Server Chassis, these modules carry all network traffic to a pair of Cisco UCS 6500 Series Fabric Interconnects (FIs). IFMs also host the Chassis Management Controller (CMC) for chassis management. In contrast to systems with fixed networking components, Cisco UCS X9508's midplane-free design enables easy upgrades to new networking technologies as they emerge making it straightforward to accommodate new network speeds or technologies in the future.

Figure 6 Cisco UCSX-I-9108 25G Intelligent Fabric Module

Each IFM supports eight 25Gb uplink ports for connecting the Cisco UCS X9508 Chassis to the FIs and 32 25Gb server ports for the eight compute nodes. IFM server ports can provide up to 200 Gbps of unified fabric connectivity per compute node across the two IFMs. The uplink ports connect the chassis to the UCS FIs, providing up to 400Gbps connectivity across the two IFMs.

Cisco UCS X210c M7 Compute Node

The Cisco UCS X9508 Chassis is designed to host up to 8 Cisco UCS X210c M7 Compute Nodes. The front view of Cisco UCS X210c M7 Compute Node is shown in Figure 7.

Figure 7 Cisco UCS X210c M7 Compute Node

The Cisco UCS X210c M7 features:

- **CPU:** Up to 2x 4th Gen Intel Xeon Scalable Processors with up to 60 cores per processor and 2.625 MB Level 3 cache per core and up to 112.5 MB per CPU.
- **Memory:** Up to 32 x 256 GB DDR5-4800 DIMMs for a maximum of 8 TB of main memory.
- **Disk storage:** Up to 6 SAS or SATA drives or NVMe drives can be configured with the choice of an internal RAID controller or passthrough controllers. Two M.2 memory cards can be added to the Compute Node with optional hardware RAID.
- **GPUs:** The optional front mezzanine GPU module allows support for up to two HHHL GPUs. Adding a mezzanine card and a Cisco UCS X440p PCIe Node allows up to four more GPUs to be supported with a Cisco UCS X210c M7.
- **Virtual Interface Card (VIC):** Up to 2 VICs including an mLOM Cisco UCS VIC 15231 or an mLOM Cisco UCS VIC 15420 and a mezzanine Cisco UCS VIC card 15422 can be installed in a Compute Node.
- **Security:** The server supports an optional Trusted Platform Module (TPM). Additional security features include a secure boot FPGA and ACT2 anticounterfeit provisions.

Cisco UCS Virtual Interface Card 15231 (VIC)

Cisco UCS X210c M7 Compute Nodes support multiple Cisco UCS VIC cards. In this solution, Cisco UCS VIC 15231 is used for the validation.

The Cisco UCS VIC 15231 are 2x100-Gbps Ethernet/FCoE-capable modular LAN on motherboard (mLOM) adapters designed exclusively for the Cisco UCS X210c Compute Node. The Cisco UCS VIC 15231 adapters enable a policy-based, stateless, agile server infrastructure that can present to the host PCIe standards-compliant interfaces that can be dynamically configured as either NICs or HBAs. In this solution, we have used 2 VIC cards per UCS Server.

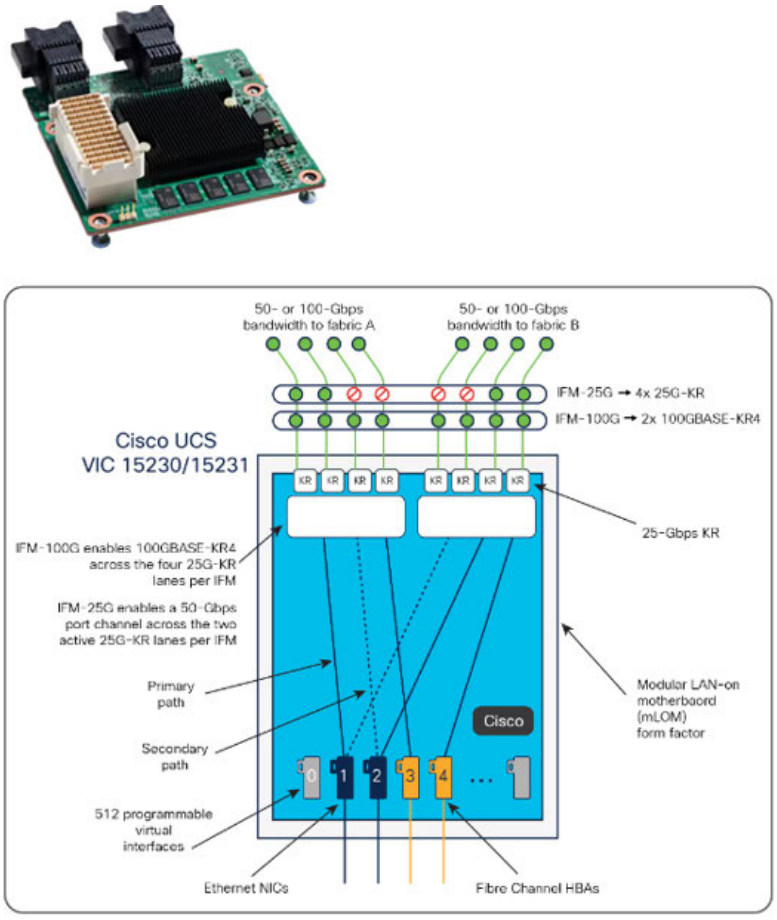


Figure 8 Cisco UCS VIC 15231

Cisco UCS 6536 Fabric Interconnects

The Cisco UCS 6536 Fabric Interconnect (FI) is a core part of the Cisco Unified Computing System, providing both network connectivity and management capabilities for the system. The Cisco UCS 6536 Fabric Interconnect offers line-rate, low-latency, lossless 10/25/40/100 Gigabit Ethernet, Fibre Channel, NVMe over Fabric, and Fibre Channel over Ethernet (FCoE) functions. In addition, by supporting a unified fabric, Cisco UCS 6536 Fabric Interconnect provides both LAN and SAN connectivity for all servers within its domain.

Figure 9 Cisco UCS 6536 Fabric Interconnects

From a networking perspective, the Cisco UCS 6536 uses a cut-through architecture, supporting deterministic, low-latency, line-rate 10/25/40/100 Gigabit Ethernet ports, a switching capacity of 7.42 Tbps per FI and 14.84 Tbps per unified fabric domain, independent of packet size and enabled services. With the X9108-IFM-25G, it enables 400Gbps bandwidth per chassis per FI domain.

Cisco Nexus 9336C-FX2 Switches

The Cisco Nexus 9336C-FX2 switch offers flexible port speeds supporting 1/10/25/40/100 Gbps in a compact 1 RU form factor with cloud-scale technology. It is designed to meet the changing needs of data centers, big data applications, and automated cloud environments.

- All 36 ports support 10/25/40/100 Gbps QSFP28 and wire-rate MACsec encryption.
- Supports 7.2 Tbps of bandwidth and over 2.8 bpps.
- Enhanced Cisco NX-OS Software designed for performance, resiliency, scalability, manageability, and programmability.
- Real-time buffer utilization per port and per queue, for monitoring traffic micro-bursts and application traffic patterns.

Figure 10 Cisco Nexus 9336C-FX2 Switch

Cisco Intersight

The Cisco Intersight platform is a Software-as-a-Service (SaaS) infrastructure lifecycle management platform that delivers simplified configuration, deployment, maintenance, and support. The Cisco Intersight platform is designed to be modular, so you can adopt services based on your individual requirements. The platform significantly simplifies IT operations by bridging applications with infrastructure, providing visibility and management from bare-metal servers and hypervisors to serverless applications, thereby reducing costs and mitigating risk. This unified SaaS platform uses a unified Open API design that natively integrates with third-party platforms and tools.

There are two modes of management operations possible with Cisco Intersight: UCSM Managed Mode (UMM) and Intersight Managed Mode (IMM). You can select the native UCSM Managed Mode (UMM) or Intersight Managed Mode (IMM) for fabric-attached Cisco UCS systems during the initial setup of the fabric Interconnects. In this solution, native IMM is used.

VMware vSphere 8.0

VMware vSphere is a virtualization platform for holistically managing large collections of infrastructures (resources including CPUs, storage, and networking) as a seamless, versatile, and dynamic operating environment. Unlike traditional operating systems that manage an individual machine, VMware vSphere aggregates the infrastructure of an entire data center to create a single powerhouse with resources that can be allocated quickly and dynamically to any application in need.

VMware vSphere 8.0 has several improvements and simplifications including, but not limited to:

- Limits with VMware vSphere 8.0 have been increased including the number of GPU devices increased to 8, the number of ESXi hosts that can be managed by Lifecycle Manager is increased

from 400 to 1000, the maximum number of VMs per cluster is increased from 8,000 to 10,000, and the number of VM DirectPath I/O devices per host is increased from 8 to 32.

- Lifecycle Management improvements including VMware vSphere Configuration Profiles as a new alternative to VMware Host Profiles, staging cluster images and remediating up to 10 ESXi hosts in parallel instead of one at a time.
- New Virtual Hardware in VM hardware version 20 supporting the latest guest operating systems, including Windows 11.
- Distributed Resource Scheduler and vMotion improvements.
- Implementation of the VMware Balanced Power Management Policy on each server, which reduces energy consumption with minimal performance compromise.

For more information about VMware vSphere and its components, go to: https://www.vmware.com/products/vsphere.html

Microsoft Windows Server 2022

Windows Server 2022 is the latest OS platform release from Microsoft. Windows Server 2022 is an excellent platform for running Microsoft SQL Server 2022 databases. It offers new features and enhancements related to security, patching, domains, clusters, storage, and support for various new hardware features, and so on. It enables Windows Server to provide best-in-class performance and a highly scalable platform for deploying SQL Server databases.

Microsoft SQL Server 2022

SQL Server 2022 (16.x) is the latest relational database from Microsoft and builds on previous releases to grow SQL Server as a platform that gives you choices of development languages, data types, on-premises or cloud environments, and operating systems. It offers various enhancements and new features that enables SQL Server deployments more reliable, highly available, performant, and secured than ever.

SQL Server 2022 can leverage new hardware capabilities from partners like Intel to provide extended capabilities. It can leverage Intel Quick Assist Technology (QAT) for offloading backup compression thereby improving backups and restores performance; Intel Advanced Vector Extension-512 can be leveraged by SQL Server 2022 engine to improve batch mode operations.

For more details about the new capabilities of SQL Server 2022, refer to: https://learn.microsoft.com/en-us/sql/sql-server/what-s-new-in-sql-server-2022?view=sql-server-ver16

SQL Server Management Studio (SSMS)

SQL Server Management Studio (SSMS) is an integrated environment for managing any SQL infrastructure. SSMS is widely used to access, configure, manage, administer, and develop all components of SQL Server, Azure SQL Database, Azure SQL Managed Instance, SQL Server on Azure VM, and Azure Synapse Analytics. SSMS provides a single comprehensive utility that combines a broad group of graphical tools with many rich script editors to provide access to SQL Server for developers and database administrators of all skill levels.

For more information about SQL Server Management Studio, refer to the following link:

https://learn.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver16

Hardware and Software Revisions

This solution can be extended to any FlexPod environment running supported software, firmware, and hardware versions as defined in the NetApp Interoperability Matrix Tool, UCS Hardware and Software Compatibility, and VMware Compatibility Guide.

The following table shows the FlexPod hardware and software revisions used in this solution.

Table 1 Hardware and Software versions

Component	Product	Version
Compute	Cisco UCSX-210C-M7 Compute Node	5.2(0.230092)
	Cisco UCSX-I-9108-25G IFM	4.3(2c)
	Cisco UCS Fabric Interconnects 6536	4.3(2.240002)
	Cisco UCS VIC 15231	5.3(2.40)
	CPU	Intel(R) Xeon(R) Platinum 8462Y+ CPU @ 2.8GHz
Network	Cisco Nexus 9336C-FX2 NX-OS	10.2(6)
Storage	NetApp AFF A400	ONTAP 9.14.1P4
	VMware vSphere	8.0.2
	VMware vCenter Appliance	8.0.2
	VMware ESXi nenic Ethernet driver	1.0.45.0
	Cisco Intersight Assist Virtual Appliance	1.0.11-4206
	Microsoft Windows Server	2022
	NetApp Host Utilities for Windows	7.2
	Microsoft SQL Server	2022 (16.0.1000.6)
	SQL Server Management Studio	20.1

Installation and Configuration

This section describes the set of steps that were followed to prepare the solution infrastructure for validation.

FlexPod Setup

Refer the following document to understand the solution design details and deploy the base FlexPod.

FlexPod Datacenter for Microsoft SQL Server 2022 and VMware vSphere 8.0

This document describes a FlexPod reference architecture and step-by-step implementation guidelines for deploying FlexPod infrastructure for Microsoft SQL Server 2022 databases.

FlexPod deployment can be automated with Infrastructure as Code (IaC) using Ansible. Below are the links to the IaC document and GitHub repository respectively for End-to-End FlexPod deployment:

- FlexPod Datacenter using IaC with Cisco IMM M7, VMware vSphere 8, and NetApp ONTAP 9.12.1
- Ansible configuration of FlexPod with Cisco UCS in IMM, NetApp ONTAP, and VMware vSphere

VLAN Configuration

Table 2 lists the VLANs configured for setting up the IP-based FlexPod environment along with their usage.

Table 2 Configured VLANs and their usage

Name	VLAN ID	Usage
Native-VLAN	2	VLAN 2 used as native VLAN instead of default VLAN (1)
OOB-MGMT-VLAN	2254	Out-of-band management VLAN to connect management port for various devices
SQL-MGMT-VLAN	2255	In-band management VLAN utilized for all in-band management connectivity – for example, ESXi hosts, VM management, and so on
SQL-NFS-VLAN	2256	NFS VLAN for mounting datastores in ESXi servers for VM's OS disks
SQL-iSCSI-A-VLAN	2258	iSCSI-A path for boot-from-san traffic as well as SQL VM storage traffic
SQL-iSCSI-B-VLAN	2259	iSCSI-B path for boot-from-san traffic as well as SQL VM storage traffic
SQL-vMotion-VLAN	2260	VMware vMotion traffic VLAN

ONTAP Storage Layout

In this solution, there are distributed SQL Server data and log volumes on two aggregates equally, to balance the performance and capacity utilization. Total 4 data and 4 log volumes have been created and distributed equally on both the aggregates. As for Windows Failover Clustering shared storage is necessary, so we will map both the data and log LUNs to the Windows VMs.

The following figure shows a detailed ONTAP storage layout scenario for the SQL Server database and log volumes. For solution validation, we created total 4 data LUNs, each of size 500 GB, and 4 log LUNs, each of size 200 GB.

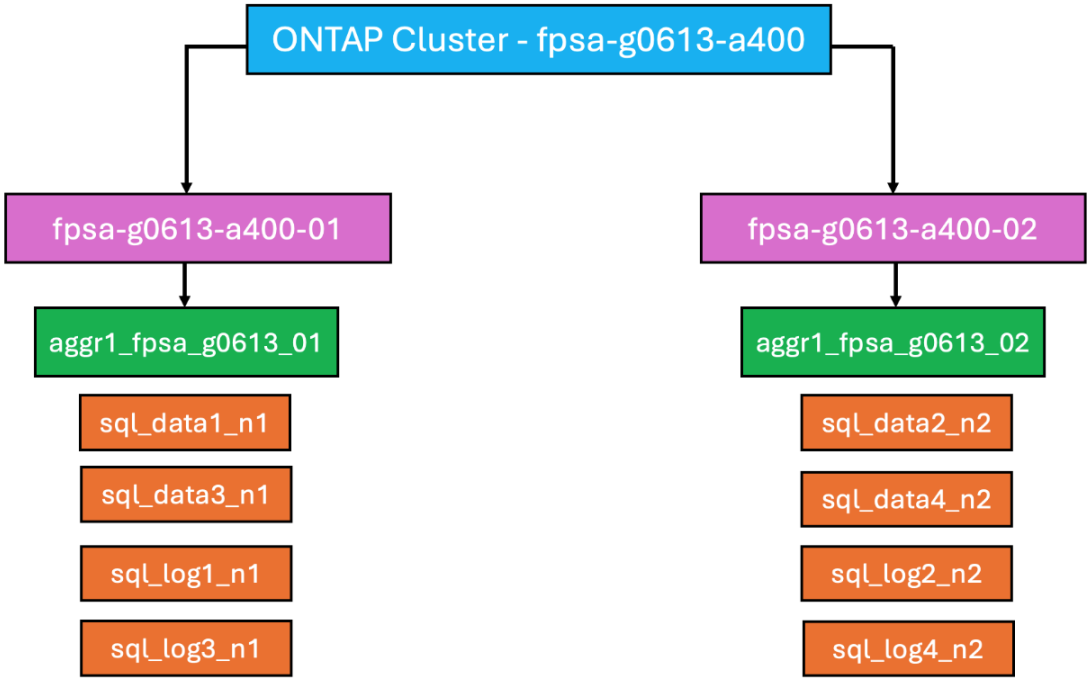


Figure 11 Storage layout for Microsoft SQL Server database data and log volumes

Cisco UCS Configuration using Cisco Intersight

This section provides more details on the specific Cisco UCS policies and settings used for configuring Cisco UCS X210c M7 compute nodes for hosting critical Microsoft SQL Server database virtual machines on vSphere ESXi environment. The Cisco UCS X210c M7 blades are installed in the Cisco UCS X9508 chassis and are connected to a pair of Cisco UCS 6536 Fabric Interconnects. The Cisco UCS fabric interconnects are managed by Intersight.

It is important to use the correct network and storage adapter policies for low latency and better storage bandwidth, since the underlying Cisco VIC resources are shared between various types of traffic such as virtual machine management traffic, storage access, ESXi host management, vMotion, and so on.

LAN Connectivity Policies

The following vNICs are defined in a LAN connectivity policy to derive the vNICs for the ESXi host networking. The LAN connectivity policy is then used in Server profiles template to derive the Server profile.

- 00-vSwitch0-A: Used for ESXi host management traffic over Fabric A
- 01-vSwitch0-B: Used for ESXi host management traffic over Fabric B
- 02-vDS0-A: Used for infrastructure management traffic such as vMotion, NFS storage access, and SQL Server virtual machine management traffic over Fabric A

- 03-vDS0-B: Used for infrastructure management traffic such as vMotion, NFS storage access, and SQL Server virtual machine management traffic over Fabric B
- 04-iSCSI-A: Used for booting the ESXi hosts from the NetApp storage array boot LUNs and direct NetApp storage access by SQL Server Guest Windows VMs using in-guest iSCSI protocol over Fabric A
- 05-iSCSI-B: Used for booting the ESXi hosts from the NetApp storage array boot LUNs and direct NetApp storage access by SQL Server Guest Windows VMs using in-guest iSCSI protocol over Fabric B

Table 3 lists the additional configuration details of the above vNICs used in this reference architecture.

Table 3 vNIC Settings

vNICs	00-vSwitch0-A	01-vSwitch0-B	02-vDS0-A	03-vDS0-B	04-iSCSI-A	05-iSCSI-B
Slot ID and PCI Link	Auto	Auto	Auto	Auto	Auto	Auto
PCI Order	0	1	2	3	4	5
Switch ID	A	B	A	B	A	B
Fabric Failover	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled
Network Group Policy for list of Allowed VLANs and Native VLAN	2254, 2255 and 2255	2254, 2255 and 2255	2255, 2256, 2260 and 2	2255, 2256, 2260 and 2	2258 and 2258	2259 and 2259
Network Control Policy for CDP and LLDP	CDP and LLDP Enabled	CDP and LLDP Enabled	CDP and LLDP Enabled	CDP and LLDP Enabled	CDP and LLDP Enabled	CDP and LLDP Enabled
QoS & MTU	Best Effort and 9000	Best Effort and 9000	Best Effort and 9000	Best Effort and 9000	Best Effort and 9000	Best Effort and 9000
Ethernet Adapter Policy	EthAdapter-VMware-Policy	EthAdapter-VMware-Policy	EthAdapter-16RXQs	EthAdapter-16RXQs	EthAdapter-16RXQs	EthAdapter-16RXQs

The adapter policy allows the administrator to declare the capabilities of the vNIC, such as the number of rings, ring sizes, and offload enablement and disablement. The transmit and receive queues defined in the default VMware policy may not be sufficient as more SQL Server databases are consolidated which would generate lot of VM management, NFS as well as iSCSI storage traffic on the FlexPod system.

As shown in the above table, the Ethernet Adapter Policy EthAdapter-16RXQs, is used for the vNICs that carry SQL VM management, NFS and iSCSI traffic.

The remaining policies and configuration steps for Cisco UCS are the same as the ones explained in the base infrastructure CVD described here:

https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_ucs_xseries_e2e_ontap_manual_deploy.html

Windows Virtual Machine Configuration

This section provides configuration recommendations for the Windows guest operating system for hosting SQL Server databases. For a detailed step-by-step process for installing the Windows Server 2022 guest operating system in the virtual machine, refer to the VMware documentation. In this solution, two Windows Virtual Machines (VMs) have been deployed in VMware vCenter, SQL-VM1 and SQL-VM2.

When the Windows guest operating system is installed in the virtual machine, you should also install the VMware tools as explained here.

Memory Reservation

SQL Server database transactions are usually CPU and memory intensive. In heavily OLTP database systems, you should reserve all the memory allocated to the SQL Server virtual machines as shown in Figure 12. This approach helps ensure that the memory assigned to the SQL Server virtual machines is committed, and it eliminates the possibility that ballooning and swapping will occur.

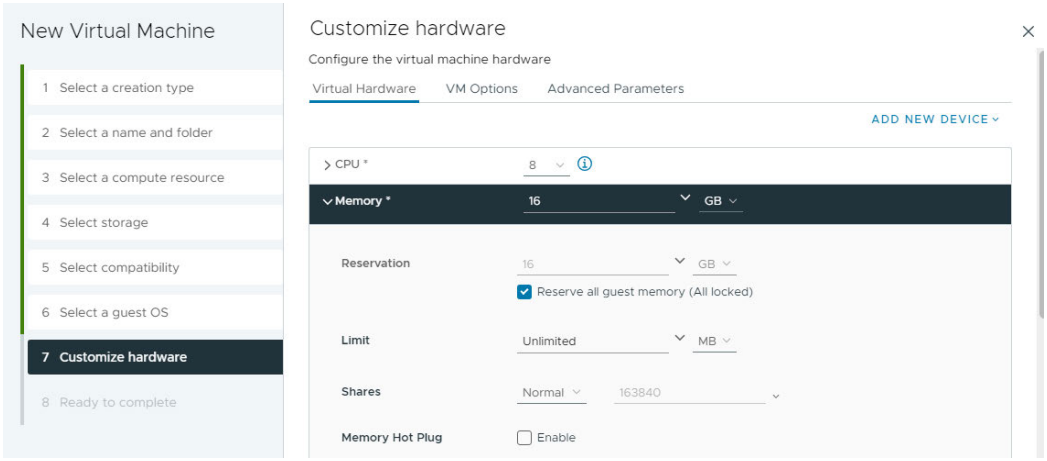


Figure 12 Memory Allocation for SQL Virtual Machine

Network Adapter Type

The network adapter of type VMXNET3 should be used for SQL Server virtual machine. VMXNET 3 is the latest generation of para virtualized NICs designed for performance. It offers several advanced features, including multiple-queue support, receive-side scaling, IPv4/IPv6 offloads, and message-signaled interrupt (MSI) and MSI-X interrupt delivery.

In this solution, each Windows VM is configured with three network adapters, with VMXNET3 as the adapter type. One adapter is connected to the IB-MGMT port group for virtual machine management and SQL Server access, and the second and third network adapters are connected to the SQL-iSCSI-A and SQL-iSCSI-B port groups respectively. These adapters are used for direct NetApp storage access using

the Microsoft software iSCSI initiator over Fabrics A and B respectively. Figure 13 shows the SQL Server virtual machine configured with three adapters.

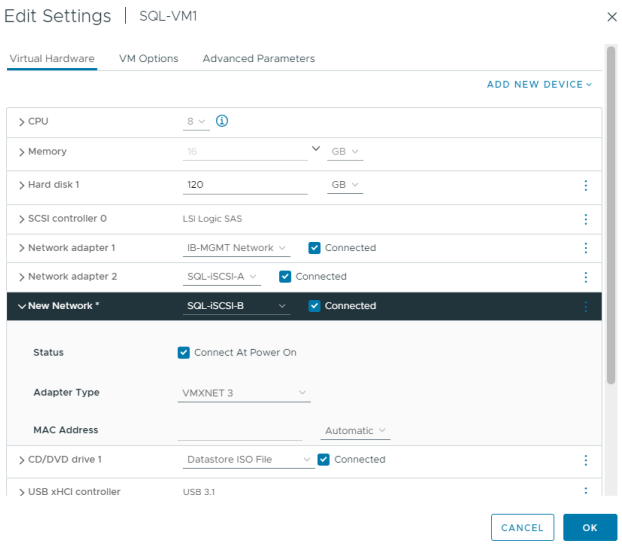


Figure 13 SQL Virtual Machine Network Configuration

Guest Power Settings

The default power policy option in Windows Server 2022 is Balanced. For SQL Server database deployments, you should set the power management option to High Performance for optimal database performance.

Add Guest Virtual Machine to the Domain

You should change the default Windows guest virtual machine name and join the virtual machine to the domain before you proceed with the storage configuration for the guest virtual machine. For detailed instructions about how to change the guest name and join the guest, click here.

Using the server manager, enable the **Remote Desktop** feature to remotely manage the guest virtual machine and turn off the firewalls in the guest virtual machine.

Figure 14 shows the final configuration of a sample VM (SQL-VM1) after it has joined to the fpmc.sa domain, enabling Remote Desktop and turning off the firewall settings, and corresponding IP addresses of the management and iSCSI storage interfaces IP addresses.

Note: The storage adapters which were created from SQL-iSCSI-A and SQL-iSCSI-B port groups have been changed to iSCSI-A and iSCSI-B.

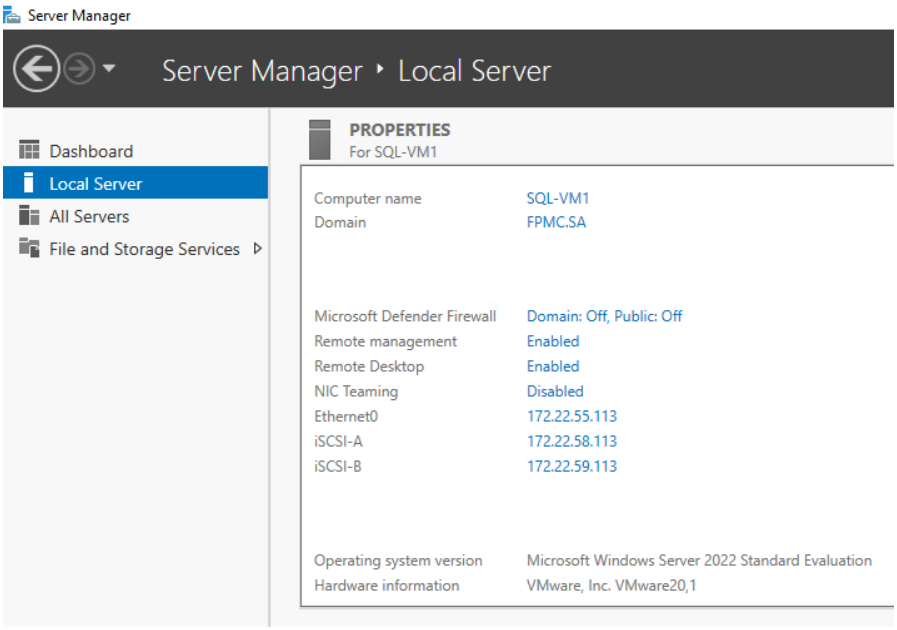


Figure 14 SQL VM Configuration

Storage Configuration in the Windows Virtual Machine

Enable Jumbo Frames on Storage Network Interfaces

Enabling jumbo frames for storage traffic provides better I/O performance for SQL Server databases. In the SQL Server guest virtual machine, make sure that jumbo frames are set to **9000** on the Ethernet adapter used for NetApp storage connectivity.

Configure Multipath Software

NetApp recommends using Windows native multipath drivers to manage storage connections in the Windows Server 2022 guest VM.

Figure 15 shows the installation of the multipath I/O feature using PowerShell. After installing this feature, enable Microsoft Device Specific Module (MSDSM) to automatically claim SAN disks for Microsoft Multipath I/O (MPIO) for the iSCSI bus type.

```
PS C:\Users\Administrator> Install-WindowsFeature -Name multipath-io

Success Restart Needed Exit Code      Feature Result
------- -------------- ---------      --------------
True    No             Success        {Multipath I/O}


PS C:\Users\Administrator> Enable-MSDSMAutomaticClaim -BusType iSCSI

VendorId ProductId
-------- ---------
MSFT2005 iSCSIBusType_0x9
False
```

Figure 15 Installing Windows MPIO Drivers

Install the NetApp Windows Unified Host Utilities on the VM

Download NetApp Host Utilities version 7.2 for Windows from this link:

https://mysupport.netapp.com/site/products/all/details/hostutilities/downloads-tab/download/61343/7.2/downloads

Unzip the file and run the executable file. NetApp Windows Unified Host Utilities setup wizard is launched. Make sure to select install support for Multipath I/O and complete the installation.

Verify that appropriate device drives are added in the MPIO utility as shown below.

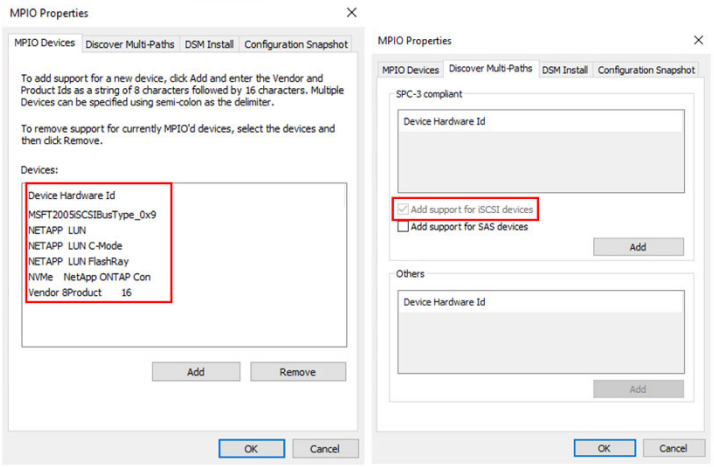


Figure 16 MPIO Properties in Windows VM

Configure iSCSI Software Initiator

This procedure provides the steps for configuring the in-guest iSCSI initiator for Windows VMs to directly access the NetApp storage LUNs.

- Start the **Microsoft iSCSI initiator service** and set it to start automatically as shown in following diagram. Find the VM initiator ID and make a note of it, as you will need it to grant the NetApp storage LUN access to the guest VM.

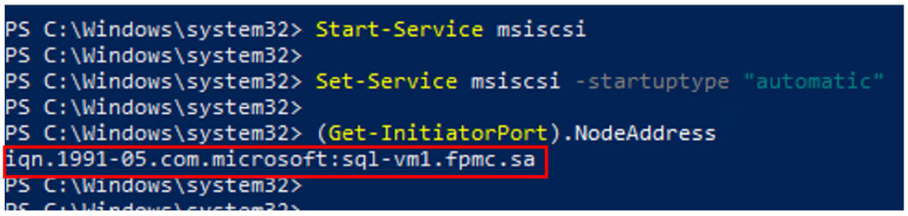


Figure 17 Starting Microsoft iSCSI initiator service on Windows VM

- On NetApp storage end, create an initiator group using the iSCSI initiator name previously noted. In this solution, one igroup is created that contains the IQNs of both the Windows VMs. Then the required data and log LUNs are mapped to this igroup.

- On Windows VM, open Server Manager, and then navigate to iSCSI Initiator. Under Target, provide the Target Name of storage SVM, and then click on Connect. Get the target name of SVM using “vserver iscsi show” command.
- Under Discovery tab, add NetApp target portals. This establishes connection to the NetApp target iSCSI IP addresses.
- Open Disk Management and initialize and format the disks with the NTFS file system and a 64-KB allocation unit size. Perform this step for all data and log disks.

Windows Failover Cluster Configuration

This section covers the details about Windows Failover Cluster setup.

Prerequisites

Before you begin, verify the following prerequisites:

- Make sure that all servers that you want to add as cluster nodes are running the same version of Windows Server. In this solution, both VMs are running on version Windows Server 2022.
- Review the hardware requirements to make sure that your configuration is supported. For more information, see Failover Clustering Hardware Requirements and Storage Options.
- To add clustered storage during cluster creation, make sure that all servers can access the storage. (You can also add clustered storage after you create the cluster). Here, we have configured clustered storage after cluster creation.
- Make sure that all servers that you want to add as cluster nodes are joined to the same Active Directory domain.
- (Optional) Create an organizational unit (OU) and move the computer accounts for the servers that you want to add as cluster nodes into the OU. As a best practice, its recommended that you place failover clusters in their own OU in Active Directory Domain Services (AD DS).
- Make sure that the account you want to use to create the cluster is a domain user who has administrator rights on all servers that you want to add as cluster nodes.

Install the Failover Clustering Feature

You must install the Failover Clustering feature on every server that you want to add as a failover cluster node. A server participating in Windows Server Failover Clustering (WSFC) is called node.

- On Windows VM (here SQL-VM1), start Server Manager. On the Manage Menu, select **Add Roles and Features**.
- On the Before you begin page, click Next.
- On the Select installation type page, select Role-based or feature-based installation and click Next.
- On the Select destination server page, select the server where you want to install the feature and then click Next.
- On the Select server roles page, click Next.
- On the Select features page, select the **Failover Clustering** check box.

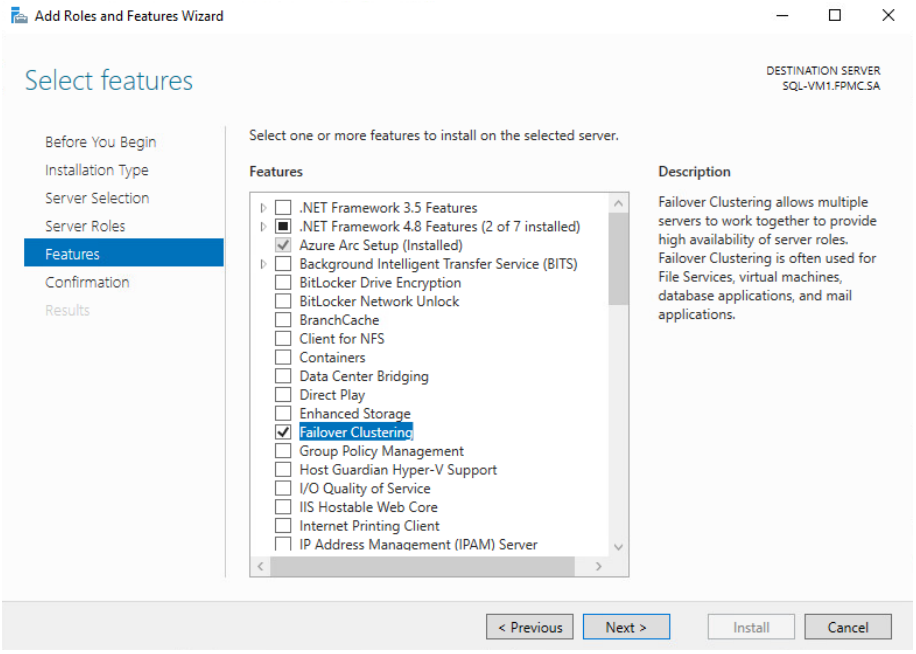


Figure 18 Installing Failover Clustering Feature on Windows VM

- To install the failover cluster management tools, select Add Features and then click Next.
- On the Confirm installation selections page, select Install. A server restart is not required for the Failover Clustering feature.
- When the installation is completed, click Close.
- Repeat this procedure on every server (here SQL-VM2) that you want to add as a failover cluster node.

Run Cluster Validation Tests

Before creating the failover cluster, it is strongly recommended that we validate the configuration to make sure the hardware and hardware settings are compatible with failover clustering. Microsoft supports a cluster solution only if the complete configuration passes all validation tests and if all hardware is certified for the version of Windows Server that the cluster nodes are running.

- On the Windows server where you installed the Failover Clustering feature, start Failover Cluster Manager. To do this on a server, start Server Manager and then on the Tools menu, select **Failover Cluster Manager**.
- In the Failover Cluster Manager pane under Management, select **Validate Configuration**.
- On the Before You Begin page, click Next.
- On the Select Servers or a Cluster page in the Enter name box, enter the NetBIOS name or the fully qualified domain name of a server that you plan to add as a failover cluster node and then click Add. Repeat this step for each server that you want to add. Then, click Next.

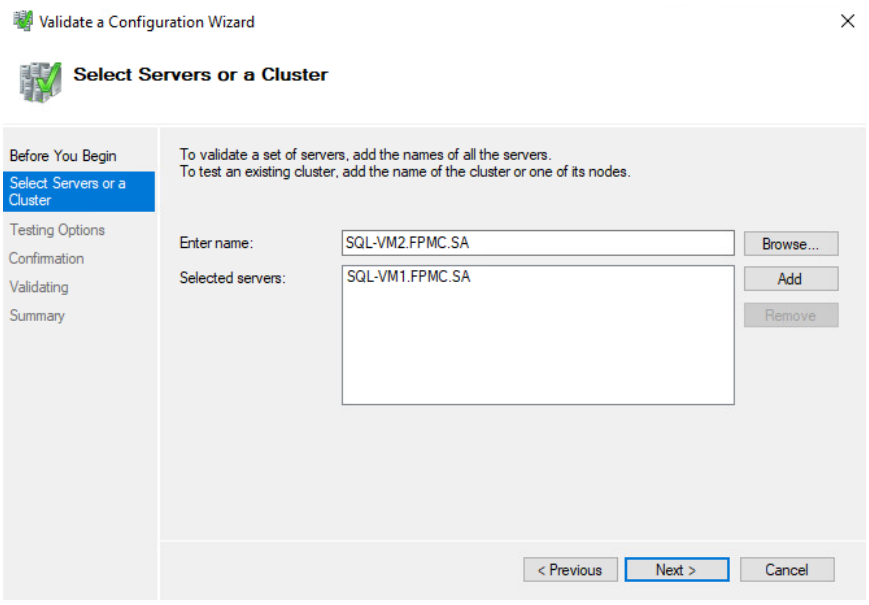


Figure 19 Select Servers for Cluster Validation Test

- On the Testing Options page, select Run all tests (recommended) and then click Next.
- On the Confirmation page, click Next. The Validating page displays the status of the running tests.

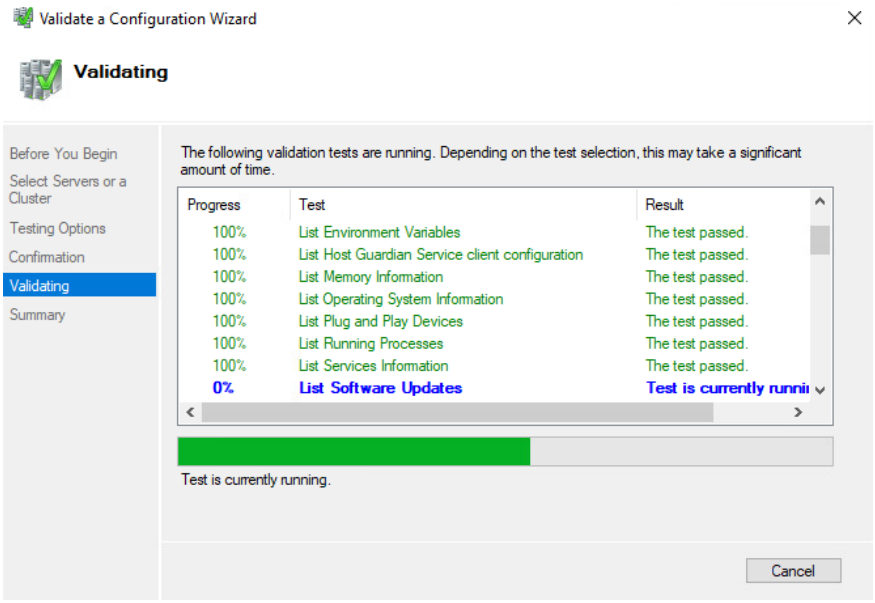


Figure 20 Cluster Validation Test in Progress

- On the Summary page, do either of the following:

- If the results indicate that the tests completed successfully, and the configuration is suited for clustering, then click Finish.
- If the results indicate that there were warnings or failures, select View Report to view the details and determine which issues must be corrected. Realize that a warning for a particular validation test indicates that this aspect of the failover cluster can be supported but might not meet the recommended best practices.

Create Windows Failover Cluster

Follow the below steps to configure Windows Failover Cluster.

- On the Windows VM (here, SQL-VM1), start Server Manager. On the Tools menu, select **Failover Cluster Manager**.
- In the Failover Cluster Manager pane, under Management pane, select **Create Cluster**. The Create Cluster Wizard opens.
- On the Before you begin page, click Next.
- On the **Select Servers** page, in the Enter name box enter the NetBIOS name or the fully qualified domain name of a server that you plan to add as a failover cluster node and then click Add. Repeat this step for each server that you want to add. Then, click Next.

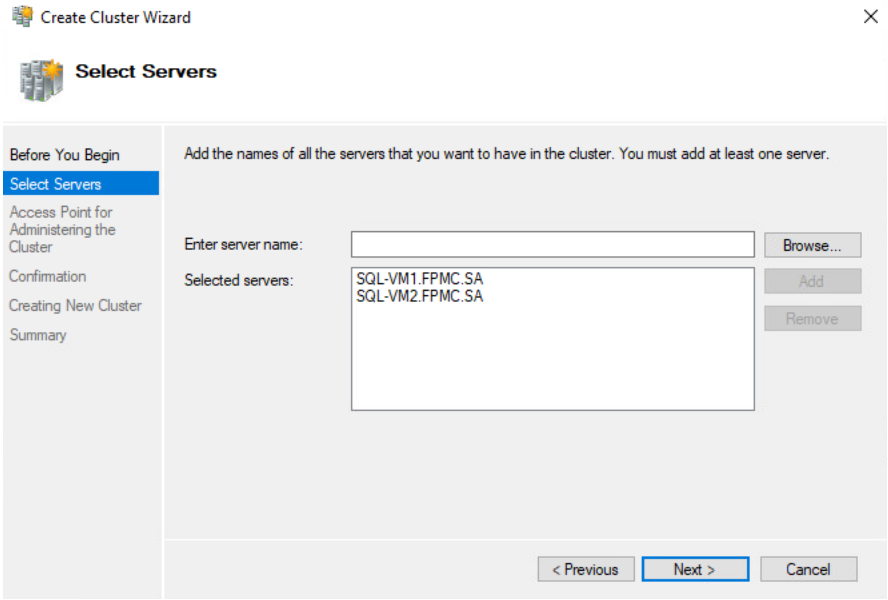


Figure 21 Servers Added for creating Windows failover cluster

- On the **Access Point for Administering the Cluster** page, do the following:
  - In the **Cluster Name** box, enter the name that you want to use to administer the cluster. Before you do, review the following information:
    - During cluster creation, this name is registered as the cluster computer object (also known as the *cluster name object* or *CNO*) in AD DS. If you specify the NetBIOS name for the cluster, the CNO is created in the same location where the

computer objects for the cluster nodes reside. This can be either the default Computers containers or an OU.

- To specify a different location for the CNO, you can enter the distinguished name of an OU in the Cluster Name box.
  For example: *CN=ClusterName, DC=Contoso, DC=com*.

- If the server does not have a network adapter that is configured to use DHCP, you must configure one or more static IP addresses for the failover cluster. Select the check box next to each network that you want to use for cluster management. Select the **Address** field next to a selected network and then enter the IP address that you want to assign to the cluster. This IP address (or addresses) will be associated with the cluster name in Domain Name System (DNS).
- Click Next.

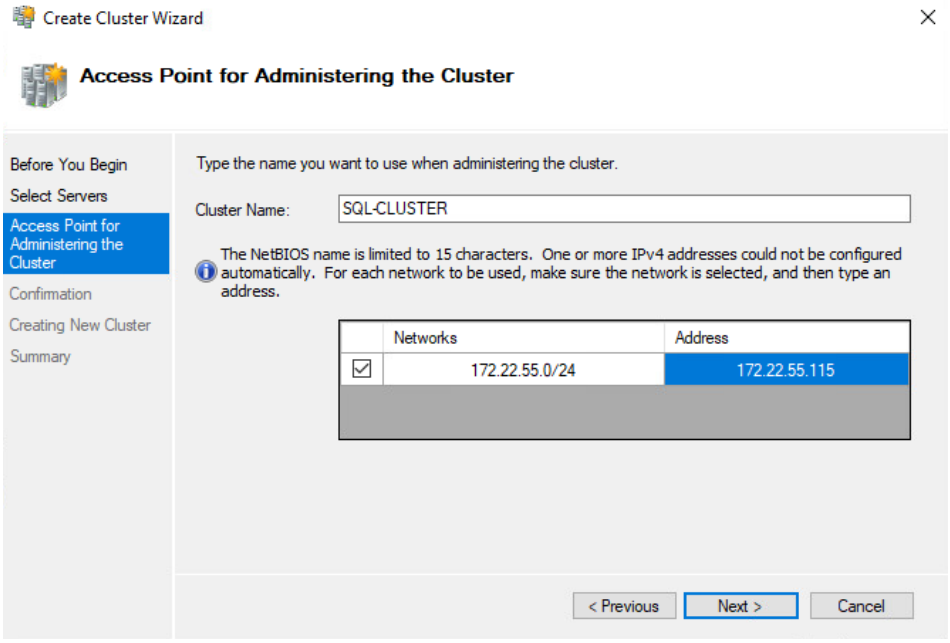


Figure 22 Access Point for Administering the Cluster

- On the **Confirmation page**, review the settings. By default, the Add all eligible storage to the cluster check box is selected. Clear this check box if you want to do either of the following:
  - You want to configure storage later. In this solution, we have configured storage later.
  - You plan to create clustered storage spaces through Failover Cluster Manager or through the Failover Clustering Windows Powershell cmdlets and have not yet created storage spaces in File and Storage Services.
- Click Next to create the Failover Cluster.
- On the **Summary** page, confirm that the failover cluster was successfully created. If there were any warnings or errors, view the summary output or select **View Report** to view the full report. Click Finish.

- To confirm that the cluster was created, verify that the cluster name is listed under **Failover Cluster Manager** in the navigation tree. You can expand the cluster name and then select items under **Nodes**, **Storage**, or **Networks** to view the associated resources.

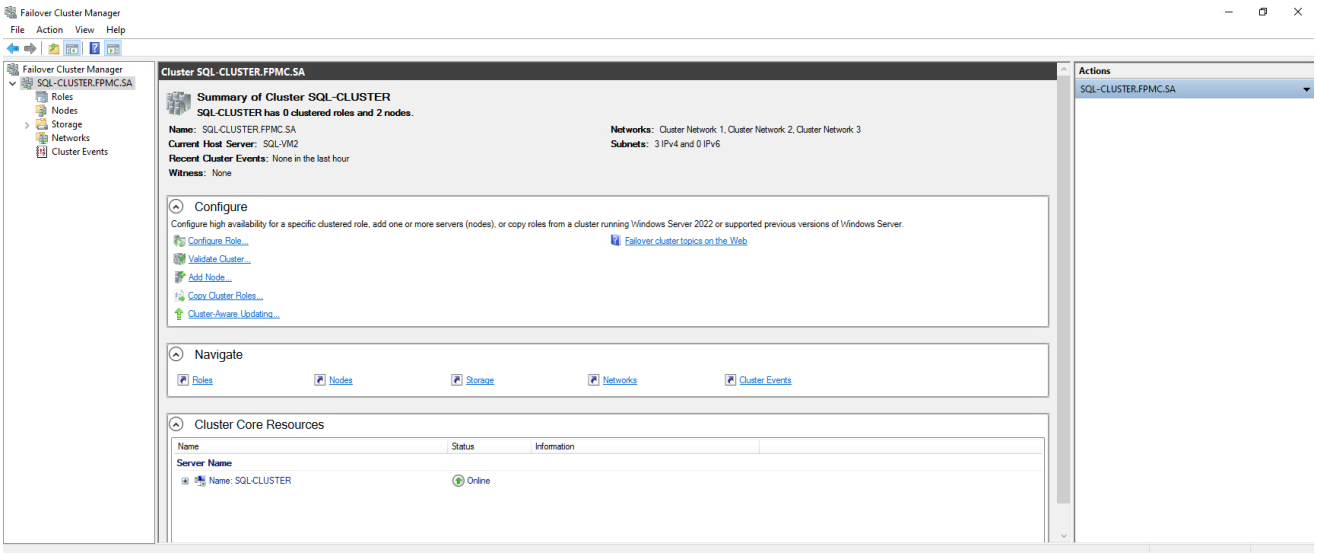


Figure 23 Windows Failover Cluster created successfully

Note that it may take some time for the cluster name to successfully replicate in DNS. After successful DNS registration and replication, if you select All Servers in Server Manager, the cluster name should be listed as a server with a Manageability status of Online.

Cluster Quorum Configuration

Windows Server Failover Clustering provides high availability for workloads running on Windows Server clusters. These resources are considered highly available if the nodes that host resources are up; however, the cluster generally requires more than half the nodes to be running, which is known as having *quorum*.

Quorum is designed to prevent split-brain scenarios that can happen when there's a partition in the network and subsets of nodes can't communicate with each other. This can cause both subsets of nodes to try to own the workload and write to the same disk, which can lead to numerous problems. However, this is prevented with Failover Clustering's concept of quorum, which forces only one of these groups of nodes to continue running, so only one of these groups stays online.

The quorum for a cluster is determined by the number of voting elements that must be part of active cluster membership for that cluster to start properly or continue running. For more detailed explanation, see the Understanding quorum.

Follow the below steps to configure Windows Failover Cluster Quorum settings:

- In **Failover Cluster Manager**, select the cluster. Under **Actions**, select **More Actions**, and then select **Configure Cluster Quorum Settings**. The Configure Cluster Quorum Wizard appears. Click Next.
- On the **Select Quorum Configuration Option** page, select one of the three configuration options and complete the steps for that option.

- In this solution, we have selected the **Advanced quorum configuration** to configure quorum management settings and to add the quorum witness. For more information about the advanced quorum configuration settings, see Node vote assignment and Dynamic quorum management.

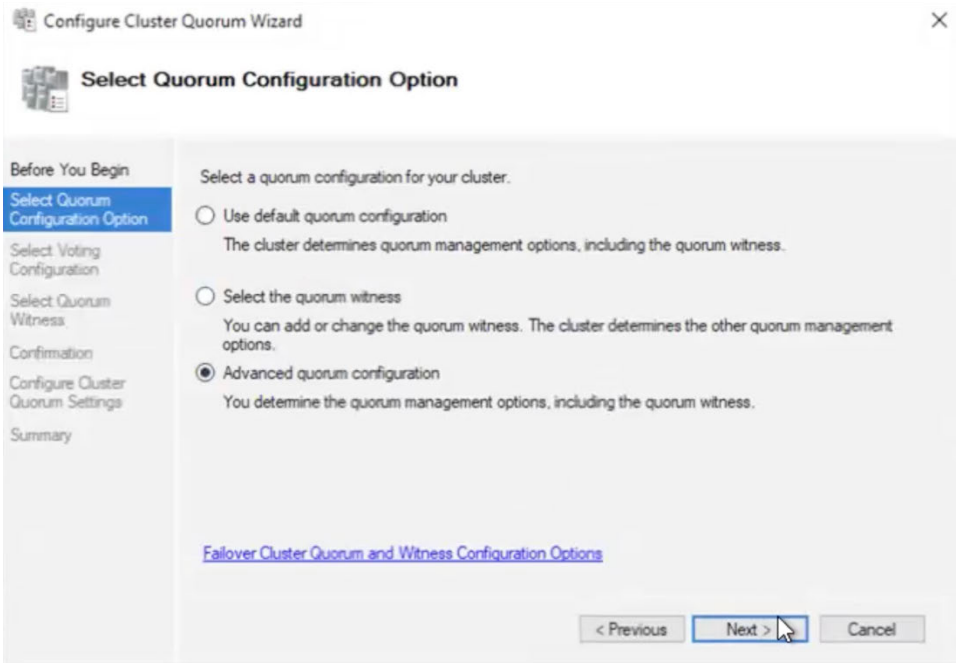


Figure 24 Select Quorum Configuration Option

- On the **Select Voting Configuration** page, select an option to assign votes to nodes. By default, all nodes are assigned a vote.
- On the **Configure Quorum Management** page, you can enable or disable the **Allow cluster to dynamically manage the assignment of node votes** option. By default, the option is enabled, and it is strongly recommended to not disable this option. This option allows the cluster to continue running in failure scenarios that is not possible when this option is disabled.
- On the **Select Quorum Witness** page, select an option to configure a disk witness, file share witness, or a cloud witness. In this solution, we have configured a **disk witness**.
- If you choose the option to configure a disk witness, on the **Configure Storage Witness** page, select the storage volume that you want to assign as the disk witness, and then complete the wizard.
  - Before this, we created a storage LUN “sql_quorum_lun” of size 4 GB on AFF A400 storage and then mapped it to both the VMs (SQL-VM1 and SQL-VM2). This disk was then used for configuring storage witness. Note that this step is necessary before you configure disk witness for cluster quorum configuration.
  - Figure 25 illustrates storage witness configuration.

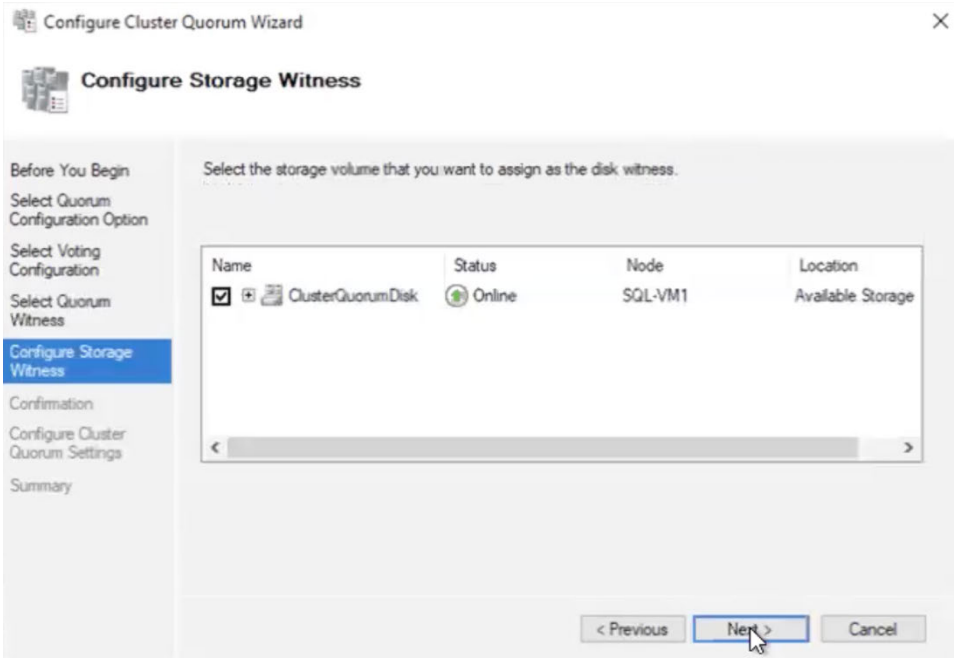


Figure 25 Configure Storage Witness

- Click Next. Confirm your selections on the confirmation page that appears and then click Next.
- After the wizard runs and the **Summary** page appears, if you want to view a report of the tasks that the wizard performed, select **View Report**. The most recent report will remain in the *systemroot***Cluster\Reports** folder with the name **QuorumConfiguration.mht**.
- After you configure the cluster quorum, it's recommended that you run the **Validate Quorum Configuration** test to verify the updated quorum settings.

Configure Clustered Storage Spaces

Storage Spaces, a technology in Windows and Windows Server that enables you to virtualize storage by grouping industry-standard disks into storage pools, and then creating virtual disks called *storage spaces* from the available capacity in the storage pools.

Prerequisites

- A minimum of three physical drives, with at least 4 GB capacity each, are required to create a storage pool in a Failover Cluster.
- The clustered storage pool must be comprised of multiple, separate disks configured at the hardware level. These disks are also known as logical unit numbers (LUNs).
- All physical disks used to create a clustered pool must pass the Failover Cluster validation tests.
- Clustered storage spaces must use fixed provisioning.

- Simple and mirror storage spaces are supported for use in Failover Cluster. Parity Spaces are not supported.
- The physical disks used for a clustered pool must be dedicated to the pool.

Steps to configure

- Open the **Failover Cluster Manager**. In the left-hand pane, expand **Storage**. Right-click on **Pools** and select **New Storage Pool**. This will start the **New Storage Pool Wizard**.
- Specify a **Name** for the Storage Pool and choose the **Storage Subsystem** that is available to the cluster and click Next. In this solution, we have created two storage pools, one for data and one for log. The following diagram shows specifying data pool name and subsystem.

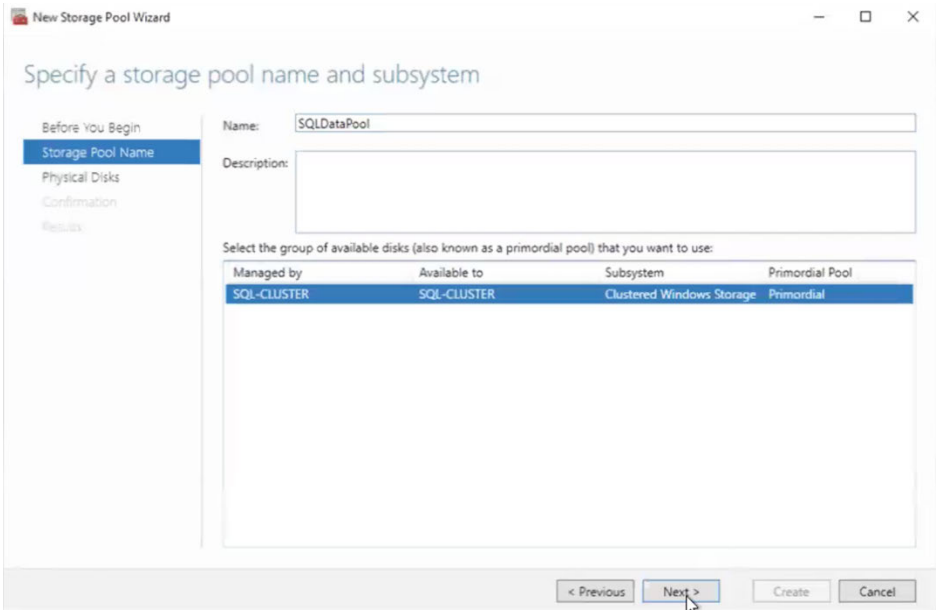


Figure 26 Specifying storage pool name and subsystem

- Select the **Physical Disks** for the storage pool and confirm the creation of the pool. Here, you need to select the data and log disks for data and log pools creation respectively. The following diagram shows selection of data disks for data pool creation. Total 4 data disks each of size 500 GB.

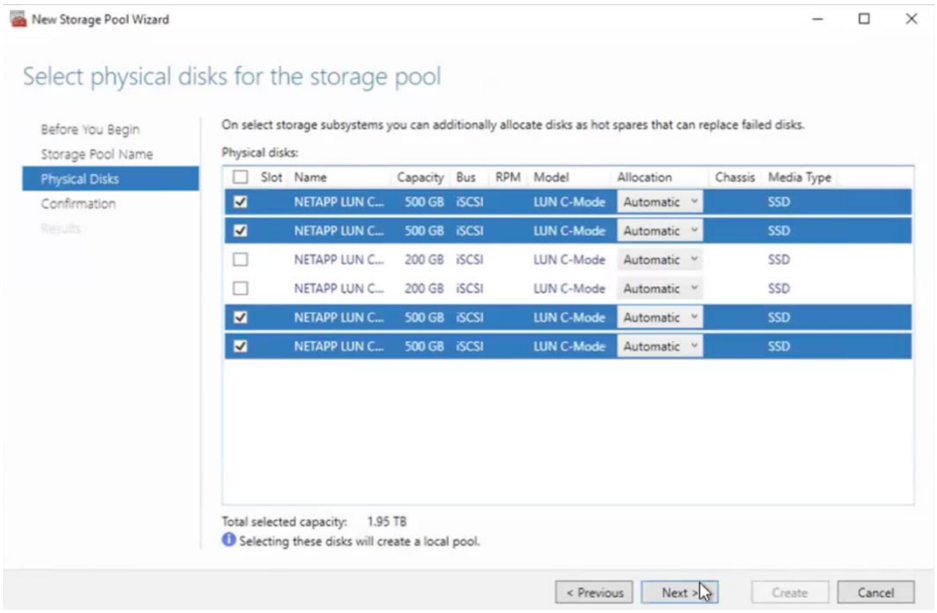


Figure 27 Selecting physical disks for the storage pool

The pool will be added to the cluster and brought Online, once created. The following diagram shows both data and log pools created.

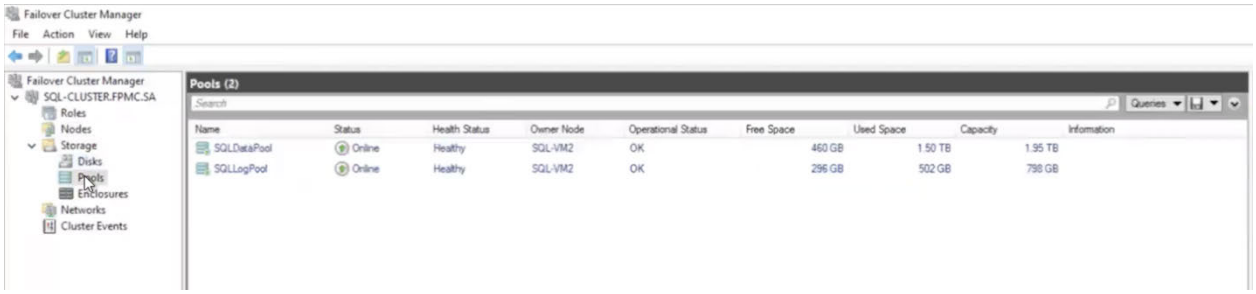


Figure 28 Storage pools created for data and log

- The next step is to create a Virtual Disk (storage space) that will be associated with a storage pool. In the **Failover Cluster Manager**, select the **storage pool** that will be supporting the Virtual Disk. Right-click and choose **New Virtual Disk**.
- This initiates the **New Virtual Disk Wizard**. Select the storage pool for the virtual disk and click Next. Here, we will create two virtual disks, one for data and one for log using their respective pools.

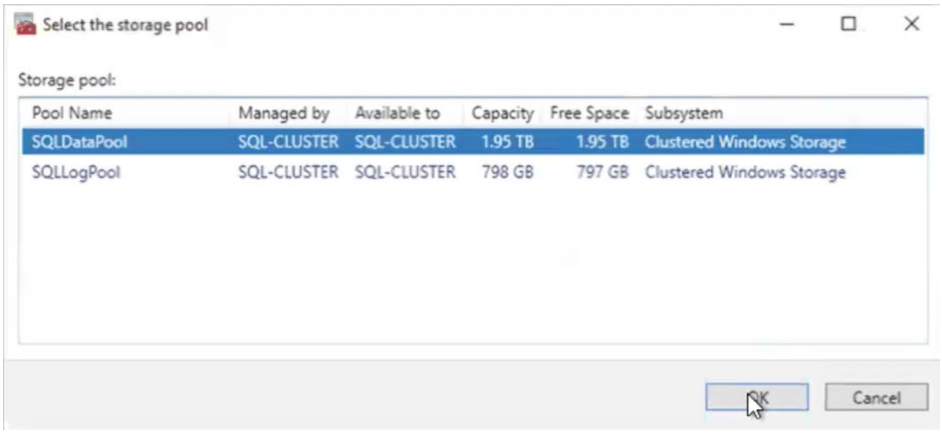


Figure 29 Select storage pool for Virtual Disk creation

- Provide a name and description for the virtual disk and click Next.
- Under the Enclosure Awareness tab, click Next.
- Specify the desired **Storage Layout** (Simple or Mirror) and click Next. In this solution, we selected Simple storage layout.
- Specify the size of the virtual disk and click Next. After you confirm your selection, the virtual disk is created.
- In this solution, two virtual disks (data and log) were created as shown in the following diagram.

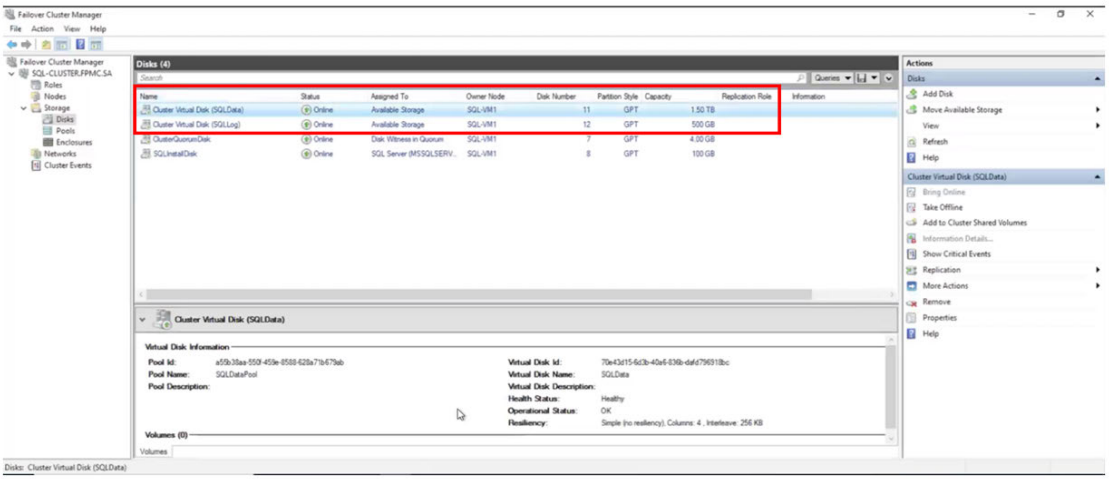


Figure 30 Virtual Disks created for data and log

- The **New Volume Wizard** is launched if you had checked the option “Create a volume when this wizard closes” on the Results page.
- The correct **Disk** and the **Server** to provision the disk should be selected. Click Next.

- Specify the size of the volume and click Next. In this solution, we will create two volumes, one for data and one for log with their respective virtual disks.
- Optionally assign a drive letter to the volume and click Next.
- Select the file system settings and click Next. The new volume will be created on the virtual disk and will be added to the Failover Cluster. Note that for data and log volumes, allocation unit size was set to 64K here. The NTFS File System should be selected.
- Note that this newly created data and log volume will be used during SQL Server failover cluster setup for data and log files storage.
- Your clustered storage space can now be used to host clustered workloads. You can also see the properties of the clustered storage space and the clustered pool that contains it, from the Failover Cluster Manager.

Configure SQL Server Failover Cluster Instance

SQL Server failover cluster instances use Windows Server Failover Clustering (WSFC) to provide high availability. A failover cluster instance (FCI) is redundant at the server-instance level. An FCI is a single instance of SQL Server that is installed across Windows Server cluster nodes and, possibly, across multiple subnets. On the network, an FCI appears as an instance of SQL Server running on a single computer, but the FCI provides failover from one WSFC node to another if the current node becomes unavailable.

Integrated Installation with Add Node

SQL Server integrated failover cluster installation consists of the following steps:

- Create and configure a single-node SQL Server failover cluster instance. When you configure the node successfully, you have a fully functional failover cluster instance. At this point, it does not have high availability because there is only one node in the failover cluster instance.
- On each node to be added to the SQL Server failover cluster instance, run Setup with Add Node functionality to add that node.

To install a new SQL Server failover cluster instance using Integrated Install with Add Node, follow the below steps:

- On the first node (here SQL-VM1) insert the SQL Server 2022 installation media, and from the root folder, double-click Setup.exe. To install from a network share, browse to the root folder on the share, and then double-click Setup.exe. For more information about how to install prerequisites, see Before Installing Failover Clustering.
- The Installation Wizard starts the SQL Server Installation Center. To create a new cluster installation of SQL Server, select **New SQL Server failover cluster installation** on the installation page.
- The System Configuration Checker runs a discovery operation on your system. To continue, click Next.
- On the Setup Support Files pages, click **Install** to install the Setup support files.
- The System Configuration Checker verifies the system state of your computer (node) before Setup continues. After check is complete, click **Next** to continue.

- On the Product key page, indicate whether you are installing a free edition of SQL Server, or whether you have a PID key for a production version of the product. For more information, see Editions and supported features of SQL Server 2022.
- On the License Terms page, read the license agreement, and then select the check box to accept the license terms and conditions. Click Next to continue.
- On the Feature Selection page, select the components for your installation. You can select any combination of check boxes, but only Database Engine, Analysis Services in tabular mode, and Analysis Services in multidimensional mode support failover clustering. For more information, see Determine the Server Mode of an Analysis Services Instance. In this solution, we have selected Database Engine Services as shown in the following diagram.

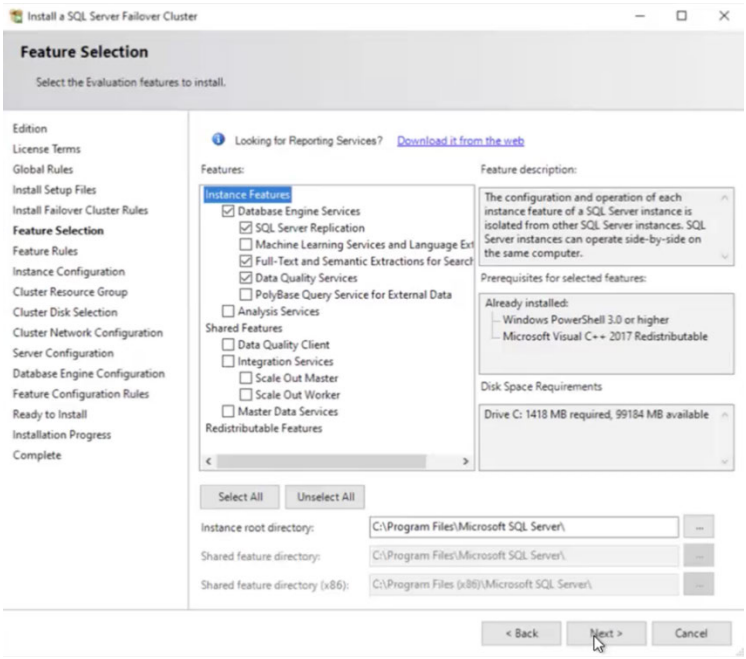


Figure 31 Feature Selection for SQL Server Failover Cluster

- SQL Server setup runs one more set of rules that are based on the features you selected to validate your configuration.
- On the Instance Configuration page, specify whether to install a default or a named instance. For more information, see Instance Configuration.
    - **SQL Server Network Name** - Specify a network name for the new SQL Server failover cluster instance. This is used to identify your failover cluster instance on the network. Here, we have given SQLSERVER-FCI as the SQL Server Network Name.
    - **Instance ID** - By default, the instance name is used as the Instance ID. This is used to identify installation directories and registry keys for your instance of SQL Server. For a default instance, the instance name and instance ID would be MSSQLSERVER, as in this solution.
    - **Instance root directory** - By default, the instance root directory is C:\Program Files\Microsoft SQL Server.

- **Detected SQL Server instances and features on this computer** – This grid shows instances of SQL Server that are on the computer where Setup is running. Click Next to continue.

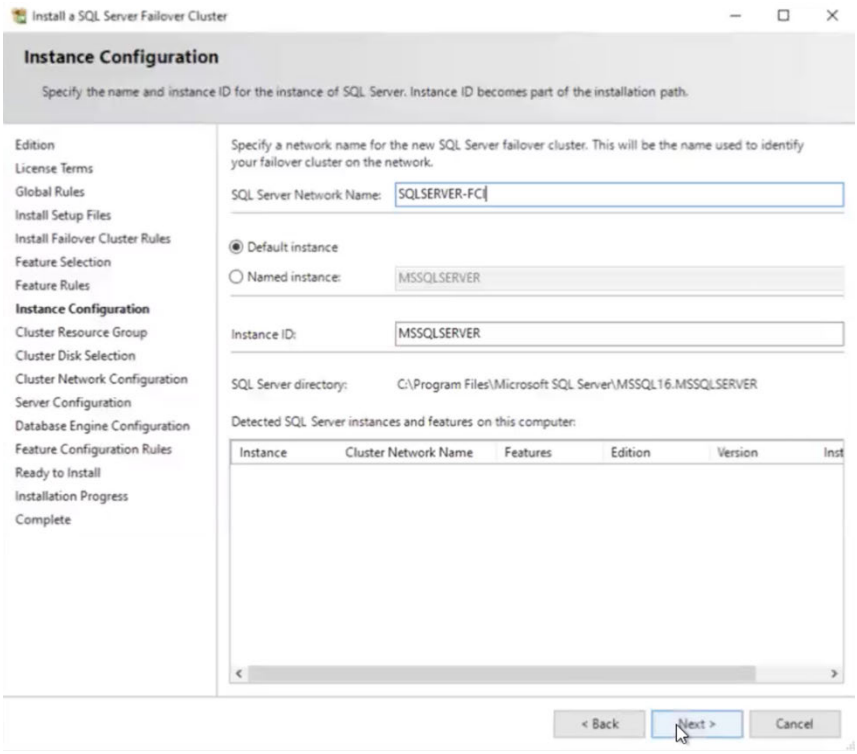


Figure 32 Instance Configuration for SQL Server Failover Cluster

- Use the Cluster Resource Group page to specify the cluster resource group, or role name, where SQL virtual server resources will be located.
- On the Cluster Disk Selection page, select the shared cluster disk resource for your SQL Server failover cluster instance. The cluster disk is where the SQL Server data will be put. Select the data and log virtual disks here.
    - More than one disk can be specified. The Available shared disks grid displays a list of available disks, whether each is qualified as a shared disk, and a description of each disk resource. Click Next to continue.

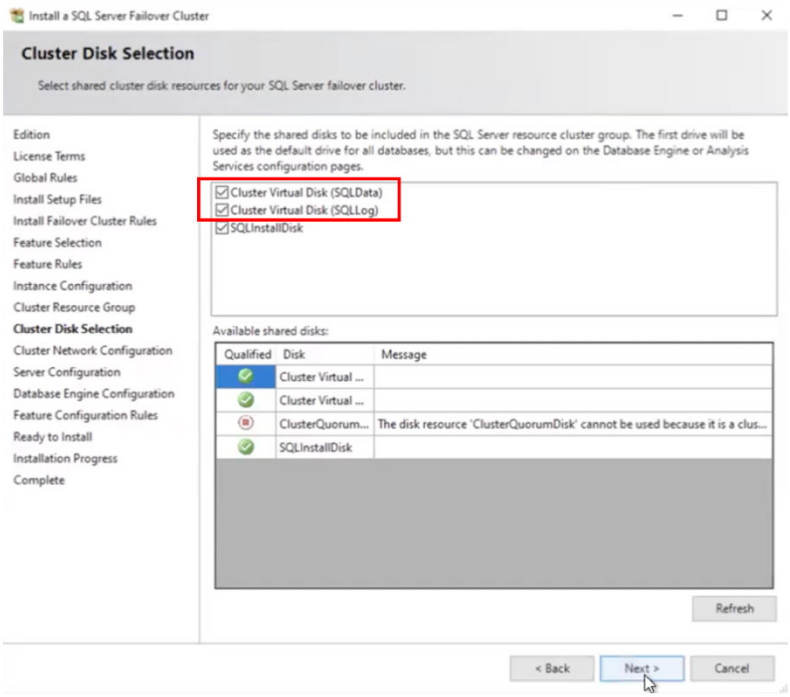


Figure 33 Cluster Disk Selection for SQL Server Failover Cluster

- On the Cluster Network Configuration page, specify the network resources for your failover cluster instance:
    - **Network Settings** - Specify the IP type and IP address for your failover cluster instance. Click Next to continue. Here, we have used static IP address.
- Use this page to specify Cluster Security Policy.
    - **Windows Server 2008 and later versions** – Service SIDs (server security IDs) are the recommended and default settings. Click Next to continue.
- Workflow for the rest of this configuration depends on the features that you have specified for your installation. You might not see all the pages, depending on your selections (Database Engine Services).
- On the Server Configuration - Service Accounts page, specify login accounts for SQL Server services. The actual services that are configured on this page depend on the features that you selected to install.
    - You can assign the same login account to all SQL Server services, or you can configure each service account individually. The startup type is set to manual for all cluster-aware services, including full-text search and SQL Server Agent, and cannot be changed during installation. For more information, see Server Configuration - Service Accounts and Configure Windows Service Accounts and Permissions.
    - To specify the same logon account for all service accounts in this instance of SQL Server, provide credentials in the fields at the bottom of the page. Then click Next.
- Use the Server Configuration - Collation tab to specify nondefault collations for the Database Engine and Analysis Services.

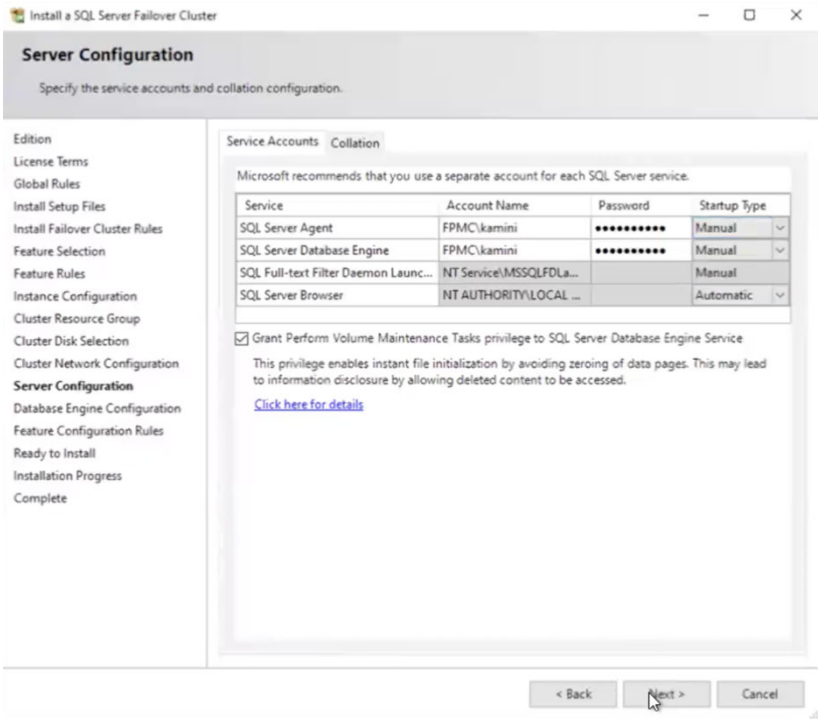


Figure 34 Server Configuration for SQL Server FCI

- Use the Database Engine Configuration - Server Configuration page to specify the following:
    - **Authentication Mode** - select Windows Authentication or Mixed Mode Authentication for your instance of SQL Server.
    - **SQL Server Administrators** - You must specify at least one system administrator for the SQL Server instance. To add the account under which SQL Server Setup is running, select **Add Current User**. To add or remove accounts from the list of system administrators, select **Add** or Remove, and then edit the list of users, groups, or computers that will have administrator privileges for the SQL Server instance. When the list is complete, click Next.
- Use the Database Engine Configuration - Data Directories page to specify nondefault installation directories. To install to default directories, click Next.
    - If you specify nondefault installation directories, make sure that the installation folders are unique to this SQL Server instance. None of the directories in this dialog box should be shared with directories from other SQL Server instances. The data and log directories should be located on the shared cluster disk for the failover cluster instance.
    - In this solution, we have used data and log volumes (created using Cluster Virtual Disk in previous section) for the SQL Server data and log directories.
    - In addition to the user database files, make sure to store the TempDB files also on the shared storage disks.
- Use the Database Engine Configuration - FILESTREAM page to enable FILESTREAM for your SQL Server instance. Click Next to continue.

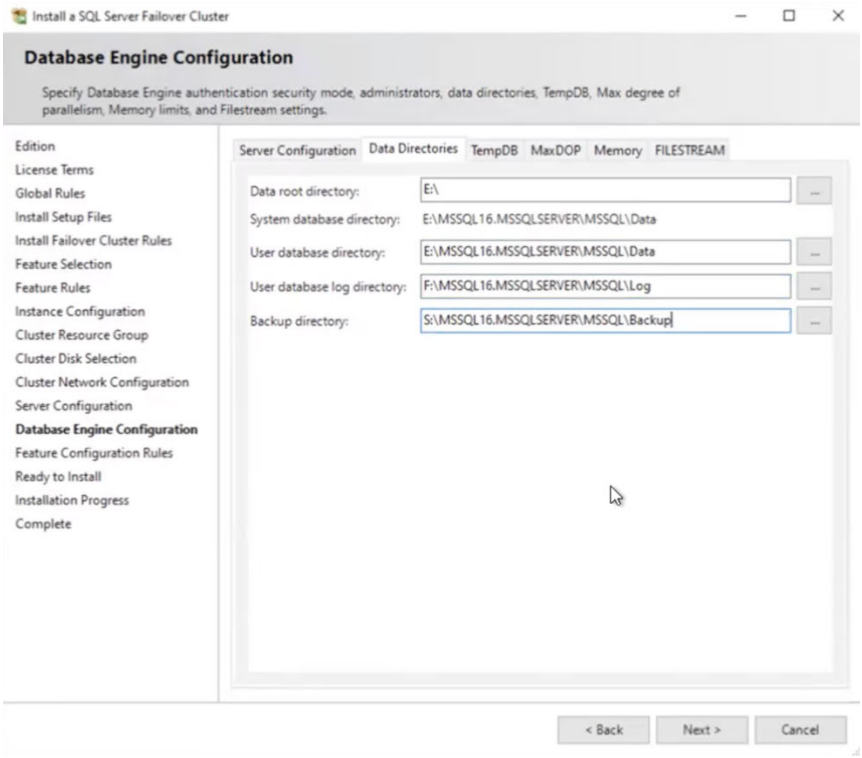


Figure 35 Database Engine Configuration for SQL Server FCI

- The System Configuration Checker runs one more set of rules to validate your configuration with the SQL Server features that you have specified.
- The Ready to Install page displays a tree view of installation options that were specified during Setup. To continue, select **Install**. Setup will first install the required prerequisites for the selected features followed by the feature installation.
- During installation, the Installation Progress page provides status so that you can monitor the installation progress as Setup continues.
- After installation, the **Complete** page provides a link to the summary log file for the installation and other important notes. To complete the SQL Server installation process, click **Close**.
- If you are instructed to restart the computer, do so now. It is important to read the message from the Installation Wizard when you have finished the Setup. For more information about Setup log files, see View and Read SQL Server Setup Log Files.

Now, we will add second node (SQL-VM2) to this SQL Server failover cluster instance. To add a node to a newly created SQL Server FCI, you must run SQL Server Setup on the node that is to be added to the SQL Server failover cluster instance. Do not run Setup on the active node.

- On second node (here SQL-VM2) insert the SQL Server 2022 installation media, and from the root folder, double-click Setup.exe. To install from a network share, navigate to the root folder on the share, and then double-click Setup.exe.

- The Installation Wizard will launch the SQL Server Installation Center. To add a node to an existing failover cluster instance, click **Installation** in the left-hand pane. Then, select **Add node to a SQL Server failover cluster**.
- The System Configuration Checker will run a discovery operation on your system. To continue, click OK.
- On the Product key page, specify the PID key for a production version of the product. Note that the product key you enter for this installation must be for the same SQL Server edition as that which is installed on the active node.
- On the License Terms page, read the license agreement, and then select the check box to accept the licensing terms and conditions. Click Next.
- The System Configuration Checker will verify the system state of your computer (here SQL-VM2) before Setup continues. After the check is complete, click Next.
- On the Cluster Node Configuration page, use the drop-down box to specify the name of the SQL Server failover cluster instance that will be modified during this Setup operation.

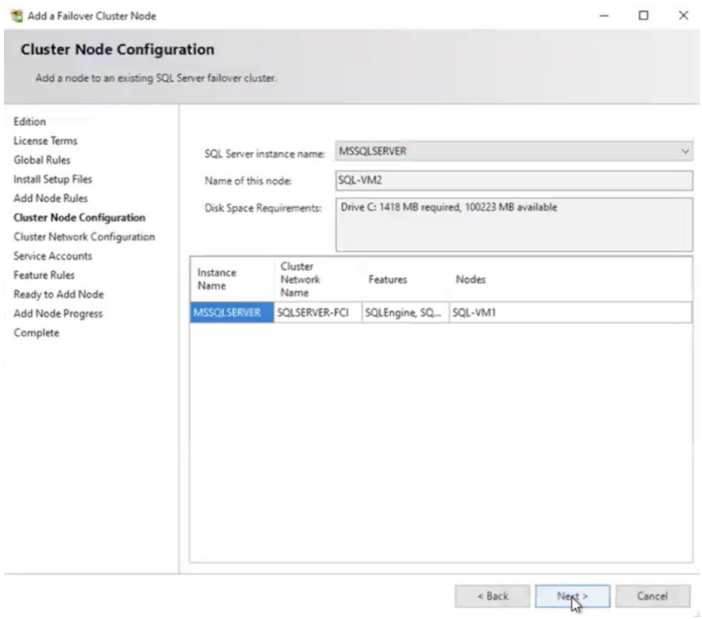


Figure 36 Cluster Node Configuration to add second node to SQL Server FCI

- On the Server Configuration - Service Accounts page, specify the login accounts for SQL Server services. For failover cluster instance installations, account name and startup type information will be pre-populated on this page based on settings provided for the active node. Click Next.
- The System Configuration Checker will run one more set of rules to validate your computer configuration with SQL Server features you have specified.
- The **Ready to Add Node** page displays a tree view of installation options that were specified during Setup.
- Add Node Progress page provides status so you can monitor installation progress as Setup continues.

- After installation, the **Complete** page provides a link to the summary log file for the installation and other important notes. To complete the SQL Server installation process, click **Close**.

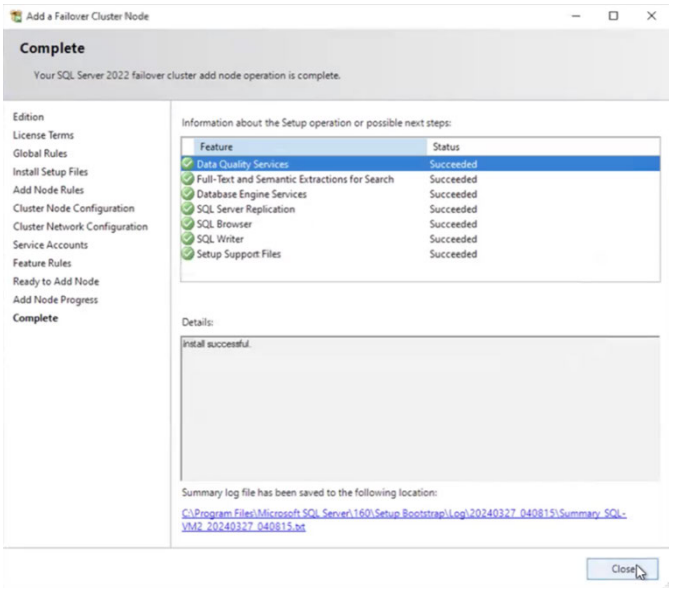


Figure 37 SQL Server failover cluster add node operation completed

This completes the installation of SQL Server Failover Cluster Instance (FCI) with two nodes (SQL-VM1 and SQL-VM2). For more detailed information on this topic, see SQL Server Failover Cluster Installation.

It is recommended to Validate Cluster Configuration again once you have installed the SQL Server Failover Cluster Instance on all the nodes. You can view the SQL Server failover cluster instance by navigating to **Roles** under **Failover Cluster Manager**.

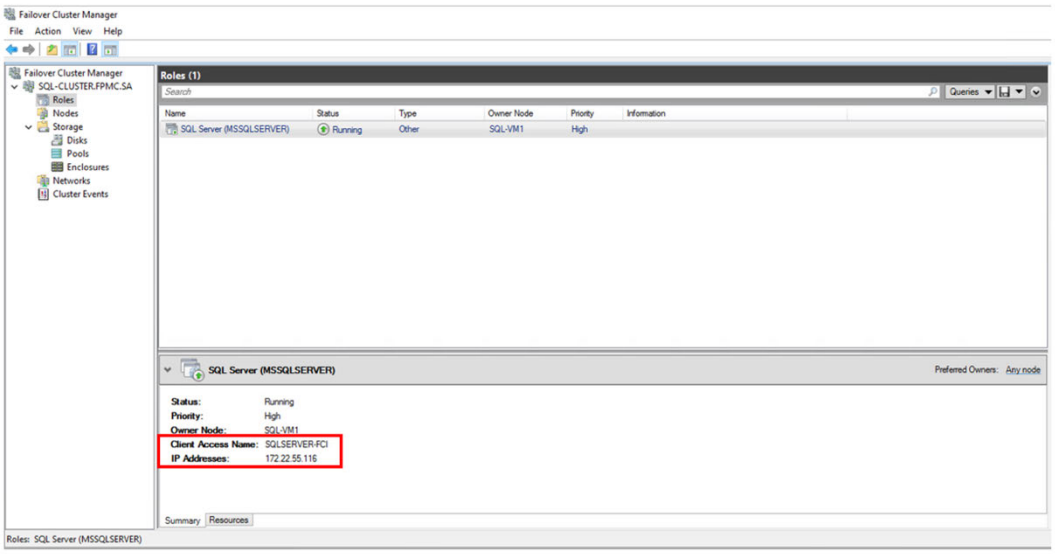


Figure 38 SQL Server FCI in Failover Cluster Manager

Install SQL Server Management Tools

Once the SQL Server FCI is up and running, we need to install the SQL Server Management Tools for managing the SQL infrastructure.

SQL Server Management Studio (SSMS) is one such tool that provides an integrated environment to access, configure, manage, administer, and develop all components of SQL Server. SSMS provides a single comprehensive utility that combines a broad group of graphical tools with many rich script editors to provide to provide access to SQL Server for developers and database administrators.

Perform the following steps on all the WSFC nodes (here SQL-VM1 & SQL-VM2) to install SSMS:

- Insert the SQL Server 2022 installation media, and from root folder, double-click Setup.exe.
- The Installation Wizard will launch the SQL Server Installation Center. To install SQL Server management tools, click **Installation** in the left-hand pane. Then, select **Install SQL Server Management Tools**.
- From this page, click on **Download SQL Server Management Studio (SSMS)**.
- Launch the downloaded SSMS-Setup-ENU.exe file. You should now see the install screen for **Microsoft SQL Server Management Studio**. Click **Install**.
- Once the installation has completed, if it says a restart is required, click **Restart**.
- Launch the SSMS app and connect to SQL Server failover cluster instance by providing server name and login credentials, as shown below.

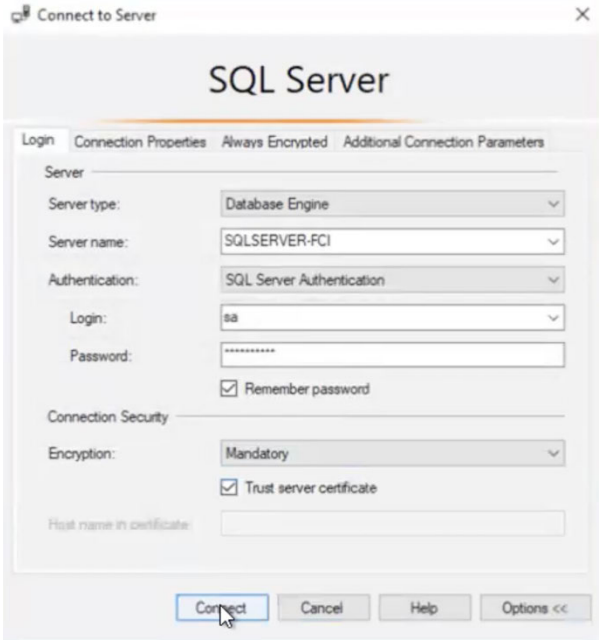


Figure 39 Connecting to SQL Server FCI using SSMS

- Once you have connected to your SQL Server FCI, you can easily view and manage the SQL Server objects like databases, server objects etc. under **Object Explorer** pane.

For detailed information about SSMS installation and configuration, see:

https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16

Creating Sample Database

This section describes how to create a database in SQL Server FCI by using SQL Server Management Studio.

- Launch the SQL Server Management Studio and connect to the SQL Server Failover Cluster Instance.
- In **Object Explorer**, connect to the instance of the SQL Server Database Engine and then expand that instance.
- Right-click **Databases**, and then select **New Database**.
- In New Database, enter a database name (here we have given name “tpcc”).
- To create the database by accepting all default values, select **OK**; otherwise, continue with the following optional steps:
    - To change the owner name, select (…) to select another owner.
    - To change the default values of the primary data and transaction log files, in the **Database files** grid, select the appropriate cell and enter the new value. For more information, see Add Data or Log Files to a Database.
        - For this solution, we have set the initial size of the data file to 190,000 MB and the log file to 65,000 MB.
        - We have set the data file to grow by 64 MB to unlimited size. Set the log file to disable auto-growth. Click OK.
        - Users can set these values depending upon their requirements and type of applications/ workloads they would like to run.
    - To change the collation of the database, select the **Options** page, and then select a collation from the list.
    - To change the recovery model, select the **Options** page and select a recovery model from the list. By default, this is set to Full.
    - As we have installed SQL Server 2022 FCI, so by default the Compatibility level is set to SQL Server 2022 (160).
    - To add a new filegroup, select the **Filegroups** page. Select **Add** and then enter the values for the filegroup.
    - To add an extended property to the database, select the **Extended Properties** page.
        - In the Name column, enter a name for the extended property.
        - In the Value column, enter the extended property text. For example, enter one or more statements that describe the database.
- Click **OK** to create the database, which can take a few minutes to complete.
- Database (tpcc) gets created and shows under Databases tab in Object Explorer as shown in following diagram.

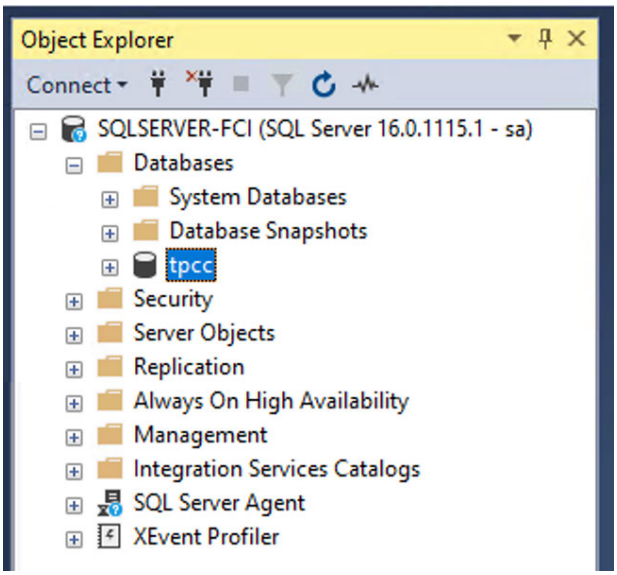


Figure 40 Sample database tpcc created

Windows Client Node Configuration

This section describes the configuration of Windows Client Node from where we will perform the testing and validation of the solution.

- Deploy an instance of Windows VM (here SQL-Client) in VMware vCenter, and make sure to add the IB-MGMT Network under Network Adapters configuration, so that this VM can access the SQL Server Failover Cluster Instance. This VM is called as the Client Node. For step-by-step process for installing the Windows Server 2022 guest operating system in the virtual machine, refer to the VMware documentation.
- Make sure to add this VM to the same domain (here fpmc.sa) as the WSFC nodes.
- Install SQL Server Management Studio (SSMS) on the client node in similar way as it was done on one of the WSFC nodes in previous section.
- To connect to the SQL Server Failover Cluster Instance on Client Node using SSMS:
    - Launch **SSMS** application. Under **Server name**, select the drop-down box and click on **Browse for more**.
    - Select **Network Server** tab. Here, it will automatically fetch your SQL Server FCI under Database Engine. In this case, SQLSERVER-FCI was retrieved automatically.
    - Once the SQL Server FCI is fetched, click on **Connect**. This will connect to the SQL Server failover cluster instance.
    - Under **Object Explorer**, you can see the SQL Server failover cluster instance being connected.
        - Expand **Databases** tab to see your sample database that you created earlier. In our solution, sample database “tpcc” was created which is seen here.

Solution Validation

One of the important use-cases of this solution is ensuring high-availability for critical databases and workloads. Due to the implementation of SQL Server Failover Cluster Instance with WSFC nodes, there is minimum downtime during failures.

When there is hardware or software failure of a server, the applications or clients connecting to the server will experience downtime. Redundant nodes protect the availability of SQL Server instance when it is FCI instead of standalone instance. Only one of the nodes in the FCI owns the WSFC resource group at a time. In case of a failure (hardware failures, operating system failures, application or service failures), or a planned upgrade, the cluster moves the resource group ownership to another WSFC node. This process is transparent to the client or application connecting to SQL Server. This minimizes the downtime the applications or clients experience during a failure.

High Availability for Databases

In this section, we will perform a failover testing to validate the high availability of the solution that consist of SQL Server failover cluster instance with Windows failover clustering nodes. In this solution, we have configured one SQL Server FCI with two WSFC nodes, as explained in previous sections.

Note that Windows Client node is also part of this testing. It will be used to execute various SQL queries using T-SQL (Transact-SQL). It will also be utilized to observe the status of the SQL Server FCI and databases during failover testing.

Prepare For Failover

Before performing failover testing, we need to create some objects for our sample database (that we created earlier).

Complete the following steps before failover testing procedure:

- Check the status of the SQL Server FCI, as to on which node it is running. Login to one of the WSFC nodes (here SQL-VM1), then start **Server Manager**. Then on the **Tools** menu, select **Failover Cluster Manager**. Expand the cluster and then select **Roles**. This will show the status of SQL Server FCI. In this solution, currently SQL Server FCI is running on SQL-VM1 node as shown below.

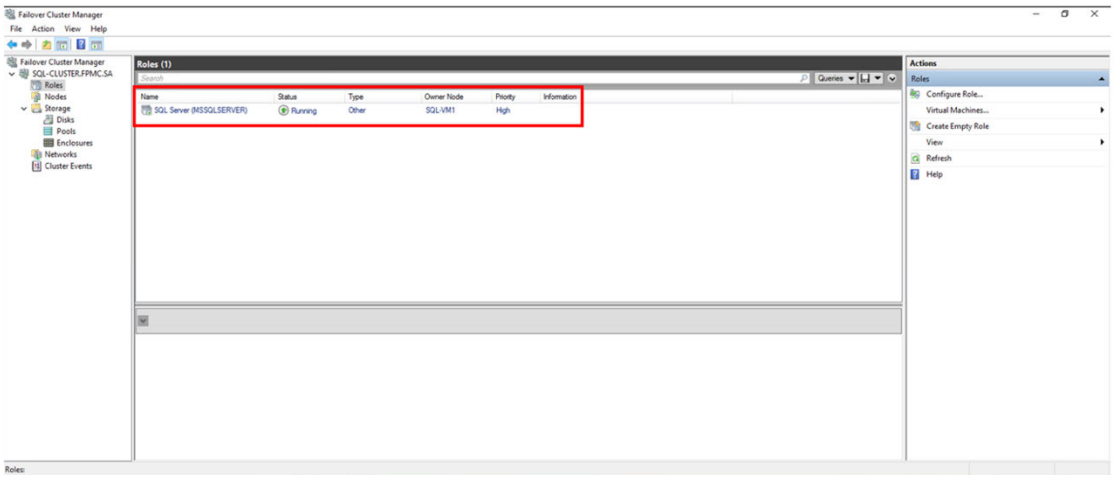


Figure 41 SQL Server FCI running on SQL-VM1 node before failover

- Now, connect to **Windows Client Node** (here SQL-Client) and launch the SQL Server Management Studio (SSMS) application.
- On SSMS, connect to your SQL Server failover cluster instance (here SQLSERVER-FCI).
- Under **Object Explorer**, right-click on the SQL Server instance, and select **New Query**.
- Write a SQL query using T-SQL to create an empty table in the sample database that you created earlier. In this solution, we had created “tpcc” database. To create an empty table called “EMP” in database tpcc, we have written the following query (using CREATE TABLE) as shown in the diagram below.

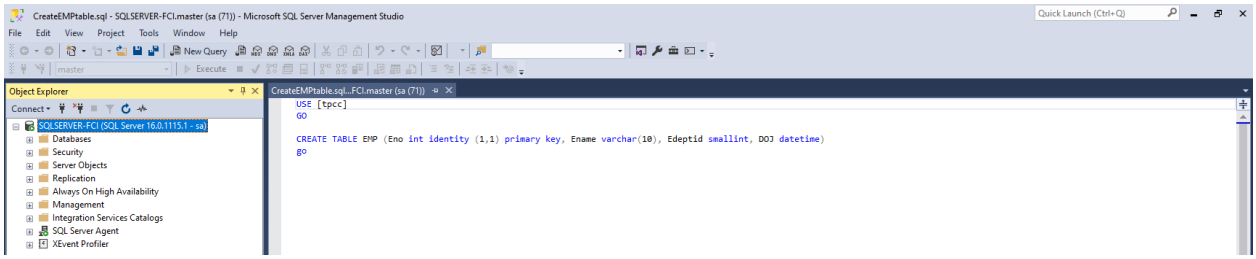


Figure 42 SQL Query to create an empty table EMP in tpcc database

Note: Make sure to use your sample database name instead of “tpcc” in the above query.

- Select the query and click on **Execute** to execute the query.
- This will create an empty table called EMP in the sample database. To view this empty table, expand the **Databases** tab under **Object Explorer**, and select your sample database (here tpcc). Expand the sample database and go to **Tables** and expand it. Here, you will see your newly created table with the name “dbo.EMP”. To view this empty table, right-click the table dbo.EMP, and then select “Select Top 1000 Rows” option. This will execute the SELECT query and you will see the empty table with column names like shown below.

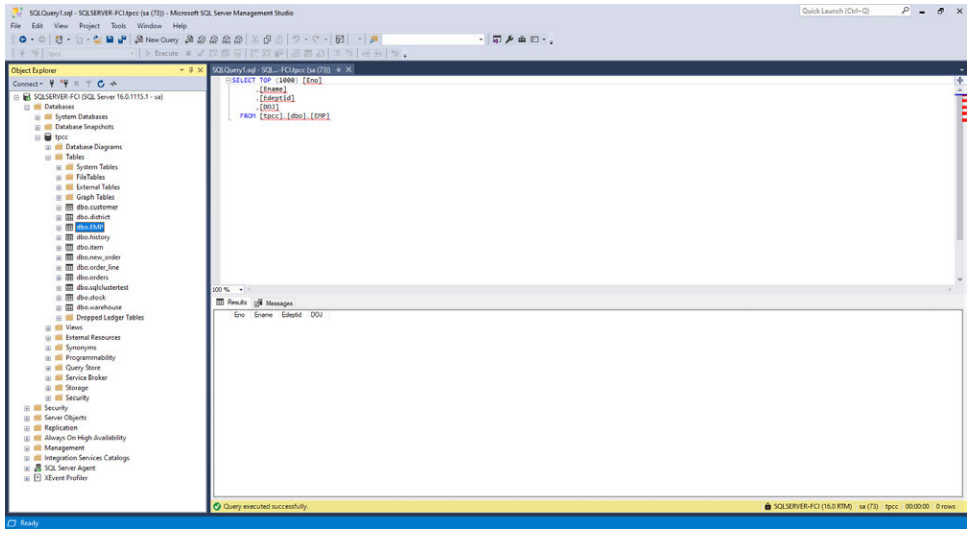


Figure 43 Empty table EMP created in tpcc database

- Next step is to execute a SQL query to insert few rows in this empty table EMP. In this solution, we have inserted four rows in the table EMP by executing the query as shown in the following diagram. Here, this query inserts a row and waits for two seconds and then increments the iterator and inserts the next row.

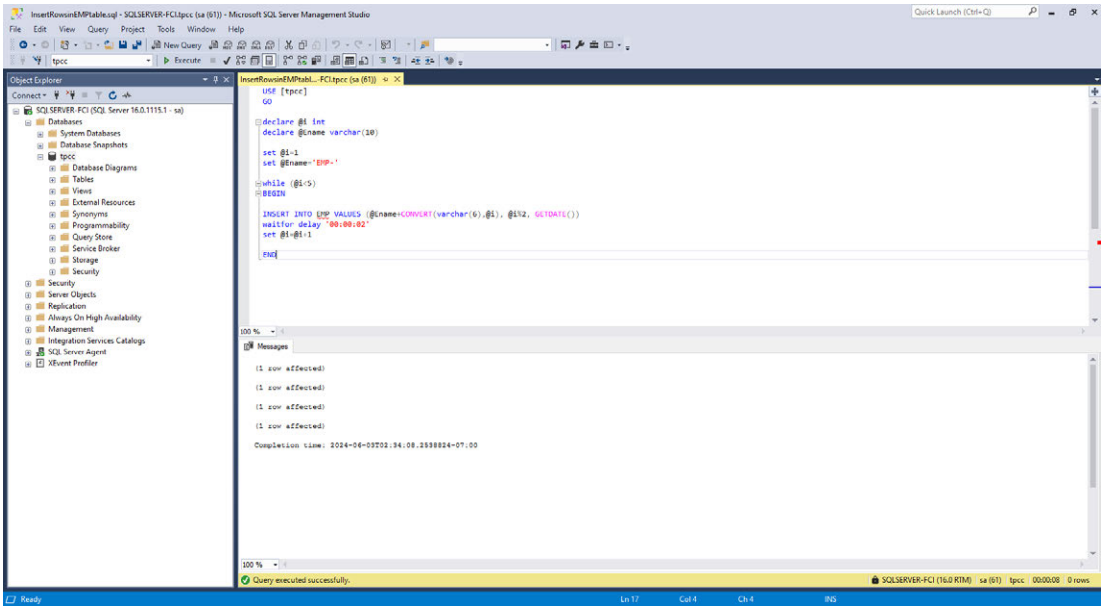
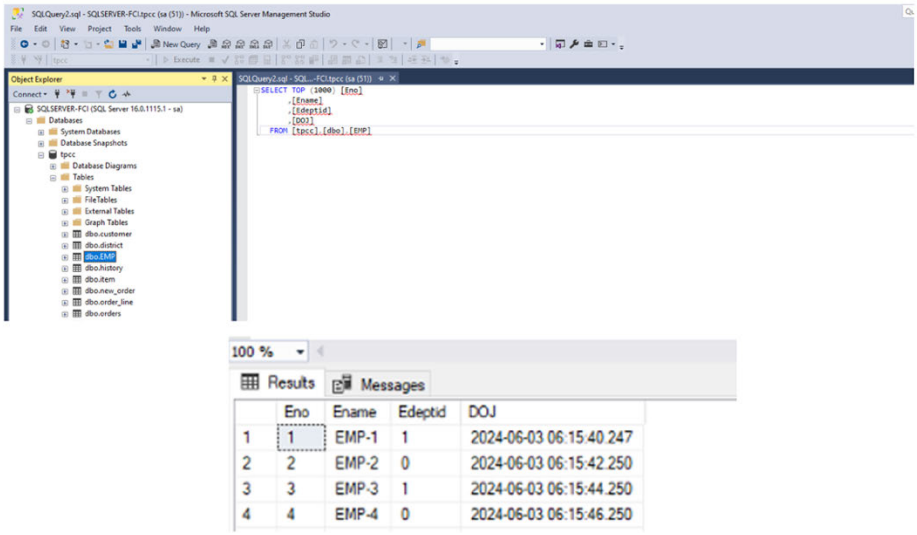


Figure 44 SQL Query executed to insert four rows in EMP table

- To view the rows inserted into the table, expand the Tables folder under the sample database (tpcc), and select the table “dbo.EMP”. Right-click this table and select “Select Top 1000 Rows” option. This will show the rows inserted into the table as shown below. We can see 4 rows inserted in EMP table under tpcc database.



	Eno	Ename	Edeptid	DOJ
1	1	EMP-1	1	2024-06-03 06:15:40.247
2	2	EMP-2	0	2024-06-03 06:15:42.250
3	3	EMP-3	1	2024-06-03 06:15:44.250
4	4	EMP-4	0	2024-06-03 06:15:46.250

Figure 45 Four rows inserted in EMP table

Failover Test

Complete the following steps to perform the failover testing procedure:

- Login to **Windows Client Node** (SQL-Client) and launch the SQL Server Management Studio (SSMS) application.
- On SSMS, connect to the SQL Server failover cluster instance (SQLSERVER-FCI).
- Under **Object Explorer**, right-click on the SQL Server instance, and select **New Query**.
- Now, we execute a SQL query (using T-SQL) to insert some large number of rows in the EMP table, so that we can perform the failover process while query execution is in progress. In this solution, we have written the query to insert 99 rows in the table EMP.
    - Note that the only change you need to do in the previous query is to update the conditional statement under while loop. In this case, we set the condition "@i<100" to insert 99 rows in the table. You can use similar kind of conditional statements to perform validation.
    - Click on **Execute** to execute this query. It will take enough time to execute, and shows as "Executing query", as illustrated in the following diagram. This gives us sufficient time to perform failover testing.
    - Use your sample database name instead of "tpcc" in the below query.

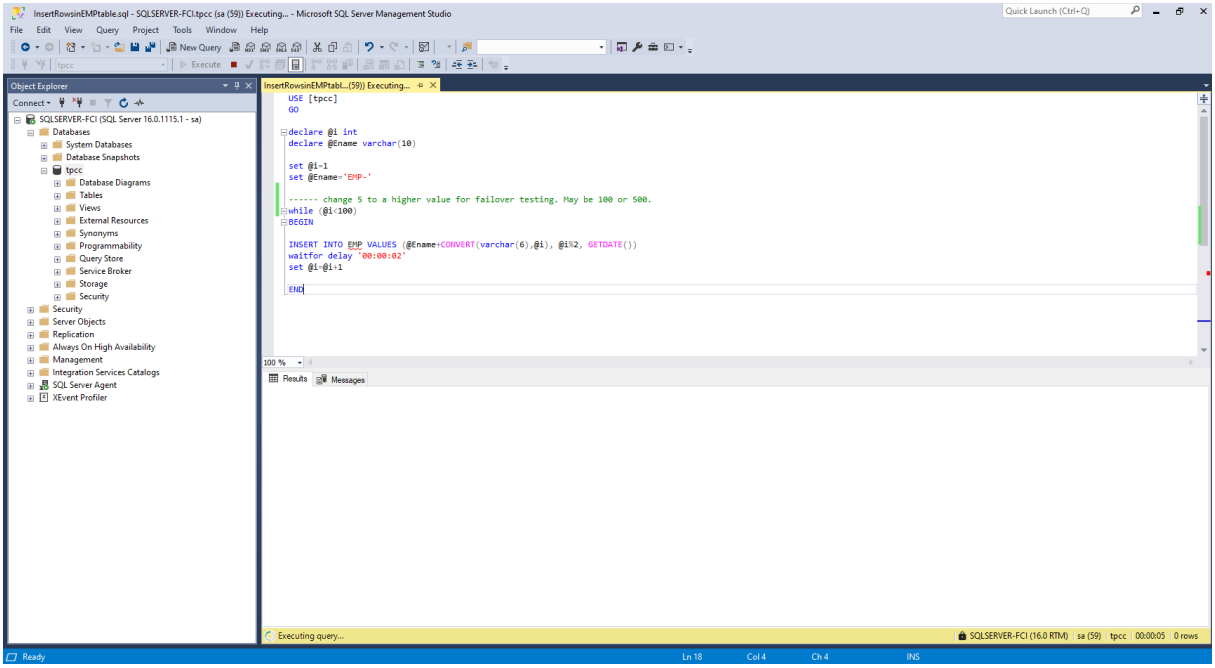


Figure 46 SQL Query execution in progress to insert 99 rows in EMP table

- To simulate failure of the WSFC node for failover testing, login to one of the WSFC nodes and launch **Failover Cluster Manager**. Expand the cluster (SQL-CLUSTER.FPMC.SA) and then select **Nodes**. Now, pause the node on which SQL Server FCI is running. In this solution, SQLSERVER-FCI is running on SQL-VM1 node. To pause SQL-VM1 node, right-click the node and select **Pause** and then select **Drain Roles**.

- SQL-VM1 node is now paused as shown in the diagram below.

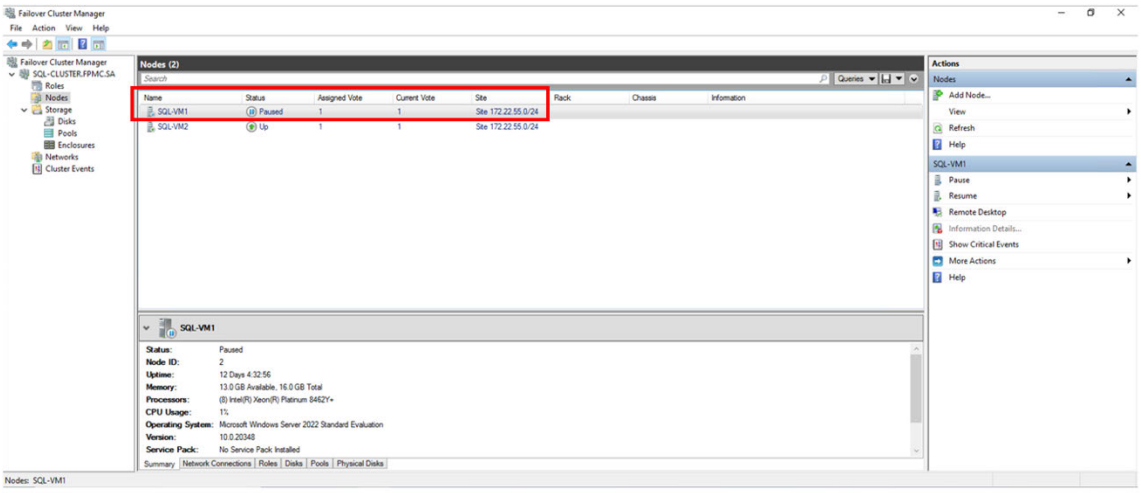


Figure 47 SQL-VM1 node paused

- Now go to the Windows Client Node and see the status of the query execution and SQL Server failover cluster instance.
  - We observe that the SQL Server FCI has been disconnected and the query execution has now stopped, as shown in the following diagram.
  - This is an expected behaviour as SQL Server FCI takes few seconds (or minutes) to failover to the other active node in the failover cluster. But this downtime is less due to failover clustering implementation. So, SQLSERVER-FCI will take couple of seconds to move to SQL-VM2 node in the cluster.

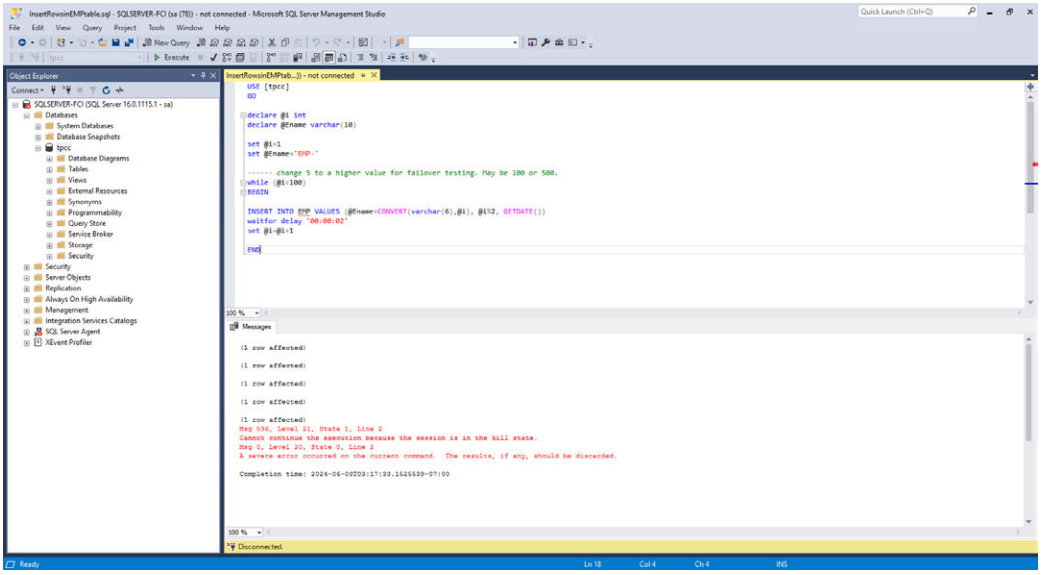


Figure 48 SQL Server FCI disconnected and query execution stopped

- Now go to WSFC node where **Failover Cluster Manager** is launched. Expand the cluster (SQL-CLUSTER.FPMC.SA) and then select **Roles**. You will observe the status of your SQL Server FCI as "**Pending**". This confirms the behaviour of SQL Server failover cluster instance that it takes some time to failover to the other active node in the cluster during failover process. This is illustrated in the below diagram.

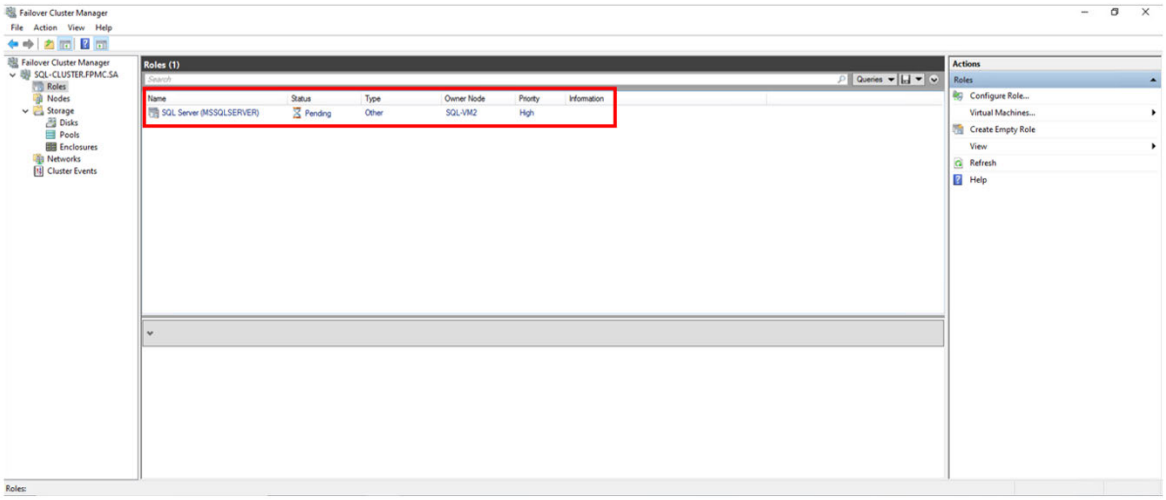


Figure 49 SQL Server FCI status showing as Pending during failover

- During the failover process, the Windows failover cluster (SQL-CLUSTER.FPMC.SA) takes some time to move the resource group ownership to the other active WSFC node (SQL-VM2). Once this ownership is completely moved to the active WSFC node, the SQL Server FCI starts running on this active node. This completes the failover process.
    - As shown in the following diagram, the SQLSERVER FCI status is now showing as "**Running**" and the owner node is now **SQL-VM2**.

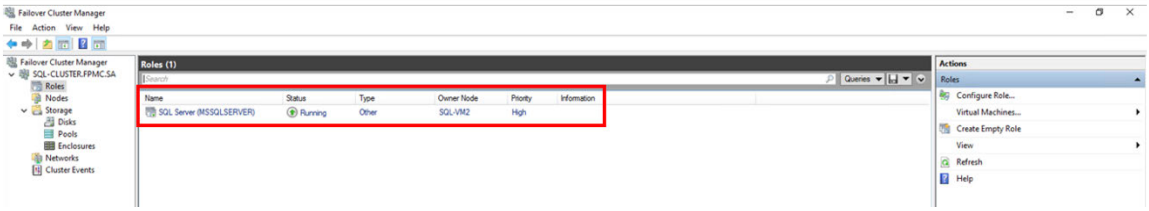


Figure 50 Failover completed successfully with SQL Server FCI Running Status

- Now the SQL Server FCI is fully up and running on active WSFC node (SQL-VM2). Go to Windows Client Node (SQL-Client) and refresh the SQL Server FCI connection in SSMS. To do so, right-click the SQL Server FCI, and select **Refresh**. Now SQL Server FCI shows as "Connected" in SSMS.
- Next step is to find out how many rows got inserted into the EMP table under tpcc database before the failure of SQL Server FCI. For this, execute a **SELECT** query on the EMP table with NOLOCK parameter, as shown in the diagram below. This query allows SQL Server to read data from EMP table by ignoring any locks therefore not get blocked by other processes.

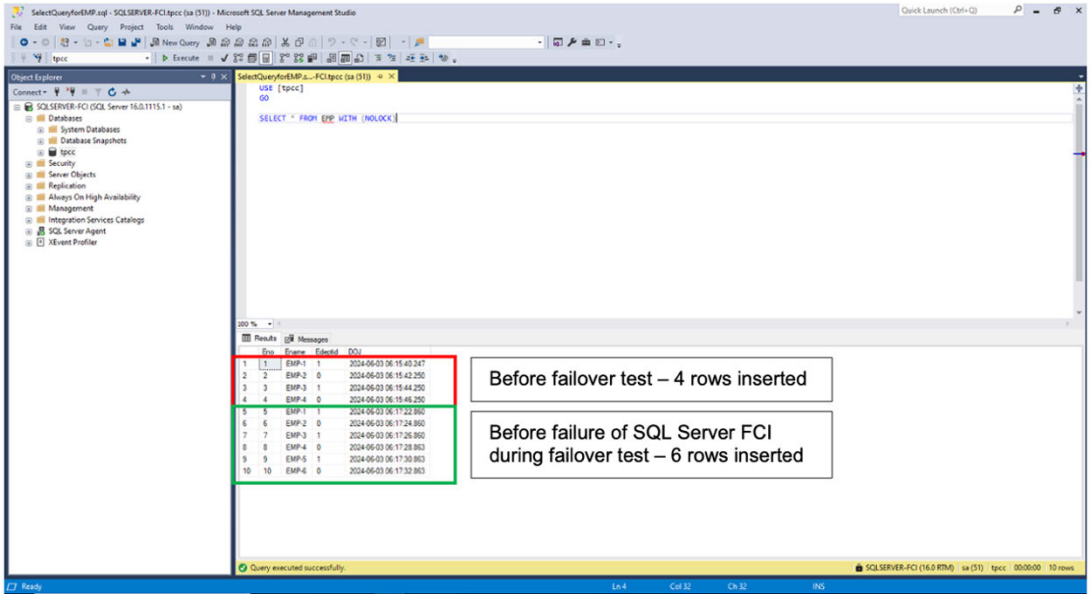


Figure 51 SQL Query to find number of rows inserted into EMP table before SQL Server FCI failure

- In the above diagram, you can see two rectangle boxes highlighted with red and green colors.
  - Number of rows in the red rectangle box were inserted before the failover process. This can be confirmed with Figure 45. Total four rows were inserted before failover.
  - Number of rows in the green rectangle box were inserted before the failure of SQLSERVER-FCI during failover process. Total six rows got inserted into the EMP table under tpcc database.
  - Note that whichever rows are inserted and committed into the table before the failure of the SQL Server FCI happens, those rows will be persisted to the disks and you will find them when the SQL server instance (SQLSERVER-FCI) comes online on the other active WSFC node (SQL-VM2).
  - In-flight transactions (which are not committed) will be undone and not shown in the tpcc database, which is an expected behaviour.
  - The Log records for transactions (which are committed but not yet applied on the data pages), will be replayed and the corresponding updates or inserts will be shown in the database.
  - This step also proves that we can easily read the data from the database (tpcc) once failover has successfully completed. It is only possible because of the failover functionality present on the nodes and SQL Server instance.
- Let's execute one SQL query to make sure we are able to write to the database after SQL Server FCI moved to the other WSFC active node (SQL-VM2).
  - Go to Windows Client node (SQL-Client) and launch the SSMS application. Connect to SQL Server FCI. Right-click the SQL Server FCI and select New Query.
  - Execute the same **INSERT** query as before to insert 99 rows into the EMP table under tpcc database.

- This query will take some time to execute as there are enough transactions to be committed. In this case, it took around 3 mins 19 seconds to execute this query as shown below.

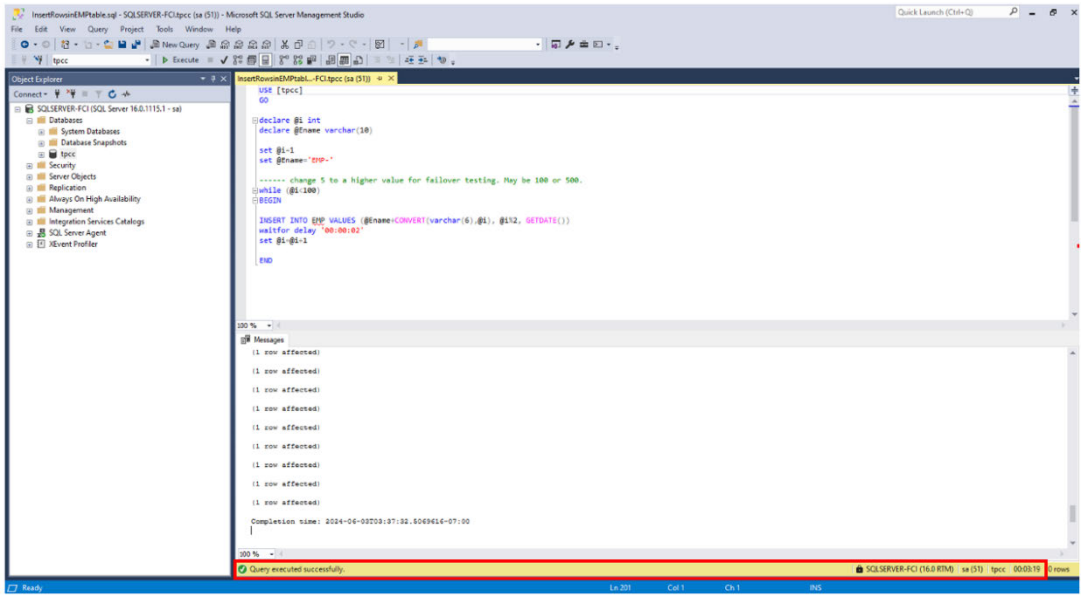


Figure 52 SQL Query to insert 99 rows into the EMP table when SQL Server FCI moved to SQL-VM2 node

- Next step is to confirm whether these 99 rows got inserted into the EMP table under tpcc database. For this, execute a **SELECT** query on the EMP table as shown in the following diagram.

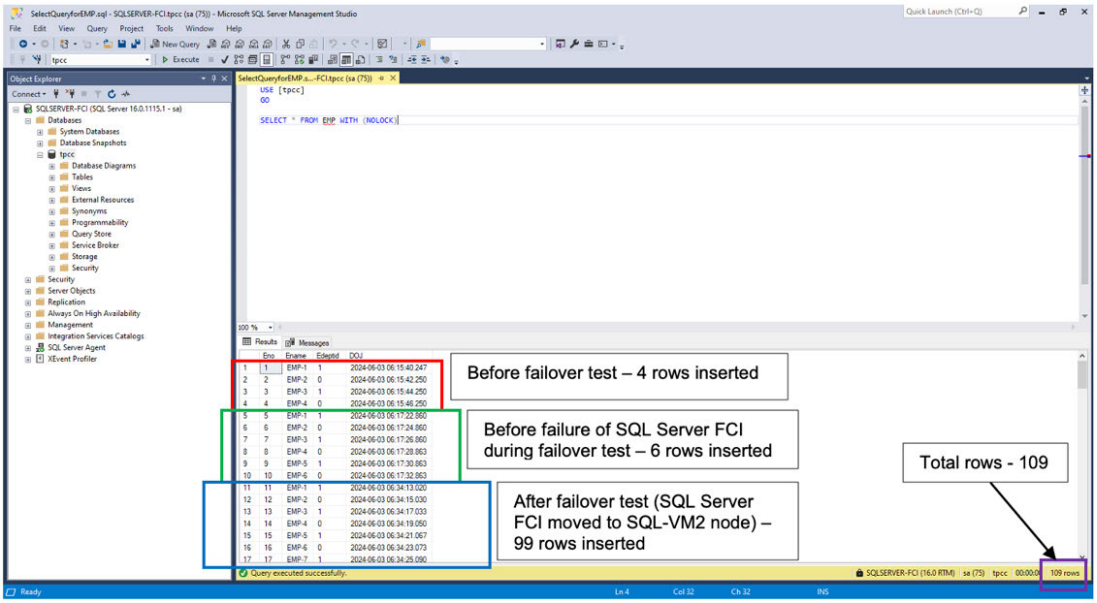


Figure 53 Writing to tpcc database successful when SQL Server FCI moved to SQL-VM2 node

- We can easily observe from the above diagram that 99 rows got inserted into the EMP table under tpcc database.
- This validates that one can easily write to the database after the SQL Server failover cluster instance has successfully moved to the other active WSFC node in the cluster.
- This test verifies the high availability of the solution in case of a failure. Users can still access the database and perform read and write operations if required.

Conclusion

Microsoft SQL Server is the most popular RDMS (Relational Database Management System) installed by the users worldwide. With the right sizing and validated designs, FlexPod is an ideal fit for the majority of the SQL Server workloads like databases. The combination of ONTAP storage and Microsoft SQL Server enables the creation of enterprise-level storage designs that can meet today's most demanding application requirements.

FlexPod with Microsoft SQL Server Clustering is a high availability and disaster recovery solution that takes advantage of Windows Server Failover Clustering (WSFC). SQL Server failover cluster instance has various features which increase application availability, provide better runs on hardware investments, and simplify high availability deployment and management.

This document illustrates the implementation of FlexPod infrastructure with SQL Server Failover Cluster Instance (FCI) consisting of Windows failover clustering nodes. High availability scenarios have been tested here and the results showcase that this solution provides continuous database availability in the event of node failures or outages. Customers can utilize this framework to deploy FlexPod with SQL Server failover clustering and build a highly available, resilient, and efficient system which can handle most of their demanding database or application requirements.

Acknowledgement

Bobby Oommen - Sr. Manager, FlexPod Solutions, NetApp

Gopu Narasimha Reddy - Technical Marketing Engr, Cisco

Where to find additional information

To learn more about the information that is described in this document, review the following documents and/or websites:

- FlexPod Home Page: https://www.flexpod.com
- NetApp Interoperability Matrix Tool: http://support.netapp.com/matrix/
- Cisco UCS Hardware and Software Interoperability Tool: http://www.cisco.com/web/techdoc/ucs/interoperability/matrix/matrix.html
- VMware Compatibility Guide: http://www.vmware.com/resources/compatibility/search.php
- NetApp Product Documentation: https://www.netapp.com/support-and-training/documentation/
- Cisco Validated Design and deployment guides for FlexPod: https://www.cisco.com/c/en/us/solutions/design-zone/data-center-design-guides/flexpod-design-guides.html
- FlexPod Datacenter for Microsoft SQL Server 2022 and VMware vSphere 8.0: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_sql2022_xseries.html
- Windows Server Failover Clustering: https://learn.microsoft.com/en-us/windows-server/failover-clustering/create-failover-cluster
- SQL Server Failover Cluster Instance: https://learn.microsoft.com/en-us/sql/sql-server/failover-clusters/windows/always-on-failover-cluster-instances-sql-server?view=sql-server-ver16

Version history

As an option, use the NetApp Table style to create a Version History table. Do not add a table number or caption.

Version	Date	Document version history
Version 1.0	July 2024	Initial release.

Refer to the Interoperability Matrix Tool (IMT) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.

Copyright information

Trademark information

NetApp