



# **Microsoft SQL Server**를 참조하십시오

## Enterprise applications

NetApp  
February 10, 2026

# 목차

Microsoft SQL Server를 참조하십시오.....	1
개요.....	1
바로 Microsoft SQL Server 워크로드입니다.....	1
데이터베이스 구성.....	1
CPU 구성.....	2
메모리 구성.....	5
공유 인스턴스와 전용 인스턴스 비교.....	8
tempdb 파일.....	9
AFF/FAS 시스템에 대한 스토리지 구성.....	10
개요.....	10
데이터베이스 파일 및 파일 그룹.....	12
스토리지 효율성.....	17
데이터 보호.....	21
재해 복구.....	23
ASA R2 시스템의 스토리지 구성.....	42
개요.....	42
데이터베이스 파일 및 파일 그룹.....	44
데이터 보호.....	48
재해 복구.....	49
데이터 보안.....	63
Snapshot 복사본.....	63
변조 방지 스냅샷.....	64
SnapMirror 복제.....	64
스토리지 가상 머신.....	64
관리 RBAC.....	65
다단계 인증(MFA).....	65
API RBAC입니다.....	65
MAV(Multi-admin verification).....	65

# Microsoft SQL Server를 참조하십시오

## 개요

ONTAP은 Microsoft SQL Server 데이터베이스를 위한 엔터프라이즈급 보안 및 성능 솔루션을 제공하는 동시에 환경을 관리할 수 있는 세계적 수준의 도구를 제공합니다.



이 문서는 이전에 게시된 기술 보고서 \_TR-4590: Microsoft SQL Server에 대한 모범 사례 가이드를 ONTAP\_로 대체합니다

NetApp는 독자가 다음에 대한 실무 지식을 보유하고 있다고 가정합니다.

- ONTAP
- Microsoft SQL Server 아키텍처 및 관리
- 백업 소프트웨어로 제공되는 NetApp SnapCenter에는 다음이 포함됩니다.
  - Microsoft Windows용 SnapCenter 플러그인
  - SQL Server용 SnapCenter 플러그인

이 모범 사례 섹션의 범위는 NetApp이 스토리지 인프라에 권장하는 원칙과 기본 표준에 따른 기술 설계로 제한됩니다. 엔드 투 엔드 구현이 범위를 벗어납니다.

NetApp 제품 간의 호환성에 대한 자세한 내용은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#) 참조하십시오.

## 바로 Microsoft SQL Server 워크로드입니다

SQL Server를 배포하기 전에 SQL Server 데이터베이스 인스턴스가 지원하는 애플리케이션의 워크로드 요구 사항을 이해해야 합니다. 애플리케이션마다 용량, 성능 및 가용성에 대한 요구사항이 다르므로 각 데이터베이스가 이러한 요구사항을 최적으로 지원하도록 설계되어야 합니다. 많은 조직에서 애플리케이션 요구 사항을 사용하여 SLA를 정의하여 데이터베이스를 여러 관리 계층으로 분류합니다. SQL Server 워크로드는 다음과 같이 분류되는 경우가 많습니다.

- OLTP는 조직에서 가장 중요한 데이터베이스입니다. 이러한 데이터베이스는 일반적으로 고객용 애플리케이션을 뒷받침하며 회사의 핵심 운영에 중요한 요소로 간주됩니다. 미션 크리티컬 OLTP 데이터베이스 및 해당 데이터베이스가 지원하는 애플리케이션에는 고성능이 요구되는 SLA가 있으며, 성능 저하에 민감하고 최대 가용성이 요구되는 경우가 많습니다. 또한 Always On Failover Clusters 또는 Always On Availability Group의 후보일 수도 있습니다. 이러한 유형의 데이터베이스에서 I/O 혼합은 일반적으로 75% ~ 90% 랜덤 읽기와 25% ~ 10% 쓰기로 특징.
- DSS(의사 결정 지원 시스템) 데이터베이스(때로는 데이터 웨어하우스라고도 함) 이러한 데이터베이스는 비즈니스 분석을 이용하는 대부분의 조직에 있어 미션 크리티컬한 기능입니다. 이러한 데이터베이스는 쿼리를 실행할 때 CPU 사용률 및 디스크의 읽기 작업에 민감합니다. 많은 조직에서 DSS 데이터베이스는 월, 분기 및 연말에 가장 중요합니다. 이 워크로드는 일반적으로 거의 100% 읽기 I/O 혼합이 있으며 IO 처리량이 IOPS보다 더 중요한 경우가 많습니다.

## 데이터베이스 구성

## CPU 구성

SQL Server 성능은 CPU 및 코어 구성에 대해 여러 가지 종속성이 있습니다.

### 하이퍼 스레딩

하이퍼스레딩은 X86 프로세서에서 수행되는 계산의 병렬화를 개선하는 SMT(동시 다중 스레딩) 구현을 의미합니다. SMT는 Intel 및 AMD 프로세서 모두에서 사용할 수 있습니다.

하이퍼 스레딩은 운영 체제에 물리적 CPU로 표시되는 논리 CPU를 생성합니다. 그런 다음 SQL Server는 이러한 추가 CPU를 보고 물리적으로 존재하는 것보다 더 많은 코어가 있는 것처럼 사용합니다. 이를 통해 병렬화를 증가시켜 성능을 크게 향상시킬 수 있습니다.

여기서 주의해야 할 점은 각 SQL Server 버전에는 사용할 수 있는 컴퓨팅 성능에 대한 제한이 있다는 것입니다. 자세한 내용은 ["SQL Server 버전별 컴퓨팅 용량 제한"](#) 참조하십시오.

### 코어 및 라이센싱

SQL Server 라이선스에는 두 가지 옵션이 있습니다. 첫 번째는 서버 + 클라이언트 액세스 라이센스(CAL) 모델이며 두 번째는 프로세서당 코어 모델입니다. 서버 + CAL 전략을 통해 SQL Server에서 사용할 수 있는 모든 제품 기능에 액세스할 수 있지만 소켓당 20개의 CPU 코어로 하드웨어 제한이 있습니다. 소켓당 20개가 넘는 CPU 코어가 있는 서버에 대해 SQL Server Enterprise Edition + CAL을 사용하는 경우에도 응용 프로그램은 해당 인스턴스에서 이러한 모든 코어를 한 번에 사용할 수 없습니다.

아래 그림에서는 시작 후 코어 제한 적용을 나타내는 SQL Server 로그 메시지를 보여 줍니다.

```

2017-01-11 07:16:30.71 Server      Microsoft SQL Server 2016
(RTM) - 13.0.1601.5 (X64)
Apr 29 2016 23:23:58
Copyright (c) Microsoft Corporation
Enterprise Edition (64-bit) on Windows Server 2016
Datacenter 6.3 <X64> (Build 14393: )

2017-01-11 07:16:30.71 Server      UTC adjustment: -8:00
2017-01-11 07:16:30.71 Server      (c) Microsoft Corporation.
2017-01-11 07:16:30.71 Server      All rights reserved.
2017-01-11 07:16:30.71 Server      Server process ID is 10176.
2017-01-11 07:16:30.71 Server      System Manufacturer:
'FUJITSU', System Model: 'PRIMERGY RX2540 M1'.
2017-01-11 07:16:30.71 Server      Authentication mode is MIXED.
2017-01-11 07:16:30.71 Server      Logging SQL Server messages
in file 'C:\Program Files\Microsoft SQL Server
\MSSQL13.MSSQLSERVER\MSSQL\Log\ERRORLOG'.
2017-01-11 07:16:30.71 Server      The service account is 'SEA-
TM\FUJIA2R30$'. This is an informational message; no user action
is required.
2017-01-11 07:16:30.71 Server      Registry startup parameters:
  -d C:\Program Files\Microsoft SQL Server
  \MSSQL13.MSSQLSERVER\MSSQL\DATA\master.mdf
  -e C:\Program Files\Microsoft SQL Server
  \MSSQL13.MSSQLSERVER\MSSQL\Log\ERRORLOG
  -l C:\Program Files\Microsoft SQL Server
  \MSSQL13.MSSQLSERVER\MSSQL\DATA\mastlog.ldf
  -T 3502
  -T 834
2017-01-11 07:16:30.71 Server      Command Line Startup
Parameters:
  -s "MSSQLSERVER"
2017-01-11 07:16:30.72 Server      SQL Server detected 2 sockets
with 18 cores per socket and 36 logical processors per socket,
72 total logical processors; using 40 logical processors based
on SQL Server licensing. This is an informational message; no
user action is required.
2017-01-11 07:16:30.72 Server      SQL Server is starting at

```

따라서 모든 CPU를 사용하려면 프로세서당 코어 라이센스를 사용해야 합니다. SQL Server 라이센스에 대한 자세한 내용은을 참조하십시오 ["SQL Server 2022: 최신 데이터 플랫폼"](#).

## CPU 선호도

성능 문제가 발생하지 않는 한 프로세서 선호도 기본값을 변경할 필요는 없습니다. 그러나 성능 문제가 무엇이고 어떻게 작동하는지 이해하는 것이 좋습니다.

SQL Server는 다음 두 가지 옵션을 통해 프로세서 선호도를 지원합니다.

- CPU 선호도 마스크
- 선호도 I/O 마스크

SQL Server는 운영 체제에서 사용할 수 있는 모든 CPU를 사용합니다(프로세서당 코어 라이센스를 선택한 경우). 또한 각 CPU에 스케줄러를 생성하여 모든 워크로드를 위한 리소스를 최대한 활용할 수 있습니다. 멀티태스킹 시 서버의 운영 체제 또는 다른 응용 프로그램은 프로세스 스레드를 하나의 프로세서에서 다른 프로세서로 전환할 수 있습니다. SQL Server는 리소스를 많이 사용하는 응용 프로그램이므로 이 경우 성능에 영향을 줄 수 있습니다. 영향을 최소화하기 위해 모든 SQL Server 로드가 미리 선택된 프로세서 그룹으로 전달되도록 프로세서를 구성할 수 있습니다. 이는 CPU 선호도 마스크를 사용하여 수행할 수 있습니다.

선호도 입출력 마스크 옵션은 SQL Server 디스크 입출력을 CPU의 하위 집합에 바인딩합니다. SQL Server OLTP 환경에서 이 확장은 I/O 작업을 실행하는 SQL Server 스레드의 성능을 크게 향상시킬 수 있습니다.

## 최대 평행 거리(MAXDOP)

기본적으로 SQL Server는 프로세서별 코어 라이센스를 선택한 경우 쿼리 실행 중에 사용 가능한 모든 CPU를 사용합니다.

이 방법은 대규모 쿼리에 유용하지만 성능 문제가 발생하고 동시성이 제한될 수 있습니다. 더 좋은 방법은 병렬 처리를 단일 CPU 소켓의 물리적 코어 수로 제한하는 것입니다. 예를 들어, 하이퍼스레딩에 관계없이 소켓당 12개의 코어가 있는 물리적 CPU 소켓 2개가 있는 서버에서 MAXDOP 12로 설정해야 합니다. MAXDOP 사용할 CPU를 제한하거나 지정할 수 없습니다. 대신 단일 배치 쿼리에서 사용할 수 있는 CPU 수를 제한합니다.



\* NetApp는 데이터 웨어하우스와 같은 DSS에 대해 50부터 시작하여 필요한 경우 조정 기능을 위 또는 아래로 탐색할 것을 MAXDOP 권장합니다. 변경할 때 응용 프로그램에서 중요한 쿼리를 측정해야 합니다.

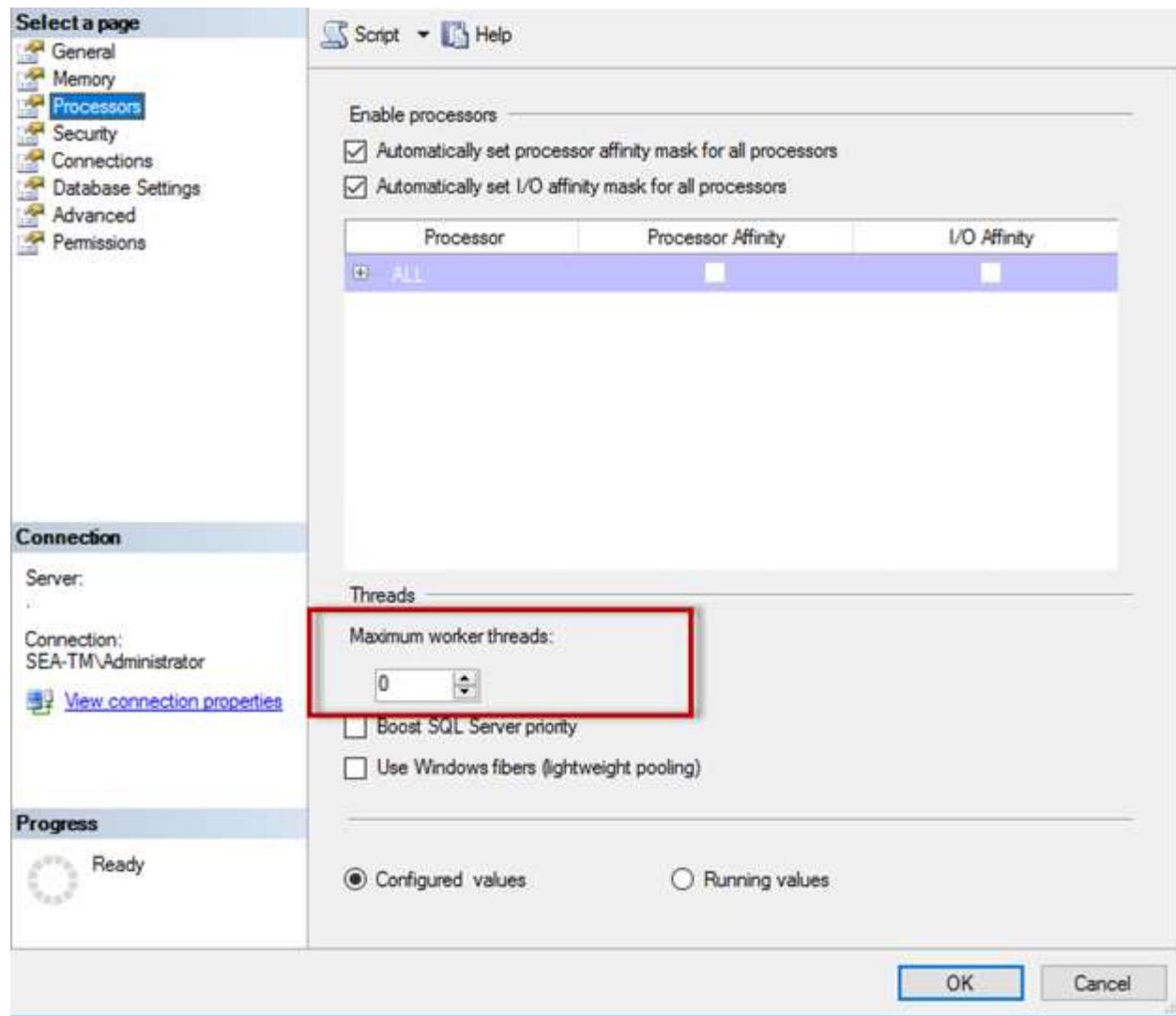
## 최대 작업자 스레드 수

최대 작업자 스레드 옵션을 사용하면 많은 수의 클라이언트가 SQL Server에 연결될 때 성능을 최적화할 수 있습니다.

일반적으로 각 쿼리에 대해 별도의 운영 체제 스레드가 만들어집니다. SQL Server에 수백 개의 동시 연결이 이루어지면 쿼리당 하나의 스레드 구성이 과도한 시스템 리소스를 소모할 수 있습니다. 이 `max worker threads` 옵션을 사용하면 SQL Server에서 더 많은 쿼리 요청을 종합적으로 처리할 수 있는 작업자 스레드 풀을 만들어 성능을 향상시킬 수 있습니다.

기본값은 0입니다. 이 값을 사용하면 SQL Server가 시작 시 작업자 스레드 수를 자동으로 구성할 수 있습니다. 이는 대부분의 시스템에서 작동합니다. Max worker 스레드는 고급 옵션이므로 숙련된 데이터베이스 관리자(DBA)의 도움 없이 변경할 수 없습니다.

작업자 스레드를 더 많이 사용하도록 SQL Server를 구성해야 하는 경우는 언제입니까? 각 스케줄러의 평균 작업 대기열 길이가 1을 초과하면 시스템에 더 많은 스레드를 추가할 수 있습니다. 단, 로드가 CPU에 바인딩되지 않거나 다른 과중한 대기가 발생하는 경우에만 가능합니다. 이러한 상황이 발생하는 경우 스레드를 더 추가해도 다른 시스템 병목 현상이 발생하기 때문에 도움이 되지 않습니다. 최대 작업자 스레드에 대한 자세한 내용은 ["최대 작업자 스레드 서버 구성 옵션을 구성합니다"](#) 참조하십시오.



SQL Server Management Studio를 사용하여 최대 작업자 스레드 구성

다음 예제에서는 T-SQL을 사용하여 최대 작업 스레드 옵션을 구성하는 방법을 보여 줍니다.

```
EXEC sp_configure 'show advanced options', 1;
GO
RECONFIGURE ;
GO
EXEC sp_configure 'max worker threads', 900 ;
GO
RECONFIGURE;
GO
```

## 메모리 구성

다음 섹션에서는 데이터베이스 성능을 최적화하는 데 필요한 SQL Server 메모리 설정에 대해 설명합니다.

## 최대 서버 메모리

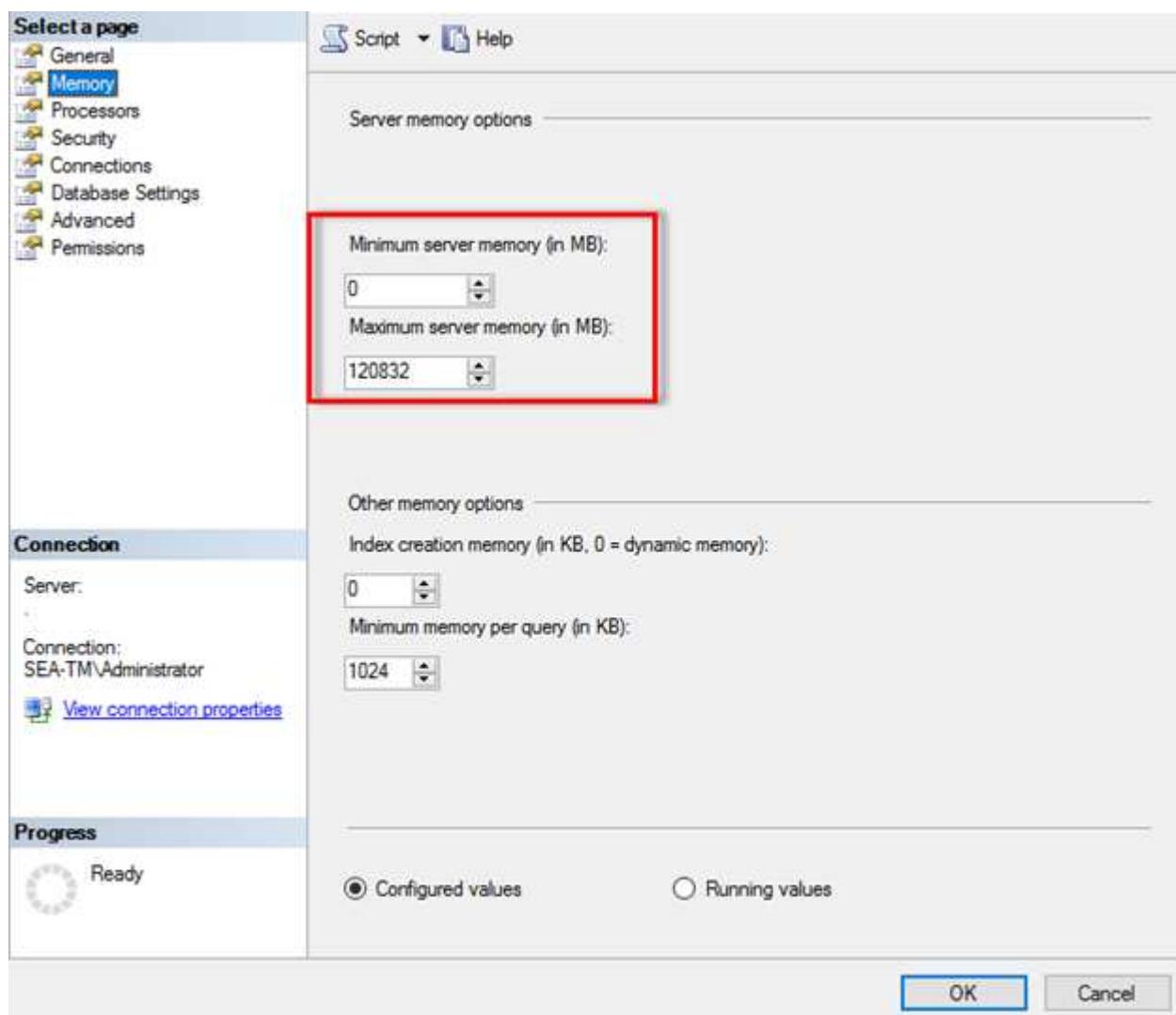
최대 서버 메모리 옵션은 SQL Server 인스턴스가 사용할 수 있는 최대 메모리 양을 설정합니다. 일반적으로 SQL Server가 실행 중인 동일한 서버에서 여러 응용 프로그램이 실행되고 있고 이러한 응용 프로그램이 제대로 작동할 수 있는 충분한 메모리를 확보하려는 경우에 사용됩니다.

일부 응용 프로그램은 시작할 때 사용 가능한 메모리만 사용하고 메모리 압력을 받는 경우에도 추가 메모리를 요청하지 않습니다. 최대 서버 메모리 설정이 재생되는 위치입니다.

SQL Server 인스턴스가 여러 개 있는 SQL Server 클러스터에서 각 인스턴스가 리소스를 놓고 경쟁할 수 있습니다. 각 SQL Server 인스턴스에 대한 메모리 제한을 설정하면 각 인스턴스에 대해 최상의 성능을 보장할 수 있습니다.



\*NetApp는 성능 문제를 방지하기 위해 운영 체제에 최소 4GB~6GB의 RAM을 남겨 둘 것을 권장합니다.



**SQL Server Management Studio**를 사용하여 최소 및 최대 서버 메모리 조정

SQL Server Management Studio를 사용하여 최소 또는 최대 서버 메모리를 조정하려면 SQL Server 서비스를 다시 시작해야 합니다. 다음 코드를 사용하여 T-SQL(Transact SQL)을 사용하여 서버 메모리를 조정할 수도 있습니다.

```
EXECUTE sp_configure 'show advanced options', 1
GO
EXECUTE sp_configure 'min server memory (MB)', 2048
GO
EXEC sp_configure 'max server memory (MB)', 120832
GO
RECONFIGURE WITH OVERRIDE
```

## 비균일 메모리 액세스

NUMA(Non Uniform Memory Access)는 프로세서 버스의 추가 부하를 방지하는 메모리 액세스 최적화 기술입니다.

NUMA가 SQL Server가 설치된 서버에 구성되어 있는 경우 SQL Server가 NUMA를 인식하며 NUMA 하드웨어에서 제대로 작동하기 때문에 추가 구성이 필요하지 않습니다.

## 인덱스가 메모리를 만듭니다

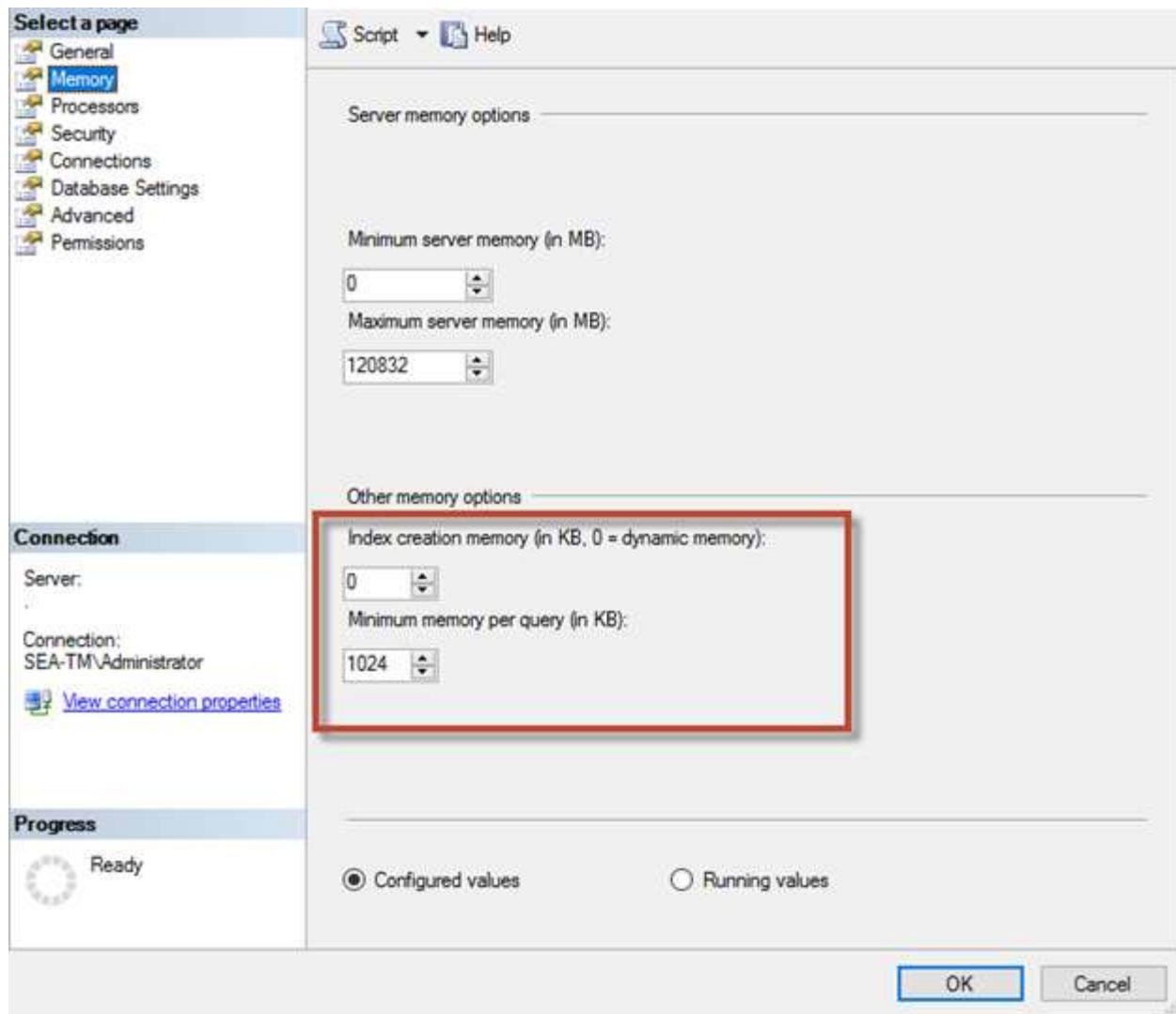
색인 메모리 만들기 옵션은 일반적으로 기본값에서 변경할 필요가 없는 또 다른 고급 옵션입니다.

인덱스 생성을 위해 처음에 할당된 최대 RAM 양을 제어합니다. 이 옵션의 기본값은 0입니다. 즉, 이 옵션은 SQL Server에서 자동으로 관리됩니다. 그러나 인덱스를 만드는 데 문제가 발생하면 이 옵션의 값을 늘리는 것이 좋습니다.

## 쿼리당 최소 메모리

쿼리를 실행하면 SQL Server는 효율적으로 실행할 최적의 메모리 양을 할당하려고 합니다.

기본적으로 쿼리당 최소 메모리 설정은 실행할 각 쿼리에 대해  $>= 1024KB$ 를 할당합니다. SQL Server에서 인덱스 생성 작업에 할당된 메모리 양을 동적으로 관리할 수 있도록 이 설정을 기본값으로 두는 것이 좋습니다. 그러나 SQL Server의 RAM이 효율적으로 실행하는 데 필요한 것보다 많은 경우 이 설정을 높이면 일부 쿼리의 성능이 향상될 수 있습니다. 따라서 SQL Server, 다른 응용 프로그램 또는 운영 체제에서 사용하지 않는 서버에서 메모리를 사용할 수 있는 한 이 설정을 높이면 SQL Server의 전반적인 성능이 향상될 수 있습니다. 사용 가능한 메모리가 없는 경우 이 설정을 늘리면 전체 성능이 저하될 수 있습니다.



## 공유 인스턴스와 전용 인스턴스 비교

SQL Server는 서버당 단일 인스턴스 또는 여러 인스턴스로 구성할 수 있습니다. 적절한 결정은 일반적으로 서버가 운영 또는 개발에 사용되는지 여부, 인스턴스가 비즈니스 운영 및 성능 목표에 중요한지를 여부 등의 요인에 따라 달라집니다.

처음에는 공유 인스턴스 구성이 더 쉬울 수 있지만 리소스가 나누어지거나 잠기는 문제가 발생할 수 있으며, 이로 인해 공유 SQL Server 인스턴스에서 데이터베이스가 호스팅되는 다른 응용 프로그램의 성능 문제가 발생할 수 있습니다.

어떤 인스턴스가 근본 원인인지 파악해야 하기 때문에 성능 문제의 해결은 복잡할 수 있습니다. 이 질문은 운영 체제 라이선스와 SQL Server 라이선스 비용을 기준으로 합니다. 애플리케이션 성능이 가장 중요한 경우에는 전용 인스턴스를 사용하는 것이 좋습니다.

Microsoft는 코어당 SQL Server의 라이센스를 인스턴스가 아닌 서버 레벨에서 부여합니다. 이러한 이유로 데이터베이스 관리자는 서버에서 처리할 수 있는 SQL Server 인스턴스를 많이 설치하여 라이센스 비용을 절감하려는 경향이 있으며, 이로 인해 나중에 주요 성능 문제가 발생할 수 있습니다.



\*NetApp는 최적의 성능을 얻기 위해 가능한 한 전용 SQL Server 인스턴스를 선택할 것을 권장합니다.

## tempdb 파일

Tempdb 데이터베이스를 많이 활용할 수 있습니다. ONTAP에 사용자 데이터베이스 파일을 최적으로 배치할 수 있을 뿐 아니라 할당 경합을 줄이기 위해 tempdb 데이터 파일을 배치하는 것도 중요합니다. tempdb는 별도의 디스크에 배치해야 하며 사용자 데이터 파일과 공유하지 않아야 합니다.

SQL Server가 새 개체를 할당하기 위해 특수 시스템 페이지에 써야 하는 경우 GAM(전역 할당 맵), SGAM(공유 전역 할당 맵) 또는 PFS(페이지 사용 가능 공간) 페이지에서 페이지 경합이 발생할 수 있습니다. 래치가 이러한 페이지를 메모리에 잠금니다. 사용 중인 SQL Server 인스턴스의 경우 tempdb의 시스템 페이지에 래치가 표시되는 데 시간이 오래 걸릴 수 있습니다. 이로 인해 쿼리 실행 시간이 느려지며 래치 경합이라고 합니다. tempdb 데이터 파일을 생성하는 방법은 다음 Best Practice를 참조하십시오.

- 또는 = 8코어: tempdb 데이터 파일 = 코어 수입니다
- 8코어 이상의 경우: 8tempdb 데이터 파일
- tempdb 데이터 파일은 같은 크기로 만들어야 합니다

다음 예제 스크립트는 크기가 같은 tempdb 파일 8개를 생성하고 tempdb를 SQL Server 2012 이상의 마운트 지점으로 이동하여 tempdb를 C:\MSSQL\tempdb 수정합니다.

```
use master

go

-- Change logical tempdb file name first since SQL Server shipped with
logical file name called tempdev

alter database tempdb modify file (name = 'tempdev', newname =
'tempdev01');

-- Change location of tempdev01 and log file

alter database tempdb modify file (name = 'tempdev01', filename =
'C:\MSSQL\tempdb\tempdev01.mdf');

alter database tempdb modify file (name = 'templog', filename =
'C:\MSSQL\tempdb\templog.ldf');

GO

-- Assign proper size for tempdev01

ALTER DATABASE [tempdb] MODIFY FILE ( NAME = N'tempdev01', SIZE = 10GB );

ALTER DATABASE [tempdb] MODIFY FILE ( NAME = N'templog', SIZE = 10GB );
```

```
GO
```

```
-- Add more tempdb files
```

```
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev02', FILENAME = N'C:\MSSQL\tempdb\tempdev02.ndf' , SIZE = 10GB , FILEGROWTH = 10%);
```

```
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev03', FILENAME = N'C:\MSSQL\tempdb\tempdev03.ndf' , SIZE = 10GB , FILEGROWTH = 10%);
```

```
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev04', FILENAME = N'C:\MSSQL\tempdb\tempdev04.ndf' , SIZE = 10GB , FILEGROWTH = 10%);
```

```
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev05', FILENAME = N'C:\MSSQL\tempdb\tempdev05.ndf' , SIZE = 10GB , FILEGROWTH = 10%);
```

```
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev06', FILENAME = N'C:\MSSQL\tempdb\tempdev06.ndf' , SIZE = 10GB , FILEGROWTH = 10%);
```

```
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev07', FILENAME = N'C:\MSSQL\tempdb\tempdev07.ndf' , SIZE = 10GB , FILEGROWTH = 10%);
```

```
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev08', FILENAME = N'C:\MSSQL\tempdb\tempdev08.ndf' , SIZE = 10GB , FILEGROWTH = 10%);
```

```
GO
```

SQL Server 2016부터 운영 체제에 표시되는 CPU 코어 수가 설치 중에 자동으로 감지되며, 이 수에 따라 SQL Server는 최적의 성능을 위해 필요한 tempdb 파일 수를 계산 및 구성합니다.

## AFF/FAS 시스템에 대한 스토리지 구성

### 개요

ONTAP 스토리지 솔루션과 Microsoft SQL Server를 결합하여 오늘날의 가장 까다로운 애플리케이션 요구사항을 충족할 수 있는 엔터프라이즈급 데이터베이스 스토리지 설계를 지원합니다.

SQL Server on ONTAP 솔루션을 최적화하려면 SQL Server I/O 패턴과 특성을 이해해야 합니다. SQL Server 데이터베이스용 스토리지 레이아웃은 SQL Server의 성능 요구사항을 지원하는 동시에 인프라 전체를 관리할 수 있는 기능을 극대화해야 합니다. 또한 우수한 스토리지 레이아웃을 통해 초기 구축을 성공적으로 수행할 수 있으며 비즈니스 성장에 따라 환경이 원활하게 확장될 수 있습니다.

## 데이터 스토리지 설계

SnapCenter를 사용하여 백업을 수행하지 않는 SQL Server 데이터베이스의 경우 데이터와 로그 파일을 별도의 드라이브에 배치하는 것이 좋습니다. 데이터를 동시에 업데이트하고 요청하는 응용 프로그램의 경우 로그 파일은 쓰기 작업이 많고 데이터 파일(응용 프로그램에 따라 다름)은 읽기/쓰기 작업이 많이 사용됩니다. 데이터 검색을 위해 로그 파일이 필요하지 않습니다. 따라서 자체 드라이브에 있는 데이터 파일에서 데이터 요청을 처리할 수 있습니다.

새 데이터베이스를 만들 때는 데이터와 로그에 대해 별도의 드라이브를 지정하는 것이 좋습니다. 데이터베이스를 만든 후 파일을 이동하려면 데이터베이스를 오프라인으로 전환해야 합니다. Microsoft 권장 사항에 대한 자세한 내용은 ["데이터 및 로그 파일을 별도의 드라이브에 저장합니다"](#)를 참조하십시오.

## 애그리게이트

애그리게이트는 NetApp 스토리지 구성에서 사용할 수 있는 최저 수준의 스토리지 컨테이너입니다. IO를 다른 기본 드라이브 세트로 분리할 것을 권장하는 일부 레거시 문서가 인터넷에 있습니다. ONTAP에서는 이 기능을 사용하지 않는 것이 좋습니다. NetApp은 데이터 파일 및 트랜잭션 로그 파일이 분리된 공유 및 전용 애그리게이트를 사용하여 다양한 I/O 워크로드 특성 테스트를 수행했습니다. 테스트 결과, 더 많은 RAID 그룹 및 드라이브를 포함하는 하나의 대형 Aggregate는 스토리지 성능을 최적화 및 개선했으며 다음과 같은 두 가지 이유로 관리자가 보다 쉽게 관리할 수 있는 것으로 나타났습니다.

- 하나의 대형 Aggregate를 통해 모든 드라이브의 I/O 기능을 모든 파일에서 사용할 수 있습니다.
- 하나의 대형 Aggregate는 디스크 공간을 가장 효율적으로 사용합니다.

고가용성(HA)을 위해 SQL Server Always On Availability Group 보조 동기식 복제본을 애그리게이트의 별도의 SVM(스토리지 가상 머신)에 배치합니다. 재해 복구를 위해 NetApp SnapMirror 기술을 사용하여 복제된 콘텐츠와 함께 DR 사이트에서 별도의 스토리지 클러스터의 일부인 애그리게이트에 비동기식 복제를 배치하십시오. NetApp은 최적의 스토리지 성능을 위해 애그리게이트에서 최소 10% 이상의 여유 공간을 사용할 것을 권장합니다.

## 볼륨

볼륨이 생성되어 애그리게이트 내에 상주합니다. 이 용어는 ONTAP 볼륨이 LUN이 아니기 때문에 혼동을 일으킬 수 있습니다. ONTAP 볼륨은 데이터를 위한 관리 컨테이너입니다. 볼륨에는 파일, LUN 또는 S3 오브젝트가 포함될 수 있습니다. 볼륨은 공간을 차지하지 않으며 포함된 데이터를 관리하는 데만 사용됩니다.

### 볼륨 설계 고려 사항

데이터베이스 볼륨 설계를 생성하기 전에 SQL Server 출력 패턴과 특성이 워크로드와 백업 및 복구 요구 사항에 따라 어떻게 다른지 이해해야 합니다. 확장 가능한 볼륨에 대한 다음 NetApp 권장 사항을 참조하십시오.

- 호스트 간에 볼륨을 공유하지 마십시오. 예를 들어, 단일 볼륨에 2개의 LUN을 생성하고 각 LUN을 다른 호스트와 공유할 수 있지만 관리가 복잡해질 수 있으므로 이러한 작업을 피해야 합니다. 동일한 호스트에서 여러 SQL Server 인스턴스를 실행하는 경우, 노드의 볼륨 제한에 근접하지 않으면 볼륨 공유를 피하고, 대신 데이터 관리를 쉽게 하기 위해 호스트당 개별 볼륨을 보유하는 것이 좋습니다.
- Windows의 26개 드라이브 문자 제한을 능가하려면 드라이브 문자 대신 NTFS 마운트 지점을 사용하십시오. 볼륨 마운트 지점을 사용할 때는 볼륨 레이블에 마운트 지점과 동일한 이름을 지정하는 것이 좋습니다.
- 필요한 경우 볼륨 자동 크기 조정 정책을 구성하여 공간 부족 상태를 방지하십시오.
- SMB 공유에 SQL Server를 설치하는 경우 SMB 볼륨에서 폴더를 생성할 때 유니코드가 설정되어 있는지 확인합니다.
- 운영 관점에서 쉽게 모니터링할 수 있도록 볼륨의 스냅샷 예비 공간 값을 0으로 설정합니다.
- 스냅샷 스케줄 및 보존 정책을 해제합니다. 대신 SnapCenter를 사용하여 SQL Server 데이터 볼륨의 스냅샷

복사본을 조정합니다.

- SQL Server 시스템 데이터베이스를 전용 볼륨에 배치합니다.
- tempdb는 SQL Server가 임시 작업 공간으로 사용하는 시스템 데이터베이스로, 특히 I/O를 많이 사용하는 DBCC CHECKDB 작업에 사용됩니다. 따라서 이 데이터베이스를 별도의 스핀들 세트가 있는 전용 볼륨에 배치하십시오. 볼륨 수가 문제가 되는 대규모 환경에서는 신중하게 계획을 수립한 후 tempdb를 더 적은 볼륨으로 통합하고 동일한 볼륨에 저장할 수 있습니다. SQL Server를 다시 시작할 때마다 이 데이터베이스가 다시 생성되므로 tempdb에 대한 데이터 보호는 높은 우선 순위가 아닙니다.
- (.mdf`랜덤 읽기/쓰기 워크로드이므로 사용자 데이터 파일을 별도의 볼륨에 배치 일반적으로 트랜잭션 로그 백업은 데이터베이스 백업보다 더 자주 생성됩니다. 따라서 트랜잭션 로그 파일 (.ldf)을 데이터 파일로부터 별도의 볼륨 또는 VMDK에 배치하여 각각에 대해 독립적인 백업 일정을 생성할 수 있도록 합니다. 또한 이 분리 방식은 로그 파일의 순차적 쓰기 I/O를 데이터 파일의 랜덤 읽기/쓰기 I/O에서 격리하고 SQL Server 성능을 크게 향상시킵니다.

## LUN을 클릭합니다

- 사용자 데이터베이스 파일과 로그 백업을 저장할 로그 디렉토리가 별도의 볼륨에 있어야 보존 정책이 SnapVault 기술과 함께 사용될 때 스냅샷을 덮어쓰지 않도록 할 수 있습니다.
- 전체 텍스트 검색 관련 파일과 같은 데이터베이스 파일과 데이터베이스를 제외한 파일을 동일한 LUN에 혼합하지 마십시오.
- 데이터베이스 보조 파일(파일 그룹의 일부로)을 별도의 볼륨에 배치하면 SQL Server 데이터베이스의 성능이 향상됩니다. 이 분리는 데이터베이스의 파일이 LUN을 다른 파일과 공유하지 않는 .mdf 경우에만 .mdf 유효합니다.
- DiskManager 또는 다른 툴을 사용하여 LUN을 생성하는 경우 LUN을 포맷할 때 파티션의 할당 단위 크기가 64K로 설정되어 있는지 확인하십시오.
- 를 참조하십시오 ["최신 SAN에 대한 ONTAP 모범 사례 하의 Microsoft Windows 및 네이티브 MPIO"](#) Windows에서 MPIO 속성의 iSCSI 장치에 다중 경로 지원을 적용하려면 다음을 수행합니다.

## 데이터베이스 파일 및 파일 그룹

초기 구축 단계에서는 ONTAP에 SQL Server 데이터베이스 파일을 적절하게 배치하는 것이 중요합니다. 따라서 비즈니스 요구 사항에 맞게 구성할 수 있는 최적의 성능, 공간 관리, 백업 및 복원 시간이 보장됩니다.

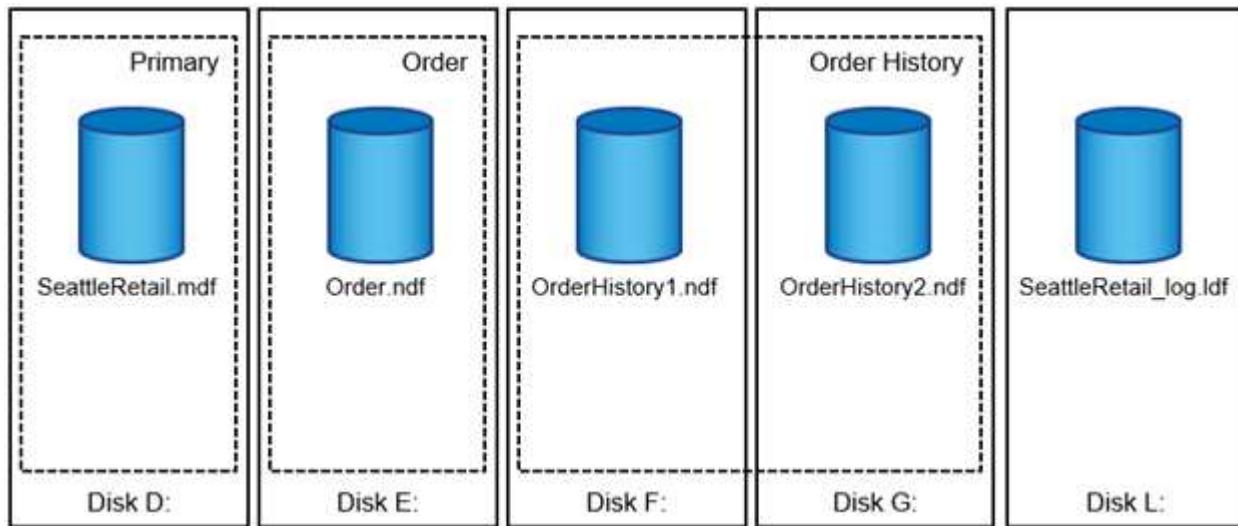
이론적으로 SQL Server(64비트)는 인스턴스당 32,767개의 데이터베이스와 524,272TB의 데이터베이스 크기를 지원하지만, 일반적인 설치에는 일반적으로 여러 개의 데이터베이스가 있습니다. 그러나 SQL Server에서 처리할 수 있는 데이터베이스 수는 로드 및 하드웨어에 따라 다릅니다. SQL Server 인스턴스가 수십, 수백 또는 수천 개의 소규모 데이터베이스를 호스팅하는 것은 드문 일이 아닙니다.

## 데이터베이스 파일 및 파일 그룹

각 데이터베이스는 하나 이상의 데이터 파일과 하나 이상의 트랜잭션 로그 파일로 구성됩니다. 트랜잭션 로그에는 데이터베이스 트랜잭션에 대한 정보와 각 세션에서 수행한 모든 데이터 수정에 대한 정보가 저장됩니다. 데이터가 수정될 때마다 SQL Server는 작업을 실행 취소(롤백)하거나 다시 실행(재생)할 수 있는 충분한 정보를 트랜잭션 로그에 저장합니다. SQL Server 트랜잭션 로그는 데이터 무결성과 견고성에 대한 SQL Server의 평판에 필수적인 부분입니다. 트랜잭션 로그는 SQL Server의 원자성, 일관성, 격리 및 내구성(ACID) 기능에 매우 중요합니다. SQL Server는 데이터 페이지가 변경되는 즉시 트랜잭션 로그에 기록합니다. 모든 DML(Data Manipulation Language) 문(예: SELECT, INSERT, UPDATE 또는 DELETE)은 완전한 트랜잭션이며, 트랜잭션 로그에서는 전체 집합 기반 작업이 수행되도록 하여 트랜잭션의 원자성을 확인합니다.

각 데이터베이스에는 기본 데이터 파일이 하나 있으며 기본적으로 확장명은 .mdf입니다. 또한 각 데이터베이스에는 보조 데이터베이스 파일이 있을 수 있습니다. 이러한 파일의 확장명은 기본적으로 .ndf입니다.

모든 데이터베이스 파일은 파일 그룹으로 그룹화됩니다. 파일 그룹은 논리적 단위로, 데이터베이스 관리를 간소화합니다. 논리 객체 배치와 물리적 데이터베이스 파일 간의 구분이 가능합니다. 데이터베이스 개체 테이블을 만들 때 기본 데이터 파일 구성에 대해 걱정하지 않고 파일 그룹을 배치할 파일 그룹을 지정합니다.



파일 그룹 내에 여러 데이터 파일을 배치할 수 있으므로 여러 스토리지 디바이스에 로드를 분산시킬 수 있으므로 시스템의 입출력 성능을 향상시킬 수 있습니다. 반면 SQL Server는 트랜잭션 로그에 순차적으로 기록하므로 트랜잭션 로그는 여러 파일의 이점을 얻지 못합니다.

파일 그룹에서 논리적 객체 배치와 물리적 데이터베이스 파일 간의 구분을 통해 데이터베이스 파일 레이아웃을 세밀하게 조정하여 스토리지 서브시스템에서 최대한 활용할 수 있습니다. 제공 워크로드를 지원하는 데이터 파일의 수는 애플리케이션에 영향을 주지 않고 I/O 요구사항과 예상 용량을 지원하기 위해 다양할 수 있습니다. 데이터베이스 레이아웃의 이러한 변형은 데이터베이스 파일이 아닌 파일 그룹에 데이터베이스 개체를 배치하는 응용 프로그램 개발자에게 영향을 주지 않습니다.

 \* NetApp는 \* 시스템 객체를 제외한 모든 항목에 대해 기본 파일 그룹을 사용하지 않을 것을 권장합니다. 사용자 객체에 대해 별도의 파일 그룹 또는 파일 집합을 만들면 특히 대규모 데이터베이스의 경우 데이터베이스 관리 및 재해 복구가 간소화됩니다.

#### 데이터베이스 인스턴스 파일 초기화

데이터베이스를 만들거나 기존 데이터베이스에 새 파일을 추가할 때 초기 파일 크기 및 자동 증가 매개 변수를 지정할 수 있습니다. SQL Server는 데이터를 기록할 데이터 파일을 선택할 때 비례 채우기 알고리즘을 사용합니다. 파일에서 사용할 수 있는 여유 공간에 비례하여 데이터의 양을 기록합니다. 파일의 여유 공간이 많을수록 처리하는 쓰기 횟수가 많아집니다.

 \* NetApp는 단일 파일 그룹에 있는 모든 파일의 초기 크기 및 자동 증가 매개 변수가 같고 증가 크기가 백분율이 아닌 메가바이트로 정의됨을 \* 권장합니다. 이렇게 하면 비례 채우기 알고리즘이 데이터 파일 간에 쓰기 작업의 균형을 고르게 유지할 수 있습니다.

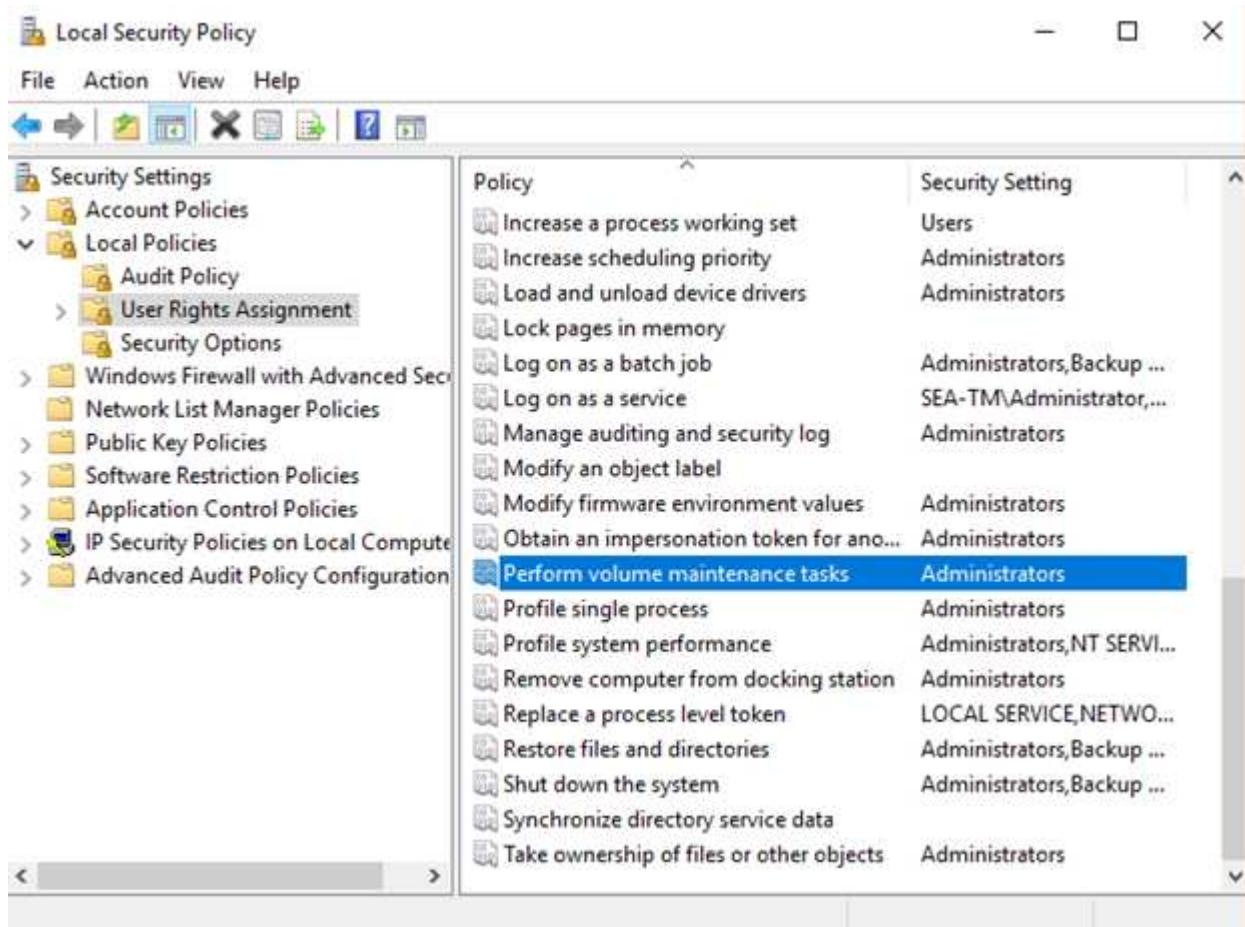
SQL Server는 파일을 늘릴 때마다 새로 할당된 공간을 0으로 채웁니다. 이 프로세스는 해당 파일에 기록해야 하는 모든 세션을 차단하거나 트랜잭션 로그가 증가하는 경우 트랜잭션 로그 레코드를 생성합니다.

SQL Server는 항상 트랜잭션 로그를 0으로 설정하며 이 동작은 변경할 수 없습니다. 그러나 인스턴트 파일 초기화를

사용하거나 사용하지 않도록 설정하여 데이터 파일의 제로화 여부를 제어할 수 있습니다. 즉각적인 파일 초기화를 사용하면 데이터 파일 증가 속도를 높이고 데이터베이스를 만들거나 복원하는 데 필요한 시간을 줄일 수 있습니다.

즉각적인 파일 초기화와 관련된 보안 위험이 있습니다. 이 옵션을 활성화하면 데이터 파일의 할당되지 않은 부분에 이전에 삭제된 OS 파일의 정보가 포함될 수 있습니다. 데이터베이스 관리자가 이러한 데이터를 검토할 수 있습니다.

SQL Server 시작 계정에 "볼륨 유지 관리 작업 수행"이라고도 하는 SA\_MANAGE\_VOLUME\_NAME 권한을 추가하여 즉각적인 파일 초기화를 활성화할 수 있습니다. 이 작업은 다음 그림과 같이 로컬 보안 정책 관리 응용 프로그램(secpol.msc)에서 수행할 수 있습니다. "볼륨 유지 관리 작업 수행" 권한에 대한 속성을 열고 SQL Server 시작 계정을 사용자 목록에 추가합니다.



사용 권한이 설정되어 있는지 확인하려면 다음 예제의 코드를 사용합니다. 이 코드는 SQL Server가 오류 로그에 추가 정보를 쓰고, 작은 데이터베이스를 만들고, 로그 내용을 읽도록 하는 두 개의 추적 플래그를 설정합니다.

```

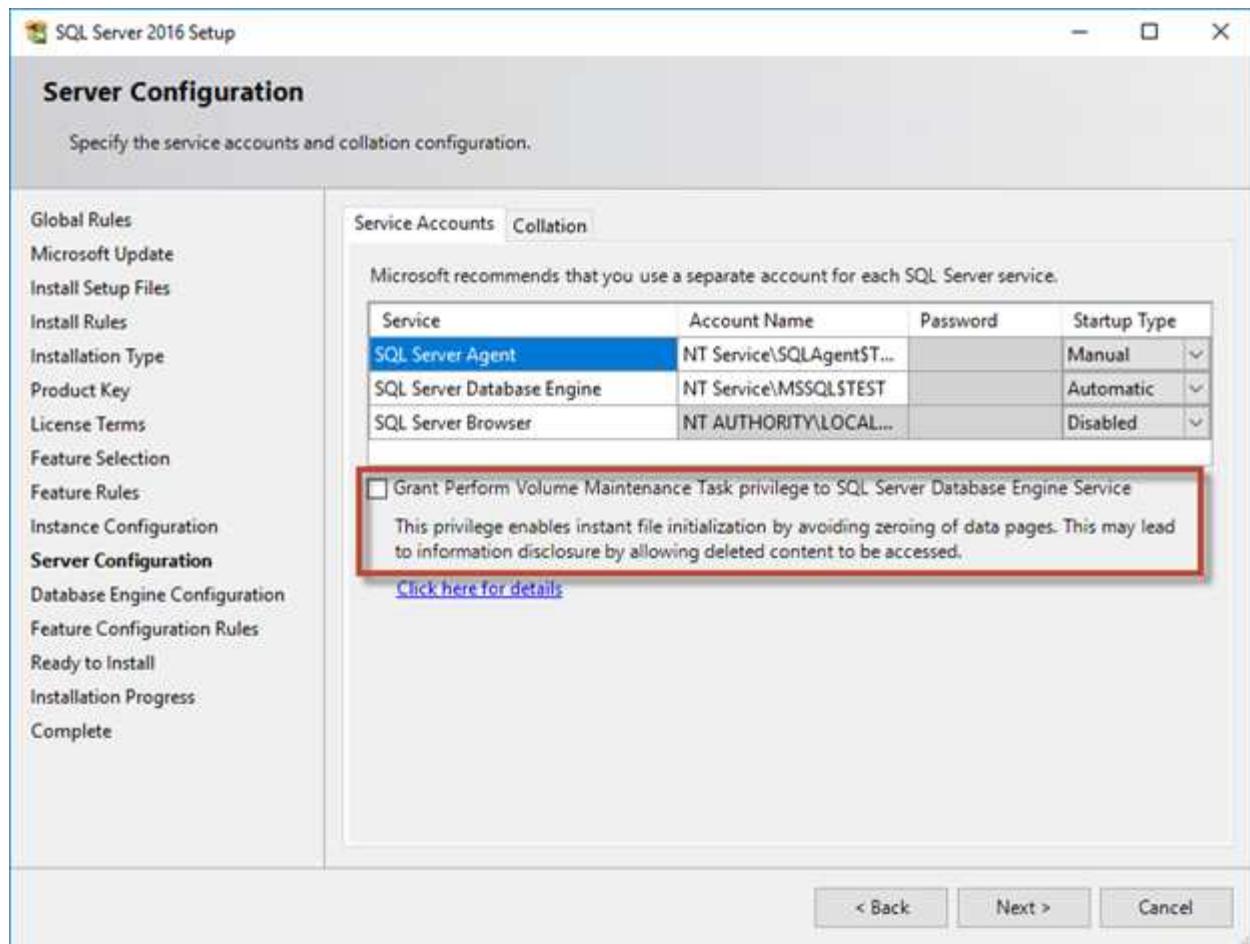
DBCC TRACEON(3004,3605,-1)
GO
CREATE DATABASE DelMe
GO
EXECUTE sp_readerrorlog
GO
DROP DATABASE DelMe
GO
DBCC TRACEOFF(3004,3605,-1)
GO

```

인스턴트 파일 초기화가 사용되지 않는 경우 SQL Server 오류 로그는 다음 예와 같이 SQL Server가 MDF 데이터 파일을 제로화하는 것 외에 LDF 로그 파일을 제로화하는 것을 보여 줍니다. 인스턴트 파일 초기화가 설정된 경우 로그 파일의 제로화만 표시됩니다.

	LogDate	ProcessInfo	Text
365	2017-02-09 08:10:07.660	spid53	Ckpt dbid 3 flush delta counts.
366	2017-02-09 08:10:07.660	spid53	Ckpt dbid 3 logging active xact info.
367	2017-02-09 08:10:07.750	spid53	Ckpt dbid 3 phase 1 ended (8)
368	2017-02-09 08:10:07.750	spid53	About to log Checkpoint end.
369	2017-02-09 08:10:07.880	spid53	Ckpt dbid 3 complete
370	2017-02-09 08:10:08.130	spid53	Starting up database 'DelMe'.
371	2017-02-09 08:10:08.150	spid53	Fixup Log Tail(progress) zeroing C:\Program Files\Microsoft SQL Server\MSSQL\DATA\DelMe.mdf
372	2017-02-09 08:10:08.160	spid53	Zeroing C:\Program Files\Microsoft SQL Server\MSSQL\DATA\DelMe_log.ldf
373	2017-02-09 08:10:08.170	spid53	Zeroing completed on C:\Program Files\Microsoft SQL Server\MSSQL\DATA\DelMe_log.ldf
374	2017-02-09 08:10:08.710	spid53	Ckpt dbid 6 started
375	2017-02-09 08:10:08.710	spid53	About to log Checkpoint begin.

볼륨 유지 관리 수행 작업은 SQL Server 2016에서 간소화되며 나중에 설치 프로세스 중에 옵션으로 제공됩니다. 다음 그림에서는 SQL Server 데이터베이스 엔진 서비스에 볼륨 유지 관리 작업을 수행할 수 있는 권한을 부여하는 옵션을 보여 줍니다.



데이터베이스 파일 크기를 제어하는 또 다른 중요한 데이터베이스 옵션은 자동 축소입니다. 이 옵션을 사용하면 SQL Server에서 정기적으로 데이터베이스 파일을 축소하고 크기를 줄이며 운영 체제에 공간을 해제합니다. 이 작업은 리소스를 많이 사용하며 새 데이터가 시스템에 유입될 때 일정 시간이 지난 후에 데이터베이스 파일이 다시 증가하기 때문에 거의 유용하지 않습니다. 데이터베이스에서 자동 축소를 사용하지 않아야 합니다.

#### 로그 디렉토리

로그 디렉토리는 트랜잭션 로그 백업 데이터를 호스트 레벨에서 저장하기 위해 SQL Server에 지정됩니다. SnapCenter를 사용하여 로그 파일을 백업하는 경우 SnapCenter에서 사용하는 각 SQL Server 호스트에 로그 백업을 수행하도록 구성된 호스트 로그 디렉토리가 있어야 합니다. SnapCenter에는 데이터베이스 저장소가 있으므로 백업, 복원 또는 클론 복제 작업과 관련된 메타데이터가 중앙 데이터베이스 저장소에 저장됩니다.

#### 호스트 로그 디렉토리의 크기는 다음과 같이 계산됩니다.

호스트 로그 디렉토리의 크기 = ((최대 DB LDF 크기 x 일일 로그 변경률 %) x (스냅샷 보존) ÷ (1 - LUN 오버헤드 공간 %))

호스트 로그 디렉토리 크기 조정 공식에서는 10%의 LUN 오버헤드 공간을 가정합니다

로그 디렉토리를 전용 볼륨 또는 LUN에 배치합니다. 호스트 로그 디렉토리의 데이터 양은 백업 크기 및 백업 보존 일수에 따라 달라집니다. SnapCenter는 SQL Server 호스트당 하나의 호스트 로그 디렉토리만 허용합니다. 호스트 로그 디렉토리는 SnapCenter → 호스트 → 플러그인 구성에서 구성할 수 있습니다.

- NetApp는 호스트 로그 디렉토리에 대해 다음을 권장합니다 \*.
- 호스트 로그 디렉토리가 백업 스냅샷 데이터를 손상시킬 수 있는 다른 유형의 데이터와 공유되지 않도록 하십시오.
- 마운트 지점을 호스팅하는 LUN에 사용자 데이터베이스나 시스템 데이터베이스를 배치하지 마십시오.
- SnapCenter에서 트랜잭션 로그를 복제할 전용 볼륨에 호스트 로그 디렉토리를 생성합니다.
- SnapCenter 마법사를 사용하여 데이터베이스를 NetApp 스토리지로 마이그레이션하여 데이터베이스가 유효한 위치에 저장되도록 함으로써 SnapCenter 백업 및 복구 작업을 성공적으로 수행할 수 있습니다. 마이그레이션 프로세스는 중단되며 마이그레이션이 진행 중인 동안 데이터베이스가 오프라인 상태가 될 수 있습니다.
- SQL Server의 FCI(Failover Cluster Instance)에 대해 다음 조건이 충족되어야 합니다.
  - 장애 조치 클러스터 인스턴스를 사용하는 경우 호스트 로그 디렉토리 LUN은 백업 중인 SQL Server 인스턴스와 동일한 클러스터 그룹에 있는 클러스터 디스크 리소스여야 SnapCenter 합니다.
  - 장애 조치 클러스터 인스턴스를 사용하는 경우 SQL Server 인스턴스와 연결된 클러스터 그룹에 할당된 물리적 디스크 클러스터 리소스인 공유 LUN에 사용자 데이터베이스를 배치해야 합니다.

## 스토리지 효율성

ONTAP 스토리지 효율성은 성능에 영향을 미치지 않고 최소한의 스토리지 공간을 사용하는 방식으로 SQL Server 데이터를 저장하고 관리하는 데 최적화되어 있습니다.

압축, 컴팩션, 중복제거와 같은 공간 효율성 기능은 지정된 양의 물리적 스토리지에 적합한 논리적 데이터의 양을 늘리기 위해 설계되었습니다. 결과적으로 비용과 관리 부담이 줄어듭니다.

상위 수준에서 압축은 수학적 프로세스이며, 그 패턴은 공간 요구사항을 감소시키는 방식으로 데이터를 감지하고 인코딩합니다. 이와 반대로, 중복제거는 실제로 반복되는 데이터 블록을 감지하여 불필요한 복사본을 제거합니다. 컴팩션을 사용하면 여러 개의 논리적 데이터 블록이 미디어에서 동일한 물리적 블록을 공유할 수 있습니다.

 스토리지 효율성과 부분 예약 간의 상호 작용에 대한 설명은 아래의 씬 프로비저닝에 대한 섹션을 참조하십시오.

## 압축

All-Flash 스토리지 시스템을 사용할 수 이전에는 어레이 기반 압축의 값이 제한되었습니다. 대부분의 I/O 집약적인 워크로드에는 허용되는 성능을 제공하기 위해 매우 많은 수의 스피드들이 필요했기 때문입니다. 스토리지 시스템에는 항상 많은 수의 드라이브가 부작용으로 필요한 것보다 훨씬 많은 용량이 포함되어 있습니다. 그러나 솔리드 스테이트 스토리지가 부상하면서 상황이 달라졌습니다. 이제 우수한 성능을 얻기 위해 드라이브를 엄청나게 오버 프로비저닝하지 않아도 됩니다. 스토리지 시스템의 드라이브 공간은 실제 용량 요구 사항과 일치할 수 있습니다.

솔리드 스테이트 드라이브(SSD)의 IOPS 용량이 증가하면 대개 회전식 드라이브에 비해 비용이 절감되며 압축 덕분에 솔리드 스테이트 미디어의 실제 용량이 늘어나 추가 절감을 달성할 수 있습니다.

데이터를 압축하는 방법에는 여러 가지가 있습니다. 대부분의 데이터베이스에는 자체 압축 기능이 포함되어 있지만 고객 환경에서는 이런 일이 거의 발생하지 않습니다. 그 이유는 일반적으로 압축된 데이터로 \*변경\* 시 성능 저하가 발생하며 일부 애플리케이션의 경우 데이터베이스 수준 압축에 대한 라이센스 비용이 많이 듭니다. 마지막으로, 데이터베이스

작업의 전반적인 성능에 영향을 미칩니다. 실제 데이터베이스 작업이 아닌 데이터 압축과 압축 해제를 수행하는 CPU를 위해 CPU당 라이센스 비용으로 높은 금액을 지불하는 것은 합리적이지 않습니다. 더 좋은 옵션은 압축 작업을 스토리지 시스템으로 오프로드하는 것입니다.

#### 적응형 압축

적응형 압축은 지연 시간이 마이크로초 단위로 측정되는 All-Flash 환경에서 조차 성능에 미치는 영향 없이 엔터프라이즈 워크로드로 철저히 테스트되었습니다. 심지어 일부 고객은 데이터가 캐시에 압축된 상태로 남아 있으므로 압축을 사용하여 성능이 향상되었다고 보고했습니다. 따라서 컨트롤러에서 가용 캐시의 양이 실질적으로 증가하기 때문입니다.

ONTAP는 4KB 유닛의 물리적 블록을 관리하며 적응형 압축은 기본 압축 블록 크기 8KB를 사용하며, 이는 데이터가 8KB 단위로 압축된다는 것을 의미합니다. 이 크기는 관계형 데이터베이스에서 가장 많이 사용되는 8KB 블록 크기와 일치합니다. 압축 알고리즘은 단일 유닛으로 더 많은 데이터가 압축되므로 효율성이 더욱 향상됩니다. 32KB의 압축 블록 크기는 8KB 압축 블록 유닛보다 더 공간 효율적입니다. 이는 기본 8KB 블록 크기를 사용하는 적응형 압축을 사용하면 효율성이 약간 낮지만 압축 블록 크기를 더 작게 만들면 큰 이점이 있습니다. 데이터베이스 워크로드에는 많은 양의 덮어쓰기 활동이 포함됩니다. 압축된 32KB 데이터 블록의 8KB를 덮어쓰려면 전체 32KB의 논리적 데이터를 다시 읽고, 압축을 풀고, 필요한 8KB 영역을 업데이트하고, 재압축을 수행한 다음 전체 32KB를 드라이브에 다시 써야 합니다. 이는 스토리지 시스템의 경우 매우 많은 비용이 드는 작업이며, 이로 인해 압축 블록 크기가 더 큰 경쟁 스토리지 어레이에서도 데이터베이스 워크로드의 성능이 크게 저하될 수 있습니다.

 적응형 압축에서 사용되는 블록 크기는 32KB까지 늘릴 수 있습니다. 이렇게 하면 스토리지 효율성이 향상될 수 있으며, 스토리지에 상당한 양의 데이터가 저장될 경우 트랜잭션 로그 및 백업 파일과 같은 대기 상태의 파일에 대해 고려해야 합니다. 경우에 따라 16KB 또는 32KB 블록 크기를 사용하는 액티브 데이터베이스가 이에 맞춰 적응형 압축의 블록 크기를 늘렸을 수도 있습니다. NetApp 또는 파트너 담당자에게 문의하여 이 솔루션이 현재 워크로드에 적합한지 여부를 확인하십시오.

 8KB보다 큰 압축 블록 크기는 스트리밍 백업 대상에서 중복제거와 함께 사용해서는 안 됩니다. 백업된 데이터의 작은 변화가 32KB 압축 기간에 영향을 미치기 때문입니다. 기간이 바뀌면 그에 따라 파일 전체에서 압축된 데이터가 달라집니다. 압축 후 중복제거가 발생하며, 이는 중복제거 엔진이 압축된 각 백업을 다르게 간주한다는 의미입니다. 스트리밍 백업의 중복제거가 필요한 경우 8KB 블록 적응형 압축만 사용해야 합니다. 더 작은 블록 크기를 사용할 수 있고 중복제거 효율성을 방해하지 않기 때문에 적응형 압축이 더 낫습니다. 유사한 이유로 호스트 측 압축도 중복제거 효율성에 지장을 줍니다.

#### 압축 정렬

데이터베이스 환경에서 적응형 압축을 수행할 때는 압축 블록 정렬과 관련된 몇 가지 사항을 고려해야 합니다. 이는 특정 블록의 랜덤 덮어쓰기가 데이터에 적용되는 경우만 해당합니다. 이 접근 방식은 파일 시스템의 시작이 4K 디바이스 경계에 맞춰 정렬되어야 하고 파일 시스템의 블록 크기가 4K의 배수여야 하는 전체 파일 시스템 정렬과 개념이 비슷합니다.

예를 들어, 파일에 대한 8KB 쓰기는 파일 시스템 자체 내에서 8KB 경계와 일치하는 경우에만 압축됩니다. 즉 파일의 첫 번째 8KB에, 두 번째 8KB에 그리고 그 이후로도 동일하게 포함되어야 합니다. 올바른 정렬을 보장하는 가장 간단한 방법은 올바른 LUN 유형을 사용하는 것입니다. 생성된 모든 파티션은 8K의 배수인 디바이스의 시작 부분에서 오프셋을 가지며 데이터베이스 블록 크기의 배수인 파일 시스템 블록 크기를 사용해야 합니다.

백업이나 트랜잭션 로그 같은 데이터는 압축된 여러 블록을 확장하는 순차적 쓰기 작업이며 따라서 정렬을 고려할 필요가 없습니다. I/O 패턴에서 고려해야 할 한 가지는 파일의 랜덤 덮어쓰기입니다.

#### 데이터 컴팩션

데이터 컴팩션은 압축 효율성을 향상하는 기술입니다. 앞서 설명한 것처럼, 적응형 압축만 사용했을 때는 절감 비율이 최대 2:1입니다. 4KB WAFL 블록에 8KB I/O를 저장하도록 제한되어 있기 때문입니다. 블록 크기가 더 큰 압축 방법을

통해 효율성이 향상됩니다. 그러나 이러한 복사본은 작은 블록 덮어쓰기가 적용되는 데이터에는 적합하지 않습니다. 32KB 단위 데이터의 압축 해제, 8KB 부분 업데이트, 재압축, 드라이브에 다시 쓰기 작업은 오버헤드를 발생시킵니다.

데이터 컴팩션은 여러 논리적 블록이 물리적 블록 내에 저장될 수 있게 합니다. 예를 들어, 텍스트 또는 부분 전체 블록과 같이 고도로 압축 가능한 데이터가 포함된 데이터베이스는 8KB에서 1KB로 압축될 수 있습니다. 컴팩션을 적용하지 않으면 이 1KB 데이터는 여전히 4KB 블록 전체를 점유할 것입니다. 인라인 데이터 컴팩션에서는 압축된 데이터 1KB를 다른 압축된 데이터와 함께 단 1KB의 물리적 공간에 저장할 수 있습니다. 이 방식은 압축 기술이 아니라 그저 드라이브의 공간을 더 효율적으로 할당하는 것이며 감지할 수 있는 성능 영향을 발생시키지 않습니다.

이로써 얻어지는 절감의 수준은 다양합니다. 이미 압축되었거나 암호화된 데이터는 일반적으로 더 압축할 수 없기 때문에 이 데이터 세트는 컴팩션의 이점을 얻지 못합니다. 반면 제로와 블록 메타데이터보다 조금 더 많이 포함하고 있으며 새롭게 초기화된 데이터 파일의 경우 80:1까지 압축합니다.

#### 온도에 민감한 스토리지 효율성

TSSE(Temperature Sensitive Storage Efficiency)는 ONTAP 9.8 이상에서 사용할 수 있습니다. 블록 액세스 히트 맵에 의존하여 자주 액세스하지 않는 블록을 식별하고 더 높은 효율성으로 압축합니다.

#### 중복 제거

중복 제거는 데이터 세트에서 중복된 블록 크기가 제거됩니다. 예를 들어, 동일한 4KB 블록이 10개 파일에 존재하면 중복제거는 파일 10개 전체에서 해당 4KB 블록을 동일한 4KB 물리적 블록으로 리디렉션합니다. 그 결과 데이터의 효율성이 10:1로 향상됩니다.

VMware 게스트 부팅 LUN과 같은 데이터는 동일한 운영 체제 파일의 여러 복사본으로 구성되어 있기 때문에 중복 제거가 매우 용이합니다. 100:1 이상의 효율성이 관찰되었습니다.

일부 데이터에 중복 데이터가 없습니다. 예를 들어, Oracle 블록에는 데이터베이스에 관해 전역적으로 고유한 헤더와 거의 고유한 트레일러가 포함되어 있습니다. 따라서 Oracle 데이터베이스의 중복 제거 기능을 사용하면 1%를 넘는 비용을 절감하는 경우는 거의 없습니다. MS SQL 데이터베이스의 중복 제거는 약간 더 낫지만 블록 수준에서 고유한 메타데이터는 여전히 제한 사항입니다.

일부 경우 16KB 및 대형 블록 크기의 데이터베이스에서 공간이 최대 15% 절약되었습니다. 각 블록의 처음 4KB에는 전역적으로 고유한 헤더가 포함되어 있고 마지막 4KB 블록에는 거의 고유한 트레일러가 포함되어 있습니다. 실제로는 거의 전적으로 제로화 데이터의 중복제거에 기인하지만 내부 블록은 중복제거 후보입니다.

많은 경쟁업체의 어레이는 데이터베이스가 여러 차례 복사된다는 추정을 기반으로 데이터베이스의 중복제거 기능을 내세웁니다. 이런 측면에서 NetApp 중복제거도 사용할 수 있지만 ONTAP은 더 나은 옵션인 NetApp FlexClone 기술을 제공합니다. 최종 결과는 같으며 기본 물리적 블록의 대부분을 공유하는 데이터베이스의 복사본이 여러 개 생성됩니다. FlexClone은 시간을 들여 데이터베이스 파일을 복사한 다음 중복제거하는 것보다 훨씬 더 효율적입니다. 실제로 이는 중복제거가 아니라 비중복이라 할 수 있습니다. 애초에 중복을 생성하지 않기 때문입니다.

#### 효율성 및 씬 프로비저닝

효율성 기능은 씬 프로비저닝의 한 형태입니다. 예를 들어, 100GB 볼륨을 점유하는 100GB LUN은 50GB까지 압축할 수 있을 것이고 볼륨은 여전히 100GB이기 때문에 실제로 절감이 실현되지는 않았습니다. 먼저 볼륨의 크기를 줄여 절감된 공간을 시스템의 어느 곳에서든 사용할 수 있게 해야 합니다. 나중에 100GB LUN으로 변경하면 데이터 압축률이 줄어들어 LUN 크기가 커지고 볼륨을 가득 채울 수 있습니다.

씬 프로비저닝은 관리를 단순화하는 동시에 가용 용량을 크게 개선하면서 비용을 절감할 수 있기 때문에 적극 권장합니다. 단순한 데이터베이스 환경에서 많은 빈 공간, 많은 수의 볼륨 및 LUN, 압축 가능한 데이터가 포함되는 경우가 많습니다. 일반 프로비저닝은 언젠가 100% 채워지고 100% 압축할 수 없는 데이터가 포함될 경우에 대비해 볼륨 및 LUN에 대한 스토리지 공간을 예약합니다. 그런 일은 일어나지 않을 것입니다. 씬 프로비저닝을 사용하면 공간을

재확보하고 다른 위치에서 사용할 수 있으며 더 작은 볼륨 및 LUN이 아닌 스토리지 시스템 자체를 기반으로 용량을 관리할 수 있습니다.

일부 고객은 특정 워크로드에 대해 또는 일반적으로 확립된 운영 및 조달 사례를 기반으로 일반 프로비저닝을 사용하는 것을 선호합니다.

 볼륨이 일반 프로비저닝되는 경우 명령을 사용한 중복제거 및 중복제거를 비롯한 해당 볼륨의 모든 효율성 기능을 완전히 비활성화하도록 주의를 기울여야 `sis undo` 합니다. 볼륨이 출력에 나타나지 않아야 `volume efficiency show` 합니다. 그렇지 않을 경우, 효율성 기능을 위해 볼륨이 부분적으로 구성됩니다. 결과적으로 덮어쓰기 보장은 서로 다르게 동작하므로 구성 과다 사용으로 인해 볼륨의 공간이 예기치 않게 부족해져서 데이터베이스 I/O 오류가 발생할 가능성이 높아집니다.

## 효율성 모범 사례

- NetApp는 \* 다음을 권장합니다.

## AFF 기본값

All-Flash AFF 시스템에서 실행되는 ONTAP에서 생성된 볼륨은 모든 인라인 효율성 기능을 사용하는 씬 프로비저닝됩니다. 일반적으로 데이터베이스에는 중복제거를 통해 이점을 얻을 수 없고 압축할 수 없는 데이터가 포함될 수 있지만 그럼에도 불구하고 기본 설정은 거의 모든 워크로드에 적합합니다. ONTAP는 절감 여부와 관계없이 모든 유형의 데이터와 I/O 패턴을 효율적으로 처리하도록 설계되었습니다. 원인을 완전히 이해하고 편차가 있는 경우에만 기본값을 변경해야 합니다.

## 일반 권장 사항

- 볼륨 및/또는 LUN이 씬 프로비저닝되지 않는 경우 절약 효과가 없고, 일반 프로비저닝과 공간 효율성이 설정된 조합을 통해 공간 부족 오류를 포함하여 예기치 않은 동작이 발생할 수 있으므로 모든 효율성 설정을 사용하지 않도록 설정해야 합니다.
- 백업 또는 데이터베이스 트랜잭션 로그와 같이 데이터를 덮어쓰지 않는 경우 냉각 기간이 짧은 TSSE를 활성화하여 효율성을 높일 수 있습니다.
- 일부 파일에는 압축할 수 없는 많은 양의 데이터가 포함되어 있을 수 있습니다. 예를 들어 파일의 응용 프로그램 수준에서 압축이 이미 활성화되어 있는 경우 암호화됩니다. 이러한 시나리오가 적용되는 경우 압축 데이터를 포함하는 다른 볼륨에서 더 효율적으로 작업할 수 있도록 압축을 해제하는 것이 좋습니다.
- 데이터베이스 백업에 32KB 압축 및 중복제거를 모두 사용하지 마십시오. 섹션을 참조하십시오 [적응형 압축](#) 를 참조하십시오.

## 데이터베이스 압축

SQL Server 자체에는 데이터를 압축하고 효율적으로 관리하는 기능도 있습니다. SQL Server는 현재 행 압축과 페이지 압축이라는 두 가지 유형의 데이터 압축을 지원합니다.

행 압축은 데이터 저장소 형식을 변경합니다. 예를 들어, 정수와 소수를 네이티브 고정 길이 형식 대신 가변 길이 형식으로 변경합니다. 또한 빈 공백을 제거하여 고정 길이 문자 문자열을 가변 길이 형식으로 변경합니다. 페이지 압축은 행 압축과 두 가지 다른 압축 전략(접두사 압축 및 사전 압축)을 구현합니다. 페이지 압축에 대한 자세한 내용은 에서 확인할 수 있습니다 ["페이지 압축 구현"](#).

데이터 압축은 현재 SQL Server 2008 이상의 Enterprise, Developer 및 Evaluation 에디션에서 지원됩니다. 데이터베이스가 자체적으로 압축을 수행할 수 있긴 하지만 SQL Server 환경에서는 이런 일이 거의 발생하지 않습니다.

다음은 SQL Server 데이터 파일의 공간을 관리하기 위한 권장 사항입니다

- SQL Server 환경에서 씬 프로비저닝을 사용하여 공간 사용률을 개선하고 공간 보장 기능을 사용할 때 전체 스토리지 요구 사항을 줄입니다.
  - 스토리지 관리자는 애그리게이트의 공간 사용량만 모니터링하면 되기 때문에 가장 일반적인 구축 구성에 대해 자동 확장 기능을 사용합니다.
- SQL Server 데이터 파일이 포함된 FAS의 모든 볼륨에서 중복 제거를 사용하도록 설정하지 마십시오. 단, 백업에 있는 데이터베이스를 단일 볼륨으로 복원하는 것과 같이 동일한 데이터의 여러 복사본이 볼륨에 포함되어 있는 것으로 알려져 있지 않습니다.

## 공간 재확보

LUN에서 사용되지 않는 공간을 복구하기 위해 공간 재확보를 주기적으로 시작할 수 있습니다. SnapCenter에서는 다음 PowerShell 명령을 사용하여 공간 재확보를 시작할 수 있습니다.

```
Invoke-SdHostVolumeSpaceReclaim -Path drive_path
```

공간 재확보를 실행해야 하는 경우 이 프로세스는 처음에 호스트의 주기를 소비하기 때문에 작업이 적은 기간 동안 실행해야 합니다.

## 데이터 보호

데이터베이스 백업 전략은 이론적인 기능이 아닌 식별된 비즈니스 요구 사항을 기반으로 해야 합니다. ONTAP의 스냅샷 기술을 결합하고 Microsoft SQL Server API를 활용하면 사용자 데이터베이스 크기에 관계없이 애플리케이션 정합성이 보장되는 백업을 신속하게 수행할 수 있습니다. 고급 또는 스케일아웃 데이터 관리 요구사항을 충족하기 위해 NetApp에서는 SnapCenter을 제공합니다.

## SnapCenter

SnapCenter는 엔터프라이즈 애플리케이션을 위한 NetApp 데이터 보호 소프트웨어입니다. SQL Server용 SnapCenter 플러그인과 Microsoft Windows용 SnapCenter 플러그인으로 관리되는 OS 작업을 통해 SQL Server 데이터베이스를 빠르고 쉽게 보호할 수 있습니다.

SQL Server 인스턴스는 독립 실행형 설치, 장애 조치 클러스터 인스턴스이거나 항상 가용성 그룹일 수 있습니다. 그 결과, 단일 창에서 데이터베이스를 보호, 클론 복제, 운영 또는 2차 복사본으로부터 복원할 수 있습니다. SnapCenter는 온프레미스, 클라우드 및 하이브리드 구성 모두에서 SQL Server 데이터베이스를 관리할 수 있습니다. 개발 또는 보고 용도로 데이터베이스 복사본을 조직 또는 대체 호스트에서 몇 분 내에 생성할 수도 있습니다.

또한 SQL Server는 생성 시 스냅샷에 올바른 데이터가 존재하도록 OS와 스토리지 간의 조정이 필요합니다. 대부분의 경우 이 작업을 수행하는 유일한 안전한 방법은 SnapCenter 또는 T-SQL을 사용하는 것입니다. 이 추가 조정 없이 생성된 스냅샷은 안정적으로 복구할 수 없습니다.

SnapCenter용 SQL Server 플러그인에 대한 자세한 내용은 ["TR-4714: NetApp SnapCenter를 사용하는 SQL Server 모범 사례 가이드"](#)를 참조하십시오.

## T-SQL 스냅샷을 사용하여 데이터베이스 보호

Microsoft는 SQL Server 2022에서 백업 작업의 스크립팅 및 자동화 경로를 제공하는 T-SQL 스냅샷을 도입했습니다. 전체 크기 복제본을 수행하는 대신 스냅샷을 위해 데이터베이스를 준비할 수 있습니다. 데이터베이스를 백업할 준비가 되면 ONTAP REST API를 활용하여 스냅샷을 생성할 수 있습니다.

다음은 샘플 백업 워크플로우입니다.

1. alter 명령으로 데이터베이스를 고정합니다. 이렇게 하면 기본 스토리지에서 정합성이 보장되는 스냅샷을 위해 데이터베이스가 준비됩니다. 고정 후에는 데이터베이스를 해제하고 backup 명령을 사용하여 스냅샷을 기록할 수 있습니다.
2. 새 백업 그룹 및 백업 서버 명령을 사용하여 스토리지 볼륨에서 여러 데이터베이스의 스냅샷을 동시에 수행합니다.
3. 전체 백업 또는 copy\_only 전체 백업을 수행합니다. 이러한 백업은 msdb에도 기록됩니다.
4. 스냅샷 전체 백업 후 일반 스트리밍 접근 방식으로 생성된 로그 백업을 사용하여 시점 복구를 수행합니다. 원하는 경우 스트리밍 차등 백업도 지원됩니다.

자세한 내용은 [을 참조하십시오 "T-SQL 스냅샷에 대한 Microsoft 설명서를 제공합니다".](#)

 \* NetApp는 SnapCenter를 사용하여 스냅샷 복사본을 생성할 것을 권장합니다 \*. 위에서 설명한 T-SQL 방법도 작동하지만 SnapCenter는 백업, 복원 및 복제 프로세스를 완벽하게 자동화합니다. 또한 검색을 수행하여 올바른 스냅샷이 생성되도록 합니다. 사전 구성이 필요하지 않습니다.

### **SnapCenter를 사용하는 SQL Server 가용성 그룹**

SnapCenter는 Windows 페일오버 클러스터로 구성된 SQL Server 가용성 그룹 데이터베이스의 백업을 지원합니다.

Microsoft SQL Server용 SnapCenter 플러그인은 Windows Server 장애 조치 클러스터의 모든 노드에 설치해야 합니다. ["문서화"](#)의 필수 구성 요소 및 SnapCenter 플러그인 설정 단계를 참조하십시오.

SnapCenter는 Windows 호스트 및 리소스의 모든 데이터베이스, 인스턴스 및 가용성 그룹을 검색하며 SnapCenter 리소스 페이지에 열거됩니다.

### **Always On 가용성 그룹의 데이터베이스 보호**

가용성 그룹의 데이터베이스는 여러 가지 방법으로 보호할 수 있습니다.

- 데이터베이스 레벨 백업: 데이터베이스 리소스 페이지에 사용할 가용성 데이터베이스를 선택하고 전체/로그 백업으로 구성된 정책을 추가하고 백업을 예약합니다. SnapCenter는 기본 복제본이든 보조 복제본이든 데이터베이스 역할에 관계없이 백업을 수행합니다. 리소스 그룹에 데이터베이스를 추가하여 보호를 구성할 수도 있습니다.
- 인스턴스 레벨 백업: 인스턴스를 선택하면 선택한 정책에 따라 인스턴스에서 실행되는 모든 데이터베이스가 보호됩니다. 기본 또는 보조 복제본으로 실행되는 가용성 데이터베이스를 포함한 모든 데이터베이스는 SnapCenter를 사용하여 백업됩니다. 리소스 그룹에 인스턴스를 추가하여 보호를 구성할 수도 있습니다.
- 가용성 그룹 레벨 백업: 정책을 구성하는 동안 SnapCenter에는 가용성 그룹 레벨 백업에 대한 고급 옵션이 있습니다. 정책의 가용성 그룹 설정을 통해 사용자는 백업을 위한 복제본 기본 설정을 선택할 수 있습니다. 1차, 2차 복제본 또는 모든 복제본을 선택할 수 있습니다. 기본 옵션은 SQL Server 가용성 그룹 구성의 백업 복제본 세트를 기반으로 합니다.

SnapCenter 정책의 가용성 그룹 설정은 가용성 그룹 수준 백업을 사용하여 가용성 그룹 데이터베이스를 보호하고 데이터베이스 또는 인스턴스 수준 백업에 적용되지 않는 경우에만 적용됩니다.

 \* NetApp는 가용성 수준 백업을 사용하여 NetApp ONTAP 스토리지에서 실행되는 모든 복제본에 백업할 것을 권장합니다.

## SnapCenter에서 로그 백업을 구성하는 중입니다

독립 실행형 SQL Server 설정에서 가용성 그룹이 설정된 경우 Windows Server 폐일오버 클러스터의 각 노드에 전용 디스크를 마운트해야 합니다. 전용 디스크를 사용하여 트랜잭션 로그 백업을 저장할 로그 디렉토리를 구성해야 합니다.

SQL Server 장애 조치 클러스터에 가용성 그룹이 설정되어 있으면 클러스터된 디스크가 SQL Server 장애 조치 클러스터 인스턴스에서 호스트 로그 디렉토리에 생성되어야 합니다.

## SnapCenter를 사용하여 가용성 그룹 설정에서 데이터베이스를 복원합니다

- SnapCenter는 보조 복제본에서 사용 가능한 최신 스냅샷으로부터 데이터베이스를 자동으로 복구하는 다시 시드 옵션을 제공합니다. 다시 시드 작업은 데이터베이스 백업을 가용성 그룹에 자동으로 복원하고 연결합니다.
- 가용성 그룹에서 복제본 데이터베이스를 복구하는 또 다른 방법은 가용성 그룹을 해제하고 전체 및 로그 복구를 수행하는 것입니다. SnapCenter를 사용하여 Norecovery 모드에서 데이터베이스를 복원한 다음 SQL Server Management Studio 또는 T-SQL을 사용하여 데이터베이스를 가용성 그룹에 다시 연결합니다.
- 데이터의 일부만 복구하기 위해 SnapCenter의 클론 기능을 사용하여 데이터베이스의 클론 복사본을 생성할 수 있습니다. 데이터베이스 복제본은 SnapCenter를 사용하여 몇 분 내에 생성한 다음 SQL Server 기본 툴을 사용하여 기본 복제본으로 데이터를 내보냅니다.

RTO 및 RPO 요구 사항을 충족하도록 데이터베이스 스토리지 레이아웃을 설정하는 Best Practice는 를 참조하십시오 "[TR-4714 NetApp SnapCenter를 사용하는 Microsoft SQL Server에 대한 모범 사례](#)".



SnapCenter는 분산된 가용성 그룹 및 포함된 가용성 그룹을 지원하지 않습니다.

## 재해 복구

### 재해 복구

엔터프라이즈 데이터베이스 및 애플리케이션 인프라에서는 다운타임을 최소화하면서 자연 재해 또는 예상치 못한 비즈니스 운영 중단으로부터 보호하기 위해 복제를 필요로 하는 경우가 많습니다.

SQL Server 상시 가용성 그룹 복제 기능은 탁월한 옵션일 수 있으며, NetApp는 데이터 보호를 상시 가동과 통합하는 옵션을 제공합니다. 그러나 ONTAP 복제 기술을 고려할 수도 있습니다. 세 가지 기본 옵션이 있습니다.

### SnapMirror를 참조하십시오

SnapMirror 기술은 LAN 및 WAN을 통한 데이터 복제를 위한 빠르고 유연한 엔터프라이즈 솔루션을 제공합니다. SnapMirror 기술은 초기 미러가 생성된 후에 변경된 데이터 블록만 타겟으로 전송하므로 네트워크 대역폭 요구사항을 크게 줄여줍니다. 동기 또는 비동기 모드로 구성할 수 있습니다.

### NetApp MetroCluster 및 SnapMirror 활성 동기화

많은 고객이 DR에는 데이터의 원격 복사본을 소유하는 것 이상의 것이 필요하며, 해당 데이터를 빠르게 활용할 수 있는 기능이 필요합니다. NetApp은 이러한 요구사항을 해결하는 두 가지 기술, 즉 MetroCluster와 SnapMirror Active Sync를 제공합니다.

MetroCluster는 낮은 수준의 동기식 미러링 스토리지와 수많은 추가 기능을 포함하는 하드웨어 구성에서 ONTAP를 가리킵니다. MetroCluster와 같은 통합 솔루션은 오늘날의 복잡한 스케일아웃 데이터베이스, 애플리케이션 및 가상화 인프라를 단순화합니다. 여러 외부 데이터 보호 제품 및 전략을 하나의 단순한 중앙 스토리지 시스템으로 대체합니다.

또한 단일 클러스터 스토리지 시스템 내에서 통합 백업, 복구, 재해 복구 및 고가용성(HA)을 제공합니다.

SnapMirror 활성 동기화는 SnapMirror Synchronous에 기반을 둡니다. MetroCluster를 사용할 경우 각 ONTAP 컨트롤러는 드라이브 데이터를 원격 위치로 복제하는 작업을 담당합니다. SnapMirror가 활성 동기화에서는 기본적으로 LUN 데이터의 독립적인 복사본을 유지하지만 해당 LUN의 단일 인스턴스를 제공하기 위해 협력하는 두 개의 ONTAP 시스템이 있습니다. 호스트 관점에서 보면 단일 LUN 엔티티입니다.

### SM-AS 및 MCC 비교

SM-AS와 MetroCluster는 전반적인 기능이 비슷하지만 RPO=0 복제 구현 방식과 관리 방식에는 중요한 차이점이 있습니다. SnapMirror 비동기식 및 동기식을 DR 계획의 일부로 사용할 수도 있지만, HA 보장 기술로 설계되지 않았습니다.

- MetroCluster 구성은 노드가 여러 사이트에 분산된 하나의 통합 클러스터와 비슷합니다. SM-AS는 특정 RPO=0 동기식으로 복제된 LUN을 지원하는 데 협력하고 있는 두 개의 다른 독립 클러스터처럼 동작합니다.
- MetroCluster 구성의 데이터는 한 번에 하나의 특정 사이트에서만 액세스할 수 있습니다. 데이터의 두 번째 사본이 반대쪽 사이트에 있지만 데이터는 수동적입니다. 스토리지 시스템 페일오버 없이는 액세스할 수 없습니다.
- MetroCluster 및 SM-AS 미러링 수행 수준은 다양합니다. MetroCluster 미러링은 RAID 계층에서 수행됩니다. 하위 수준 데이터는 SyncMirror를 사용하여 미러링된 형식으로 저장됩니다. LUN, 볼륨 및 프로토콜 계층에서는 미러링 사용이 거의 보이지 않습니다.
- 이에 반해 SM-AS 미러링은 프로토콜 계층에서 발생합니다. 두 클러스터는 전반적으로 독립적인 클러스터입니다. 두 개의 데이터 복제본이 동기화되면 두 클러스터에서 쓰기 작업만 미러링하면 됩니다. 한 클러스터에서 쓰기가 발생하면 다른 클러스터에 복제됩니다. 쓰기가 양쪽 사이트에서 완료된 경우에만 호스트에 인식됩니다. 이 프로토콜 분할 동작 이외에 두 클러스터는 정상적인 ONTAP 클러스터입니다.
- MetroCluster의 주요 역할은 대규모 복제입니다. RPO=0 및 제로에 가까운 RTO로 전체 스토리지를 복제할 수 있습니다. 이렇게 하면 장애 조치 프로세스가 간단해집니다. 장애 조치에는 한 가지 "장애 조치"만 필요하며 용량 및 IOPS 측면에서 매우 원활하게 확장됩니다.
- SM-AS의 주요 활용 사례 중 하나는 세분화된 복제입니다. 모든 데이터를 단일 유닛으로 복제하기 원하지 않거나 특정 워크로드를 선택적으로 페일오버할 수 있어야 하는 경우가 있습니다.
- SM-AS의 또 다른 주요 활용 사례는 Active-Active 작업이며, 성능 특성이 동일한 두 위치에 있는 서로 다른 두 클러스터에서 완전히 사용할 수 있는 데이터 복제본을 사용할 수 있도록 하고, 원하는 경우 사이트 간에 SAN을 확장할 필요가 없습니다. 두 사이트에서 애플리케이션을 이미 실행할 수 있으므로 페일오버 작업 중에 전체 RTO가 감소합니다.

### SnapMirror를 참조하십시오

다음은 SQL Server용 SnapMirror에 대한 권장사항입니다.

- SMB를 사용하는 경우 대상 SVM은 소스 SVM이 속한 동일한 Active Directory 도메인의 구성원이어야 NAS 파일 내에 저장된 ACL(액세스 제어 목록)이 재해로부터 복구하는 동안 손상되지 않습니다.
- 소스 볼륨 이름과 동일한 대상 볼륨 이름을 사용할 필요는 없지만, 대상 볼륨을 대상에 마운트하는 프로세스를 더 간편하게 관리할 수 있습니다. SMB를 사용하는 경우 소스 네임스페이스와 경로 및 디렉토리 구조의 대상 NAS 네임스페이스를 동일하게 만들어야 합니다.
- 일관성을 위해 컨트롤러에서 SnapMirror 업데이트를 예약하지 마십시오. 대신, 전체 또는 로그 백업이 완료된 후 SnapCenter에서 SnapMirror를 업데이트하도록 SnapMirror를 활성화합니다.
- SQL Server 데이터가 포함된 볼륨을 클러스터의 서로 다른 노드에 분산하여 모든 클러스터 노드가 SnapMirror 복제 활동을 공유할 수 있도록 합니다. 이러한 분산은 노드 리소스 사용을 최적화합니다.
- 빠른 데이터 복구에 대한 수요가 많은 경우 동기식 복제를 사용하고 RPO에 유연성을 제공하는 비동기식 솔루션을

사용합니다.

SnapMirror에 대한 자세한 내용은 를 참조하십시오 ["TR-4015:ONTAP 9용 SnapMirror 구성 및 모범 사례 가이드"](#).

## MetroCluster

있습니다

MetroCluster 환경에서 Microsoft SQL Server를 구축하려면 MetroCluster 시스템의 물리적 설계에 대해 몇 가지 설명이 필요합니다.

MetroCluster는 별도의 위치 또는 장애 도메인에 있는 두ONTAP 클러스터 간에 데이터와 구성의 동기식으로 미러링합니다. MetroCluster은 두 가지 목표를 자동으로 관리하여 애플리케이션에 대해 지속적으로 사용 가능한 스토리지를 제공합니다.

- 클러스터에 기록된 데이터를 동기식으로 미러링하여 복구 시점 목표(RPO)를 0으로 설정합니다.
- 구성을 미러링하고 두 번째 사이트에서 데이터 액세스를 자동화하여 RTO(복구 시간 목표)가 거의 없는 환경입니다.

MetroCluster는 두 사이트에 있는 두 개의 독립적인 클러스터 간에 데이터와 구성의 자동으로 미러링하여 단순성 기능을 제공합니다. 스토리지가 한 클러스터 내에서 프로비저닝되면 두 번째 사이트의 두 번째 클러스터에 자동으로 미러링됩니다. NetApp SyncMirror ®는 제로 RPO로 모든 데이터의 전체 복사본을 제공합니다. 즉, 한 사이트의 워크로드가 언제든지 반대쪽 사이트로 전환하고 데이터 손실 없이 데이터를 계속 제공할 수 있습니다. MetroCluster는 2차 사이트에서 NAS와 SAN 프로비저닝된 데이터에 대한 액세스를 제공하는 스위치오버 프로세스를 관리합니다. 검증된 솔루션으로 설계된 MetroCluster에는 프로토콜 시간 초과 기간 내에 또는 대개 120초 미만으로 전환할 수 있는 크기 조정과 구성이 포함되어 있습니다. 따라서 RPO가 거의 0에 가까우며 애플리케이션이 장애 발생 없이 데이터에 계속 액세스할 수 있습니다. MetroCluster는 백엔드 스토리지 패브릭에 정의된 다양한 버전으로 사용할 수 있습니다.

**MetroCluster는 3가지 구성으로 사용할 수 있습니다**

- IP 연결이 포함된 HA 쌍
- FC 연결이 포함된 HA 쌍
- FC 연결이 포함된 단일 컨트롤러



'접속'이라는 용어는 사이트 간 복제에 사용되는 클러스터 접속을 의미합니다. 호스트 프로토콜을 참조하지 않습니다. 모든 호스트측 프로토콜은 클러스터 간 통신에 사용되는 연결 유형에 관계없이 MetroCluster 구성에서 평소와 같이 지원됩니다.

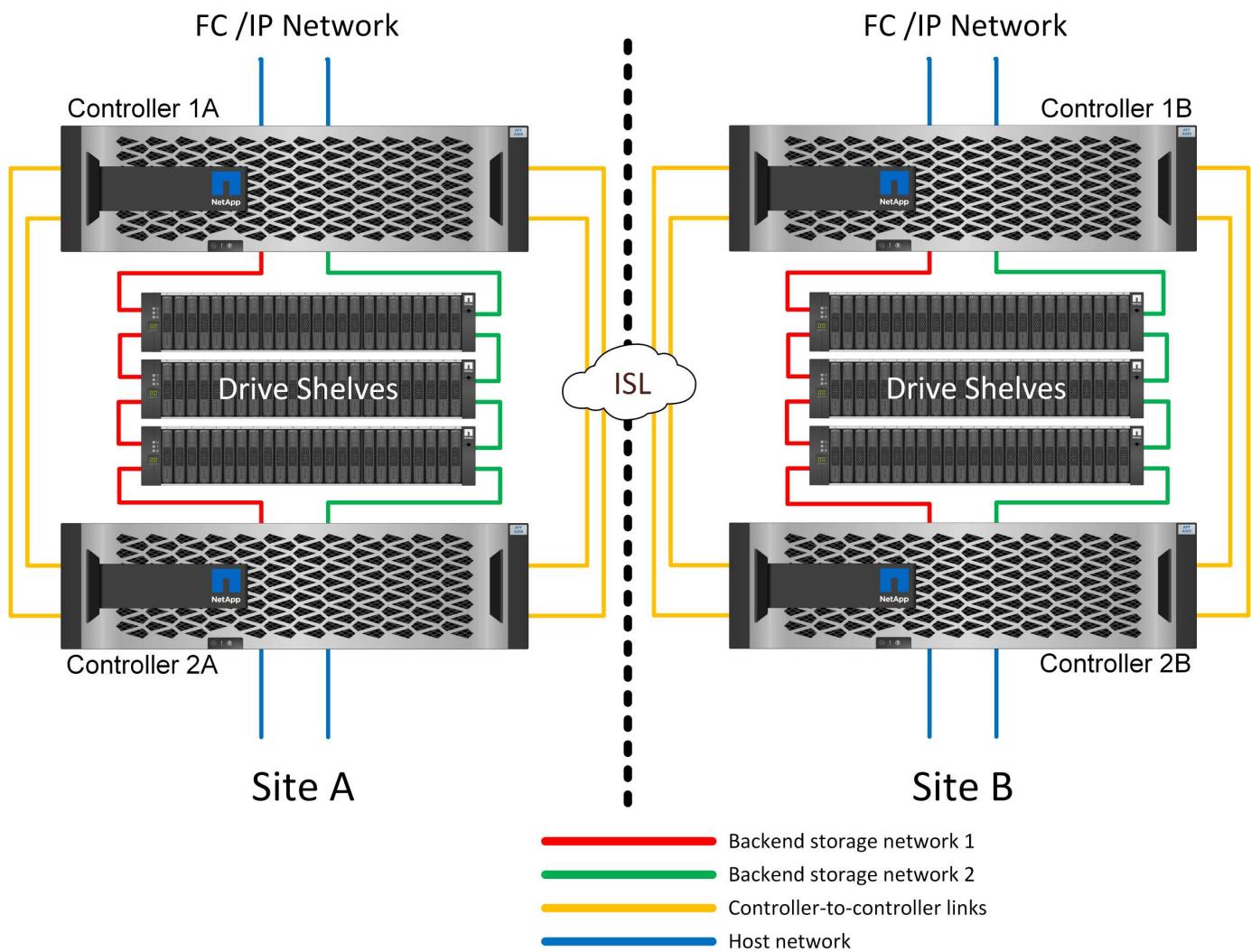
**MetroCluster IP를 선택합니다**

HA 쌍 MetroCluster IP 구성은 사이트당 2~4개의 노드를 사용합니다. 이 구성 옵션은 2노드 옵션에 비해 복잡성과 비용을 증가시키지만, 내부 중복이라는 중요한 이점을 제공합니다. 컨트롤러 장애가 간단하더라도 WAN을 통한 데이터 액세스가 필요하지 않습니다. 데이터 액세스는 대체 로컬 컨트롤러를 통해 로컬에 유지됩니다.

대부분의 고객은 인프라 요구 사항이 더 간단하기 때문에 IP 연결을 선택하고 있습니다. 과거에는 다크 파이버 및 FC 스위치를 사용하여 고속 사이트 간 연결을 제공하기가 일반적으로 더 쉬웠지만, 오늘날의 고속, 짧은 지연 시간 IP 회로는 보다 쉽게 사용할 수 있었습니다.

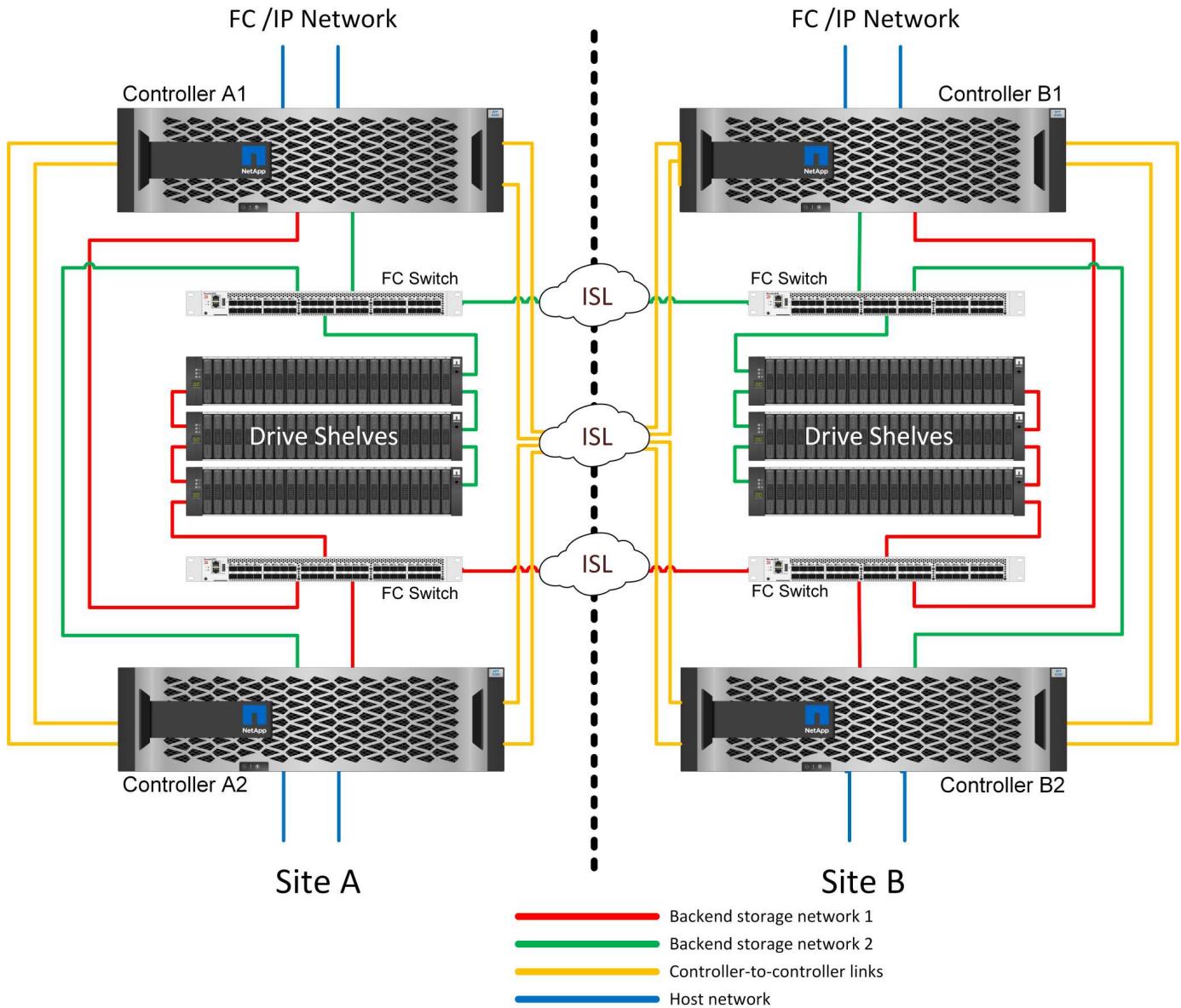
또한 사이트 간 연결만 컨트롤러를 위한 것이므로 아키텍처가 더욱 단순합니다. FC SAN 연결 MetroCluster에서 컨트롤러는 반대쪽 사이트의 드라이브에 직접 기록하므로 SAN 연결, 스위치 및 브리지가 추가로 필요합니다. 반면, IP 구성의 컨트롤러는 컨트롤러를 통해 반대쪽 드라이브에 씁니다.

자세한 내용은 공식 ONTAP 설명서 및 [MetroCluster IP 솔루션 아키텍처 및 설계](#)를 참조하십시오.



### HA-쌍 FC SAN 연결 MetroCluster

HA 쌍 MetroCluster FC 구성은 사이트당 2개 또는 4개의 노드를 사용합니다. 이 구성 옵션은 2노드 옵션에 비해 복잡성과 비용을 증가시키지만, 내부 중복이라는 중요한 이점을 제공합니다. 컨트롤러 장애가 간단하더라도 WAN을 통한 데이터 액세스가 필요하지 않습니다. 데이터 액세스는 대체 로컬 컨트롤러를 통해 로컬에 유지됩니다.

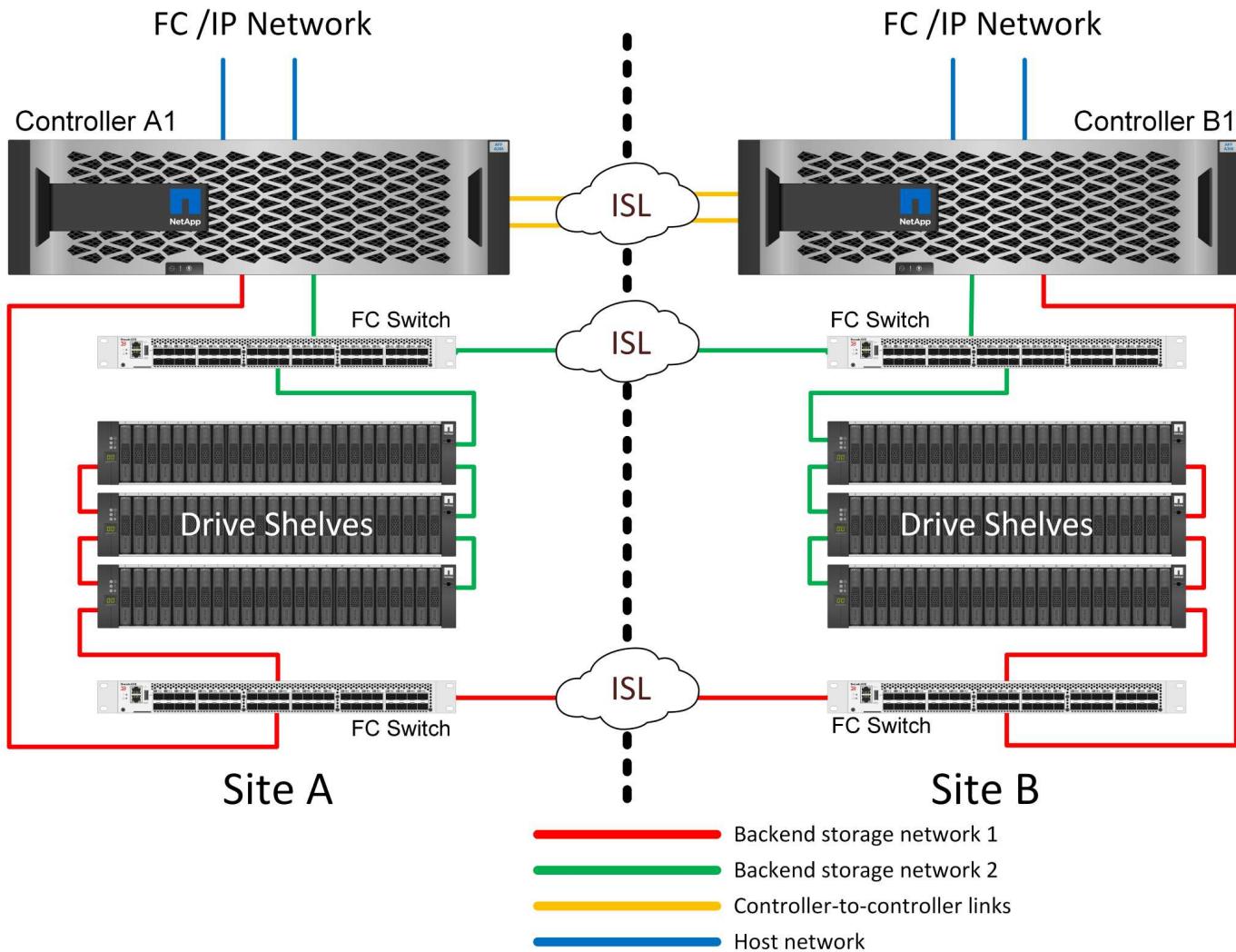


일부 멀티사이트 인프라는 액티브-액티브 운영을 위해 설계되지 않았지만 운영 사이트 및 재해 복구 사이트로 더 많이 사용됩니다. 이 상황에서 HA 쌍 MetroCluster 옵션이 일반적으로 다음과 같은 이유로 더 권장됩니다.

- 2노드 MetroCluster 클러스터는 HA 시스템이지만, 컨트롤러의 예상치 못한 장애나 계획된 유지 관리를 위해서는 반대쪽 사이트에서 데이터 서비스를 온라인으로 전환해야 합니다. 사이트 간 네트워크 연결이 필요한 대역폭을 지원할 수 없는 경우 성능이 영향을 받습니다. 유일한 옵션은 다양한 호스트 OS 및 관련 서비스를 대체 사이트로 폐일오버하는 것입니다. HA Pair MetroCluster 클러스터는 동일한 사이트 내에서 단순한 폐일오버가 발생하기 때문에 이 문제가 해소됩니다.
- 일부 네트워크 토플로지는 사이트 간 액세스용으로 설계되지 않은 대신 서로 다른 서브넷이나 격리된 FC SAN을 사용합니다. 이런 경우 2노드 MetroCluster 클러스터는 다른 사이트의 서버에 데이터를 제공할 수 없기 때문에 더 이상 HA 시스템으로 작동하지 않습니다. 완벽한 이중화를 제공하려면 HA Pair MetroCluster 옵션이 필요합니다.
- 2개 사이트 인프라를 고가용성 단일 인프라로 간주하는 경우 2노드 MetroCluster 구성이 적합합니다. 하지만 사이트 장애 후 시스템이 오랫동안 작동해야 하는 경우에는 단일 사이트 내에서 HA를 계속 제공하기 때문에 HA 2노드가 선호됩니다.

## 2노드 FC SAN 연결 MetroCluster

2노드 MetroCluster 구성은 사이트당 하나의 노드만 사용합니다. 이 설계는 구성과 유지 관리가 필요한 구성 요소가 적기 때문에 HA 쌍 옵션보다 단순합니다. 또한 케이블 연결 및 FC 스위칭에 대한 인프라 요구도 줄었습니다. 마지막으로 비용을 절감할 수 있습니다.



이 설계의 분명한 영향은 단일 사이트에서 컨트롤러 장애가 발생하면 반대쪽 사이트에서 데이터를 사용할 수 있다는 것입니다. 이러한 제한이 반드시 문제가 되는 것은 아닙니다. 많은 기업은 기본적으로 단일 인프라로 작동하는 지연 시간이 짧은 확장된 고속 네트워크를 통해 멀티사이트 데이터 센터를 운영하고 있습니다. 이 경우 MetroCluster의 2노드 버전을 사용하는 것이 좋습니다. 현재 여러 서비스 공급자가 두 노드 시스템을 페타바이트 규모로 사용하고 있습니다.

## MetroCluster 복원력 기능

MetroCluster 솔루션에는 단일 장애 지점이 없습니다.

- 각 컨트롤러에는 로컬 사이트의 드라이브 웰프에 대한 2개의 독립적 경로가 있습니다.
- 각 컨트롤러에는 원격 사이트의 드라이브 웰프에 대한 두 개의 독립적 경로가 있습니다.
- 각 컨트롤러에는 반대쪽 사이트에 있는 컨트롤러에 대한 독립적인 경로가 2개 있습니다.
- HA 쌍 구성에서 각 컨트롤러에는 로컬 파트너에 대한 두 가지 경로가 있습니다.

요약하면, MetroCluster의 데이터 제공 기능에 영향을 주지 않으면서 구성의 모든 구성 요소를 제거할 수 있습니다. 두 옵션 간의 복원력에서 유일한 차이점은 HA 쌍 버전이 사이트 장애 발생 후 전체 HA 스토리지 시스템이라는 점입니다.

## SyncMirror

MetroCluster를 사용하는 SQL Server에 대한 보호는 최대 성능, 스케일아웃 동기식 미러링 기술을 제공하는 SyncMirror를 기반으로 합니다.

### SyncMirror를 사용한 데이터 보호

가장 간단한 수준인 동기식 복제는 미러링된 스토리지의 양쪽에서 변경 사항이 확인되기 전에 수행되어야 함을 의미합니다. 예를 들어, 데이터베이스에서 로그를 작성하거나 VMware 게스트에 패치를 적용하는 경우 쓰기가 손실되지 않아야 합니다. 프로토콜 레벨에서 스토리지 시스템은 두 사이트의 비휘발성 미디어에 커밋될 때까지 쓰기를 인증해서는 안 됩니다. 그래야만 데이터 손실의 위험 없이 진행하는 것이 안전합니다.

동기식 복제 솔루션을 설계하고 관리하는 첫 번째 단계는 동기식 복제 기술을 사용하는 것입니다. 가장 중요한 고려 사항은 계획된 고장 시나리오와 예상치 못한 다양한 장애 시나리오 중에 발생할 수 있는 상황을 이해하는 것입니다. 모든 동기식 복제 솔루션이 동일한 기능을 제공하는 것은 아닙니다. 데이터 손실이 0인 복구 지점 목표(RPO)를 제공하는 솔루션이 필요한 경우 모든 장애 시나리오를 고려해야 합니다. 특히 사이트 간 연결 손실로 인해 복제가 불가능할 때 예상되는 결과는 무엇입니까?

### SyncMirror 데이터 가용성

MetroCluster 복제는 NetApp SyncMirror 기술을 기반으로 하며 동기식 모드로 효율적으로 전환하거나 아웃하도록 설계되었습니다. 이 기능은 동기식 복제를 필요로 하지만 데이터 서비스를 위해 고가용성이 필요한 고객의 요구사항을 충족합니다. 예를 들어 원격 사이트에 대한 연결이 끊어진 경우 일반적으로 스토리지 시스템이 복제되지 않은 상태로 계속 작동하도록 하는 것이 좋습니다.

대부분의 동기식 복제 솔루션은 동기식 모드에서만 작동할 수 있습니다. 이러한 유형의 모든 또는 무관 복제를 도미노 모드라고도 합니다. 이러한 스토리지 시스템은 데이터의 로컬 및 원격 복제본이 동기화되지 않도록 하는 대신 데이터 제공을 중지합니다. 복제가 강제로 중단되면 재동기화는 시간이 매우 오래 걸리고 미러링이 다시 설정되는 동안 고객이 완전한 데이터 손실에 노출되도록 할 수 있습니다.

원격 사이트에 연결할 수 없는 경우 SyncMirror가 동기식 모드를 원활하게 전환할 수 있을 뿐만 아니라 연결이 복원되면 RPO=0 상태로 빠르게 다시 동기화할 수 있습니다. 또한 재동기화 중에 원격 사이트의 오래된 데이터 복제본을 사용 가능한 상태로 유지할 수 있으므로 데이터의 로컬 및 원격 복제본이 항상 존재합니다.

도미노 모드가 필요한 경우 NetApp은 SnapMirror Synchronous(SM-S)를 제공합니다. Oracle DataGuard 또는 SQL Server Always On Availability Groups와 같은 애플리케이션 레벨 옵션도 있습니다. OS 수준 디스크 미러링은 옵션이 될 수 있습니다. 자세한 정보와 옵션은 NetApp 또는 파트너 계정 팀에 문의하십시오.

### MetroCluster이 포함된 SQL Server

제로 RPO로 SQL Server 데이터베이스를 보호하는 한 가지 옵션은 MetroCluster입니다. MetroCluster는 사이트 간에 전체 인프라를 쉽게 복제할 수 있는 단순한 고성능 RPO = 0 복제 기술입니다.

SQL Server는 단일 MetroCluster 시스템에서 최대 수천 개의 데이터베이스로 확장할 수 있습니다. SQL Server 독립 실행형 인스턴스 또는 장애 조치 클러스터 인스턴스가 있을 수 있습니다. MetroCluster 시스템은 데이터베이스 관리에 대한 모범 사례를 추가하거나 변경하지 않을 수도 있습니다.

MetroCluster에 대한 완전한 설명은 본 문서의 범위를 벗어나지만 원칙은 간단합니다. MetroCluster는 신속한 페일오버로 RPO=0 복제 솔루션을 제공할 수 있습니다. 이 기반을 바탕으로 구축하는 것은 요구 사항에 따라 다릅니다.

예를 들어 갑작스러운 사이트 손실 후 기본 빠른 DR 절차에서는 다음과 같은 기본 단계를 사용할 수 있습니다.

- MetroCluster switchover를 강제 적용합니다
- FC/iSCSI LUN 검색 수행(SAN만 해당)
- 파일 시스템을 마운트합니다
- SQL 서비스를 시작합니다

이 방법의 주요 요구 사항은 원격 사이트에서 실행 중인 OS입니다. SQL Server 설치 시 사전 구성되어야 하며 동일한 빌드 버전으로 업데이트해야 합니다. 또한 SQL Server 시스템 데이터베이스는 원격 사이트로 미러링되고 재해가 선언된 경우 마운트할 수 있습니다.

가상화 데이터베이스를 호스팅하는 볼륨, 파일 시스템 및 데이터 저장소가 전환 전에 재해 복구 사이트에서 사용되지 않는 경우에는 관련 볼륨에 설정할 필요가 없습니다. `dr-force-nvfail`

## **SnapMirror 활성 동기화**

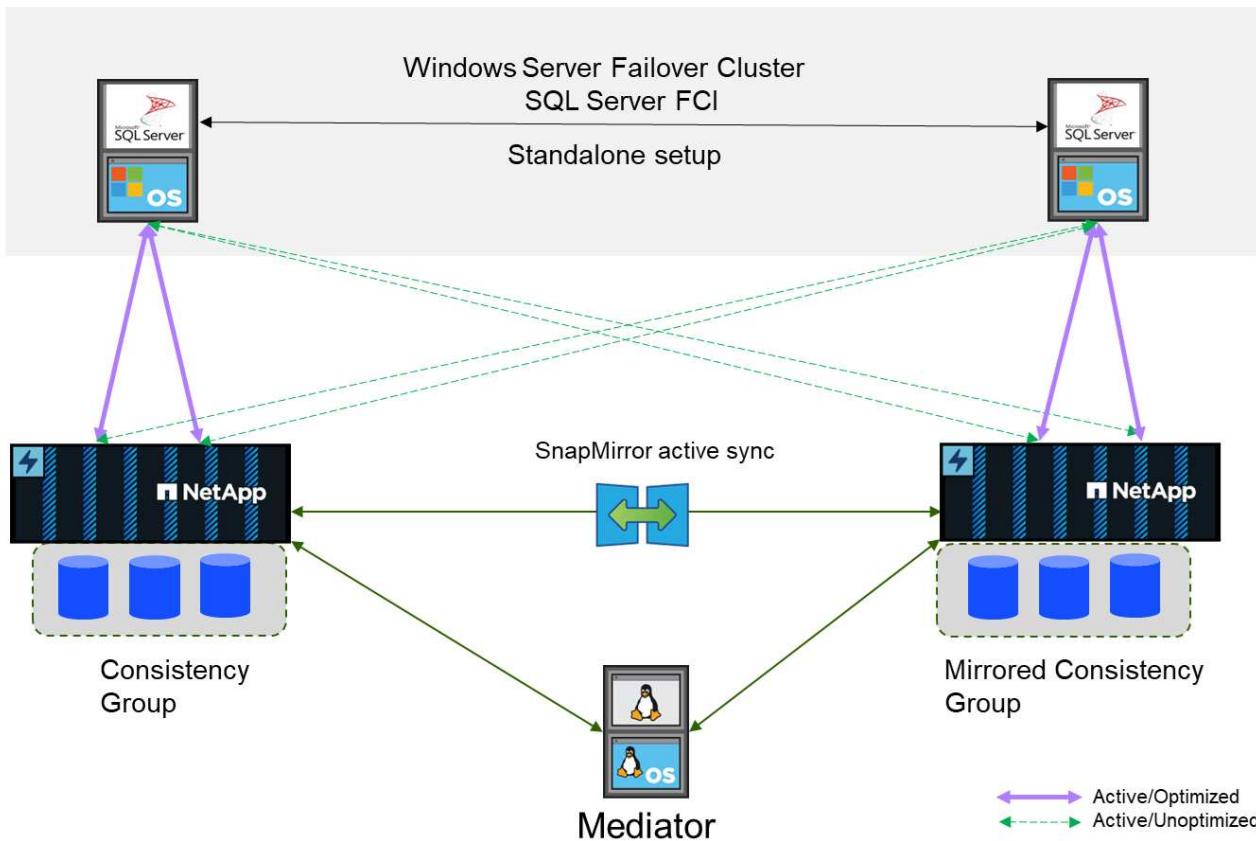
개요

SnapMirror Active Sync를 사용하면 개별 SQL Server 데이터베이스 및 애플리케이션이 스토리지 및 네트워크가 중단되어도 작업을 계속할 수 있으며, 수동 작업 없이 투명한 스토리지 페일오버를 수행할 수 있습니다.

ONTAP 9.15.1부터 SnapMirror 액티브 동기화는 기존 비대칭 구성 외에 대칭 액티브/액티브 아키텍처를 지원합니다. 대칭 액티브/액티브 기능은 비즈니스 연속성 및 재해 복구를 위해 동기식 양방향 복제를 제공합니다. 또한 여러 장애 도메인에 걸쳐 데이터에 대한 동시 읽기 및 쓰기 액세스를 통해 중요한 SAN 워크로드에 대한 데이터 액세스를 보호함으로써 무중단 운영을 보장하고 재해 또는 시스템 장애 시 다운타임을 최소화할 수 있습니다.

SQL Server 호스트는 FC(Fibre Channel) 또는 iSCSI LUN을 사용하여 스토리지에 액세스합니다. 복제된 데이터의 복제본을 호스팅하는 각 클러스터 간의 복제입니다. 이 기능은 스토리지 수준 복제이므로 독립 실행형 호스트 또는 장애 조치 클러스터 인스턴스에서 실행되는 SQL Server 인스턴스는 클러스터 중 하나에서 읽기/쓰기 작업을 수행할 수 있습니다. 계획 및 구성 단계는 ["SnapMirror 활성 동기화에 대한 ONTAP 문서"](#) 참조하십시오.

대칭 액티브/액티브를 사용한 **SnapMirror** 액티브 싱크 아키텍처



- 동기 복제\*\*

정상 작동 시 각 복제본은 항상 RPO=0 동기식 복제본이며 한 가지 예외가 있습니다. 데이터를 복제할 수 없는 경우 ONTAP는 데이터를 복제해야 하는 요구 사항을 해제하고 다른 사이트의 LUN이 오프라인 상태가 되는 동안 한 사이트에서 입출력 서비스를 재개합니다.

- 스토리지 하드웨어\*\*

다른 스토리지 재해 복구 솔루션과 달리 SnapMirror Active Sync는 비대칭적 플랫폼 유연성을 제공합니다. 각 사이트의 하드웨어는 동일할 필요가 없습니다. 이 기능을 사용하면 SnapMirror 액티브 동기화를 지원하는 데 사용되는 하드웨어를 적절한 크기로 조정할 수 있습니다. 전체 운영 워크로드를 지원해야 하는 경우 원격 스토리지 시스템이 기본 사이트와 동일할 수 있지만 재해로 인해 I/O가 감소할 경우 원격 사이트의 소규모 시스템보다 비용 효율적입니다.

- ONTAP 중재자\*\*

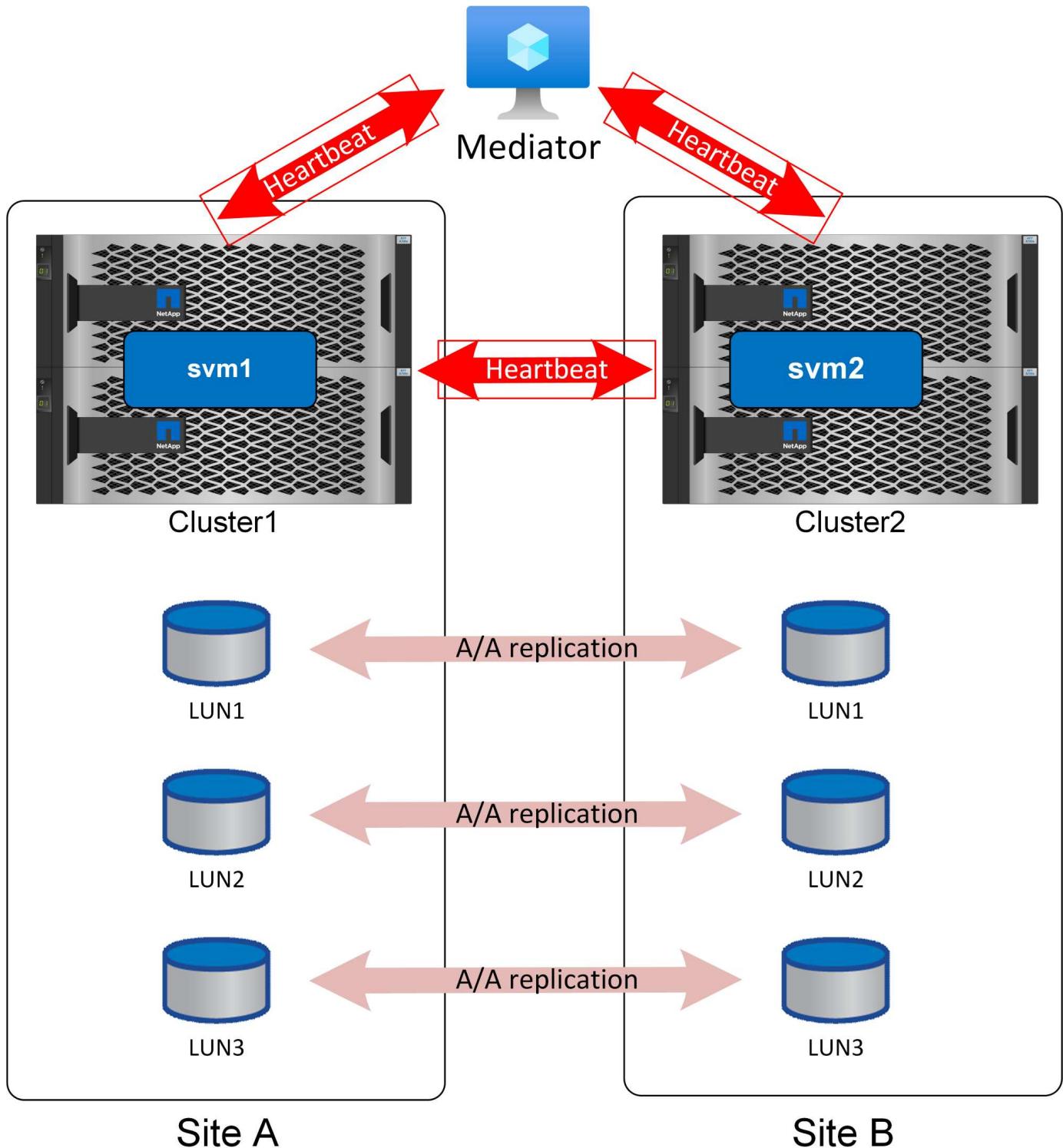
ONTAP 중재자는 NetApp 지원에서 다운로드되는 소프트웨어 응용 프로그램이며 일반적으로 작은 가상 컴퓨터에 구축됩니다. ONTAP 중재자는 타이브레이커가 아닙니다. SnapMirror 활성 동기화 복제에 참여하는 두 클러스터에 대한 대체 통신 채널입니다. 자동화된 운영은 직접 연결과 중재자를 통해 파트너로부터 받은 응답에 따라 ONTAP에 의해 주도됩니다.

#### ONTAP 중재자

장애 조치를 안전하게 자동화하려면 중재자가 필요합니다. 이상적으로는 독립적인 3차 사이트에 배치되지만 복제에 참여하는 클러스터 중 하나와 같은 위치에 배치하면 대부분의 요구 사항에 대해 여전히 기능을 수행할 수 있습니다.

중재자는 실제로 동점을 깨는 역할을 하지 않지만, 실질적으로는 그런 기능을 제공합니다. 중재자는 클러스터 노드의

상태를 결정하는 데 도움을 주고 사이트 장애 발생 시 자동 전환 프로세스를 지원합니다. 중재자는 어떠한 상황에서도 데이터를 전송하지 않습니다.



자동 장애 조치의 가장 큰 과제는 브레이크 분할 문제이며, 두 사이트가 서로 연결이 끊어지면 이 문제가 발생합니다. 어떻게 해야 합니까? 서로 다른 두 사이트가 자신을 데이터의 정상적인 복제본으로 지정하도록 하고 싶지 않지만, 단일 사이트가 상대 사이트의 실제 손실과 반대쪽 사이트와 통신할 수 없는 간의 차이를 어떻게 알 수 있습니까?

중재자가 사진을 입력하는 위치입니다. 세 번째 사이트에 배치되고 각 사이트에 해당 사이트에 대한 별도의 네트워크 연결이 있는 경우 각 사이트에 대한 추가 경로를 통해 다른 사이트의 상태를 확인할 수 있습니다. 위의 그림을 다시 보고

다음 시나리오를 고려하십시오.

- 중재자가 한 사이트 또는 두 사이트에서 작동하지 않거나 연결할 수 없는 경우 어떻게 됩니까?
  - 두 클러스터는 복제 서비스에 사용되는 동일한 링크를 통해 서로 계속 통신할 수 있습니다.
  - RPO=0 보호로 데이터를 계속 제공합니다
- 사이트 A에 장애가 발생하면 어떻게 됩니까?
  - 사이트 B는 두 통신 채널이 모두 다운되는 것을 볼 수 있습니다.
  - 사이트 B가 데이터 서비스를 인수하지만 RPO = 0 미러링이 없습니다
- 사이트 B에 장애가 발생하면 어떻게 됩니까?
  - 사이트 A는 두 통신 채널이 모두 다운되는 것을 볼 수 있습니다.
  - 사이트 A가 데이터 서비스를 인수하지만 RPO = 0 미러링이 없음

고려해야 할 다른 시나리오가 있습니다: 데이터 복제 링크의 손실. 사이트 간 복제 링크가 손실되면 RPO=0 미러링이 불가능할 것입니다. 그러면 어떻게 됩니까?

이는 선호 사이트 상태에 의해 제어됩니다. SM-AS 관계에서 사이트 중 하나가 다른 사이트에 대한 보조 사이트입니다. 이 작업은 일반 작업에는 영향을 주지 않으며 모든 데이터 액세스는 대칭이지만 복제가 중단되면 작업을 재개하려면 연결을 끊어야 합니다. 그 결과 기본 사이트가 미러링 없이 운영을 계속하고 복제 통신이 복구될 때까지 보조 사이트가 입출력 처리를 중지합니다.

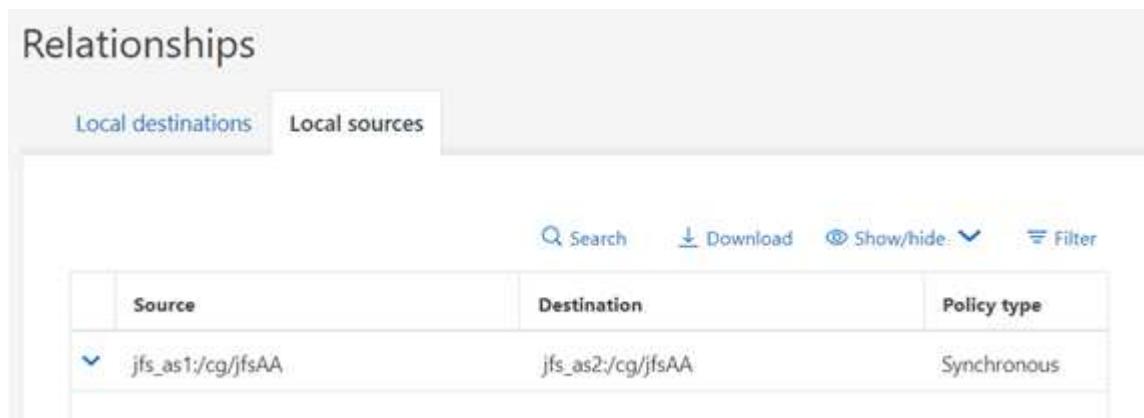
기본 사이트

SnapMirror 액티브 동기화 동작은 대칭이며 한 가지 중요한 예외 기본 사이트 구성이 있습니다.

SnapMirror 활성 동기화는 한 사이트를 "소스"로 간주하고 다른 사이트는 "대상"으로 간주합니다. 이는 단방향 복제 관계를 의미하지만 입출력 동작에는 적용되지 않습니다. 복제는 양방향이고 대칭이며 입출력 응답 시간은 미러의 양쪽에서 동일합니다.

`source` 지정은 기본 사이트를 제어합니다. 복제 링크가 손실되면 소스 복제본의 LUN 경로는 계속 데이터를 제공하고 대상 복제본의 LUN 경로는 복제가 다시 설정되고 SnapMirror가 동기식 상태로 다시 전환될 때까지 사용할 수 없게 됩니다. 그러면 경로가 데이터 제공을 재개합니다.

소스/대상 구성은 SystemManager를 통해 볼 수 있습니다.



The screenshot shows the 'Relationships' section of the SystemManager interface. The 'Local destinations' tab is selected. A table displays a single relationship entry:

Source	Destination	Policy type
jfs_as1:/cg/jfsAA	jfs_as2:/cg/jfsAA	Synchronous

At the top of the interface, there are search, download, and filter buttons.

또는 CLI에서:

```
Cluster2::> snapmirror show -destination-path jfs_as2:/cg/jfsAA

        Source Path: jfs_as1:/cg/jfsAA
        Destination Path: jfs_as2:/cg/jfsAA
        Relationship Type: XDP
        Relationship Group Type: consistencygroup
        SnapMirror Schedule: -
        SnapMirror Policy Type: automated-failover-duplex
        SnapMirror Policy: AutomatedFailOverDuplex
        Tries Limit: -
        Throttle (KB/sec): -
        Mirror State: Snapmirrored
        Relationship Status: InSync
```

핵심은 소스는 클러스터 1의 SVM입니다. 위에서 언급한 바와 같이 "원본" 및 "대상"이라는 용어는 복제된 데이터의 흐름을 설명하지 않습니다. 두 사이트 모두 쓰기를 처리하여 반대쪽 사이트로 복제할 수 있습니다. 실제로 두 클러스터 모두 소스와 타겟입니다. 하나의 클러스터를 소스로 지정하면 복제 링크가 손실된 경우 읽기-쓰기 스토리지 시스템으로 존속하는 클러스터를 제어할 수 있습니다.

네트워크 토플로지

균일한 액세스

균일한 액세스 네트워킹은 호스트가 두 사이트(또는 동일한 사이트 내의 장애 도메인)의 경로를 액세스할 수 있음을 의미합니다.

SM-AS의 중요한 기능은 호스트의 위치를 알 수 있도록 스토리지 시스템을 구성하는 기능입니다. 특정 호스트에 LUN을 매핑할 때 해당 LUN이 지정된 스토리지 시스템에 근접한지 여부를 지정할 수 있습니다.

근접 설정

근접성은 특정 호스트 WWN 또는 iSCSI 이니시에이터 ID가 로컬 호스트에 속함을 나타내는 클러스터별 구성입니다. LUN 액세스를 구성하는 두 번째 선택적 단계입니다.

첫 번째 단계는 일반적인 igrup 구성입니다. 각 LUN은 해당 LUN에 액세스해야 하는 호스트의 WWN/iSCSI ID가 포함된 igrup에 매핑되어야 합니다. LUN에 대한 액세스 권한이 있는 호스트를 제어합니다.

두 번째 단계는 호스트 근접성을 구성하는 것입니다. 액세스를 제어하지 않고 \_priority\_를 제어합니다.

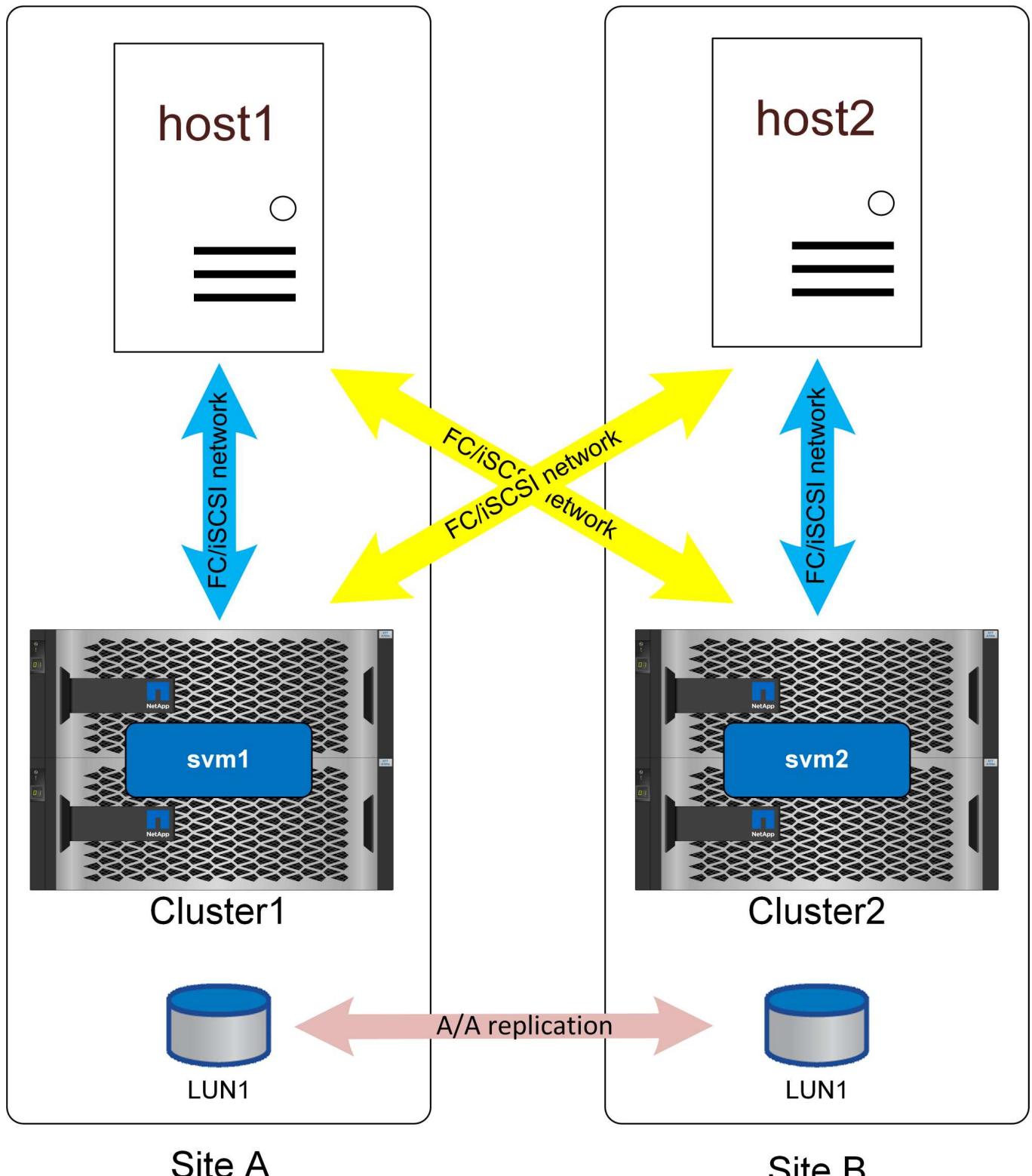
예를 들어 사이트 A의 호스트가 SnapMirror 활성 동기화로 보호되는 LUN을 액세스하도록 구성할 수 있고 SAN이 사이트 간에 확장되므로 사이트 A의 스토리지 또는 사이트 B의 스토리지를 사용하여 해당 LUN에 대한 경로를 사용할 수 있습니다

근접 설정이 없으면 두 스토리지 시스템 모두 활성/최적화된 경로를 홍보하기 때문에 해당 호스트는 두 스토리지 시스템을 동일하게 사용합니다. 사이트 간 SAN 대기 시간 및/또는 대역폭이 제한된 경우 이를 원하지 않을 수 있으며, 정상적인 작업 중에 각 호스트가 로컬 스토리지 시스템에 대한 경로를 우선적으로 사용하도록 할 수 있습니다. 호스트 WWN/iSCSI ID를 로컬 클러스터에 근위부 호스트로 추가하여 구성합니다. 이 작업은 CLI 또는 SystemManager에서

수행할 수 있습니다.

## **AFF**

AFF 시스템의 경우 호스트 근접성이 구성된 경우 경로가 아래와 같이 표시됩니다.



Active/Optimized Path

Active Path

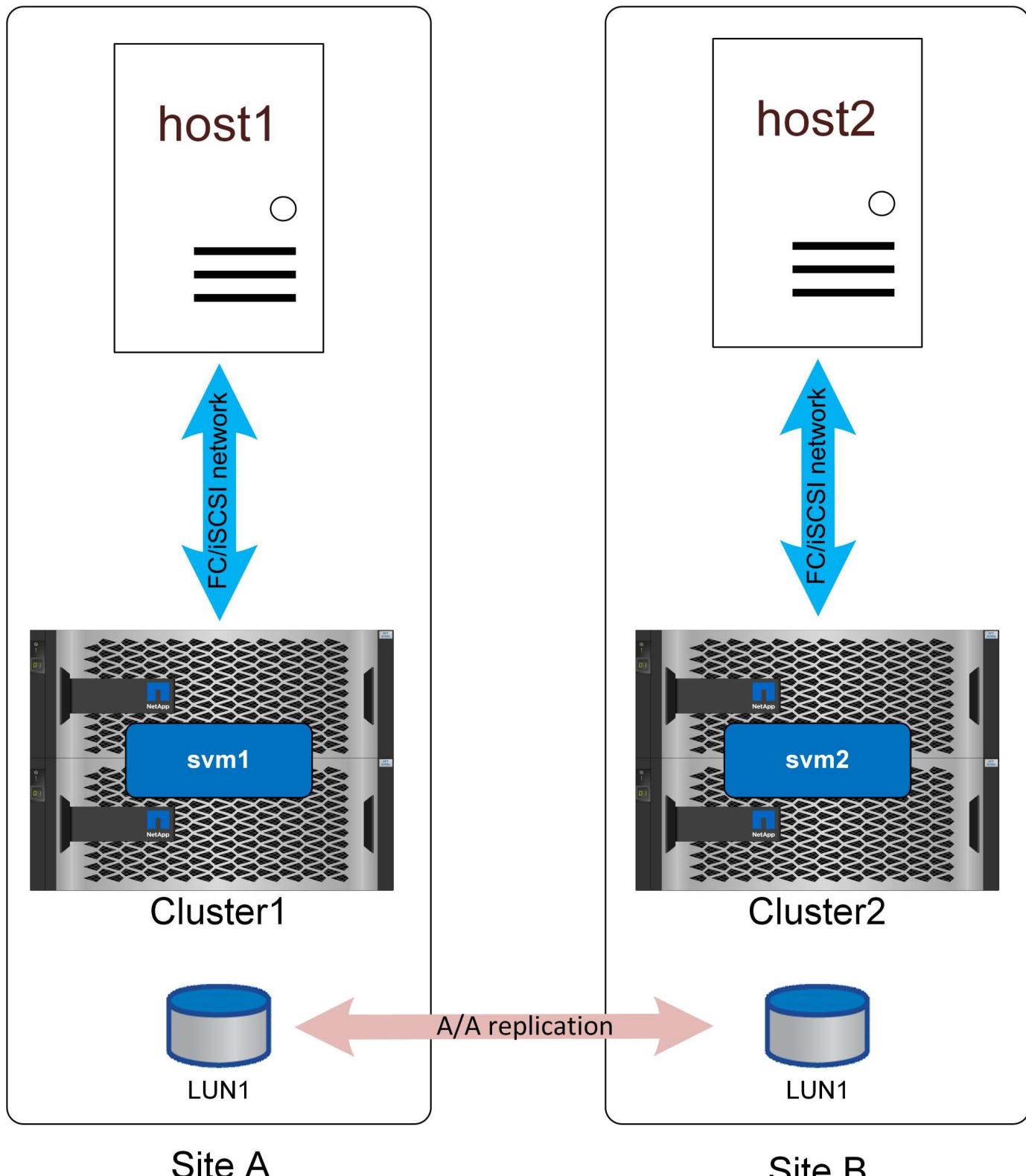
정상 작동 시 모든 입출력은 로컬 입출력입니다. 읽기와 쓰기는 로컬 스토리지 시스템에서 서비스됩니다. 물론 쓰기 입출력도 인식되기 전에 로컬 컨트롤러에 의해 원격 시스템으로 복제되어야 하지만 모든 읽기 입출력은 로컬로 처리되며 사이트 간 SAN 링크를 통하여 추가적인 지연 시간이 발생하지 않습니다.

최적화되지 않은 경로는 액티브/최적화 경로가 모두 손실되는 경우에만 사용됩니다. 예를 들어, 사이트 A의 전체 스토리지에 전원이 공급되지 않으면 사이트 A의 호스트가 사이트 B의 어레이에 대한 경로에 액세스할 수 있으므로 지연 시간이 더 길더라도 작동 상태를 유지할 수 있습니다.

로컬 클러스터를 통한 중복 경로가 있습니다. 이러한 경로는 단순성을 위해 이러한 다이어그램에 표시되지 않습니다. ONTAP 스토리지 시스템은 HA 자체이므로 컨트롤러 장애가 사이트 장애로 이어질 수 없습니다. 영향을 받는 사이트에서 사용되는 로컬 경로가 변경될 뿐입니다.

#### 비균일 액세스

비균일 액세스 네트워킹은 각 호스트가 로컬 스토리지 시스템의 포트에만 액세스할 수 있음을 의미합니다. SAN은 사이트(또는 동일한 사이트 내의 장애 도메인)에 걸쳐 확장되지 않습니다.



## Active/Optimized Path

이 접근 방식의 주요 이점은 SAN의 단순성입니다. 네트워크를 통해 SAN을 확장할 필요가 없어졌습니다. 일부 고객은 사이트 간에 지연 시간이 충분히 짧거나 사이트 간 네트워크를 통해 FC SAN 트래픽을 터널링할 수 있는 인프라가

부족합니다.

비균일 액세스의 단점은 복제 링크 손실을 비롯한 특정 장애 시나리오로 인해 일부 호스트가 스토리지에 액세스할 수 없게 된다는 것입니다. 클러스터 이외의 데이터베이스와 같이 단일 인스턴스로 실행되는 애플리케이션은 로컬 스토리지 연결이 끊긴 경우 해당 마운트의 단일 호스트에서만 실행 중이므로 실패합니다. 데이터는 여전히 보호되지만 데이터베이스 서버는 더 이상 액세스할 수 없습니다. 원격 사이트에서 자동 프로세스를 통해 재시작해야 합니다. 예를 들어 VMware HA는 한 서버에서 모든 경로 다운 상황을 감지하고 경로를 사용할 수 있는 다른 서버에서 VM을 다시 시작할 수 있습니다.

반면, Oracle RAC와 같은 클러스터된 애플리케이션은 두 개의 사이트에서 동시에 사용할 수 있는 서비스를 제공할 수 있습니다. 사이트를 잃었다고 해서 애플리케이션 서비스 전체가 손실되는 것은 아닙니다. 인스턴스는 계속 사용할 수 있으며 정상적인 사이트에서 실행됩니다.

대부분의 경우, 사이트 간 링크를 통해 스토리지에 액세스하는 애플리케이션의 자연 시간 오버헤드가 허용할 수 없는 경우가 많습니다. 즉, 사이트의 스토리지 손실로 인해 장애가 발생한 사이트의 서비스를 종료해야 하기 때문에 일관된 네트워킹의 가용성 향상은 최소화됩니다.



로컬 클러스터를 통한 중복 경로가 있습니다. 이러한 경로는 단순성을 위해 이러한 다이어그램에 표시되지 않습니다. ONTAP 스토리지 시스템은 HA 자체이므로 컨트롤러 장애가 사이트 장애로 이어질 수 없습니다. 영향을 받는 사이트에서 사용되는 로컬 경로가 변경될 뿐입니다.

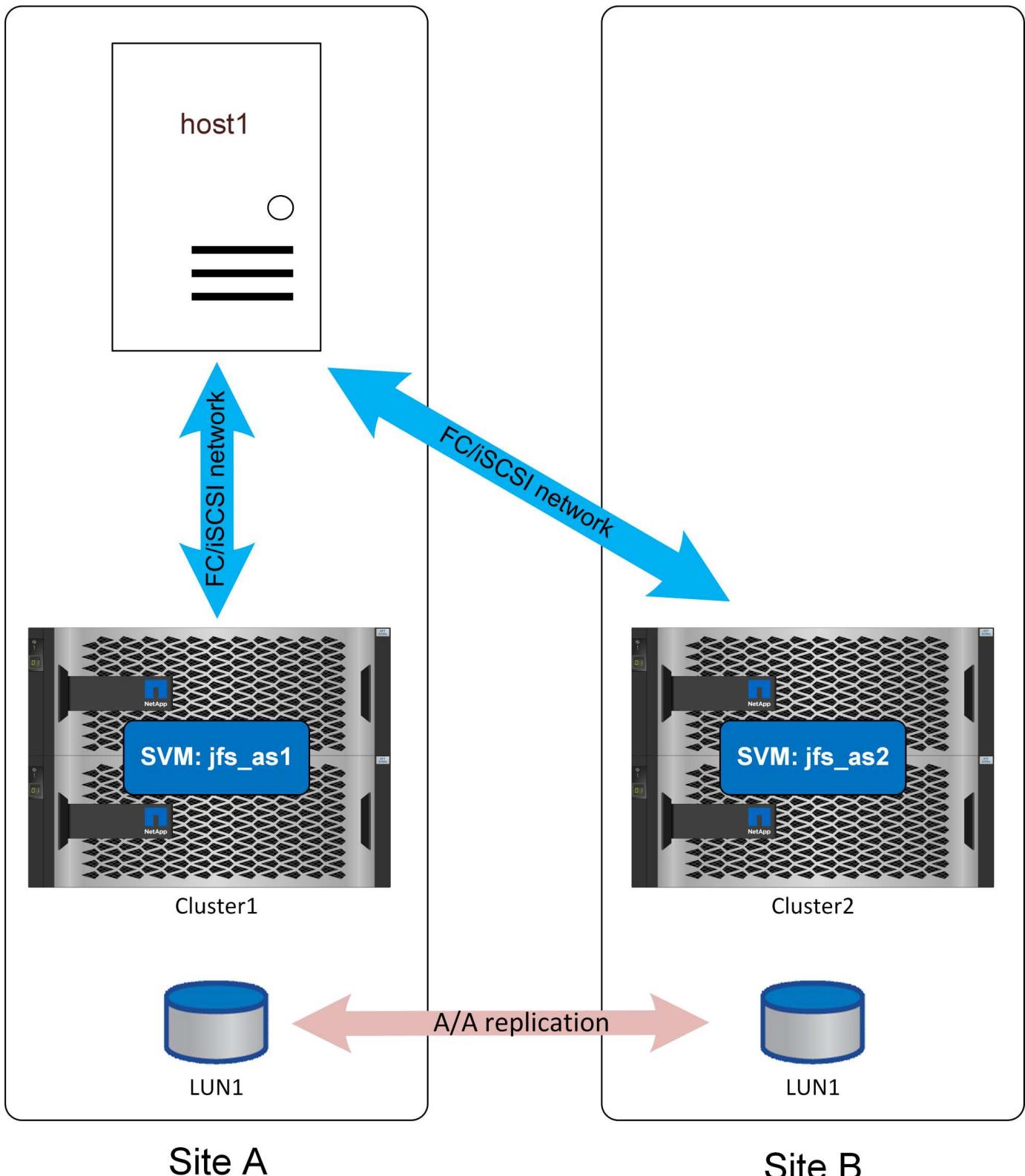
개요

여러 가지 방법으로 SnapMirror 활성 동기화에서 작동하도록 SQL Server를 구성할 수 있습니다. 정답은 사용 가능한 네트워크 연결, RPO 요구사항 및 가용성 요구사항에 따라 다릅니다.

**SQL Server**의 독립 실행형 인스턴스입니다

파일 레이아웃 및 서버 구성에 대한 Best Practice는 설명서에서 권장하는 방법과 동일합니다 "[ONTAP 기반 SQL Server](#)".

독립 실행형 설정을 사용하면 한 사이트에서만 SQL Server를 실행할 수 있습니다. 아마도 "[균일](#)" access가 사용될 것입니다.



균일한 액세스를 사용하더라도 두 사이트의 스토리지 장애로 인해 데이터베이스 작업이 중단되지 않습니다. 데이터베이스 서버가 포함된 사이트의 전체 사이트 장애는 물론 중단이 발생할 수 있습니다.

일부 고객은 사전 구성된 SQL Server 설정으로 원격 사이트에서 실행 중인 OS를 구성할 수 있으며, 운영 인스턴스의 빌드 버전과 동일한 버전으로 업데이트할 수 있습니다. 장애 조치에는 대체 사이트에서 해당 독립 실행형 SQL Server 인스턴스를 활성화하고, LUN을 검색하고, 데이터베이스를 시작해야 합니다. 스토리지 측에서 별도의 작업이 필요하지

않으므로 Windows Powershell cmdlet을 사용하여 전체 프로세스를 자동화할 수 있습니다.

"비균일" 액세스를 사용할 수도 있지만 데이터베이스에 스토리지에 대한 가용 경로가 없기 때문에 데이터베이스 서버가 있는 스토리지 시스템에 장애가 발생하면 데이터베이스가 중단될 수 있습니다. 일부 경우에는 이 방법이 여전히 허용됩니다. SnapMirror 활성 동기화는 여전히 RPO=0 데이터 보호를 제공하며, 사이트 장애 발생 시 나머지 복제본은 활성 상태이며 위에서 설명한 것과 동일한 액세스 절차를 사용하여 작업을 재개할 수 있습니다.

단순하고 자동화된 페일오버 프로세스는 가상화 호스트를 사용하여 더욱 쉽게 구성할 수 있습니다. 예를 들어, SQL Server 데이터 파일이 부팅 VMDK와 함께 보조 스토리지에 동기식으로 복제되면 재해 발생 시 전체 환경을 대체 사이트에서 활성화할 수 있습니다. 관리자는 정상적인 사이트에서 호스트를 수동으로 활성화하거나 VMware HA와 같은 서비스를 통해 프로세스를 자동화할 수 있습니다.

### **SQL Server** 장애 조치 클러스터 인스턴스입니다

SQL Server 장애 조치 인스턴스는 물리적 서버 또는 가상 서버에서 게스트 운영 체제로 실행되는 Windows 장애 조치 클러스터에서 호스팅될 수도 있습니다. 이 다중 호스트 아키텍처는 SQL Server 인스턴스와 스토리지 복구 기능을 제공합니다. 이러한 배포는 향상된 성능을 유지하면서 강력한 장애 조치 프로세스를 원하는 수요가 많은 환경에 유용합니다. 장애 조치 클러스터 설정에서 호스트 또는 운영 스토리지가 영향을 받으면 SQL 서비스가 보조 호스트로 페일오버되고, 동시에 보조 스토리지를 사용하여 IO를 제공할 수 있습니다. 자동화 스크립트나 관리자 개입이 필요하지 않습니다.

실패 시나리오

전체 SnapMirror 액티브 동기화 애플리케이션 아키텍처를 계획하려면 계획된 페일오버 및 예상치 못한 다양한 페일오버 시나리오에서 SM-AS가 어떻게 반응하는지 이해해야 합니다.

다음 예에서는 사이트 A가 기본 사이트로 구성되어 있다고 가정합니다.

복제 접속이 끊어졌습니다

SM-AS 복제가 중단되면 클러스터에서 변경 내용을 반대편 사이트로 복제할 수 없기 때문에 쓰기 입출력을 완료할 수 없습니다.

#### **사이트 A(기본 사이트)**

기본 사이트에서 복제 링크 실패의 결과는ONTAP가 복제된 쓰기 작업을 다시 시도하기 때문에 쓰기 입출력 처리가 약 15초 동안 일시 중지되는 것입니다. 이 경우 복제 링크에 도달할 수 없는 것으로 판단됩니다. 15초가 지나면 사이트 A 시스템이 읽기 및 쓰기 IO 처리를 재개합니다. SAN 경로는 변경되지 않으며 LUN은 온라인 상태로 유지됩니다.

#### **사이트 B**

사이트 B는 SnapMirror 활성 동기화 기본 사이트가 아니므로 약 15초 후에 해당 LUN 경로를 사용할 수 없게 됩니다.

스토리지 시스템 장애

스토리지 시스템 장애의 결과는 복제 링크 손실의 결과와 거의 동일합니다. 정상적인 사이트에서 약 15초의 입출력 일시 중지 시간이 발생합니다. 15초가 지나면 평소와 같이 해당 사이트에서 입출력이 재개됩니다.

중재자의 상실

중재자 서비스는 스토리지 운영을 직접 제어하지 않습니다. 클러스터 간 대체 제어 경로 역할을 합니다. 이는 주로 브레인 분할 시나리오의 위험 없이 장애 조치를 자동화하는 데 있습니다. 정상 작동 시 각 클러스터가 파트너에 변경

사항을 복제하고 있으므로 각 클러스터가 온라인 상태이고 데이터를 제공하고 있는지 확인할 수 있습니다. 복제 링크가 실패하면 복제가 중지됩니다.

안전한 자동 폐일오버를 위해 중재자가 필요한 이유는 스토리지 클러스터에서 양방향 통신 손실이 네트워크 중단이나 실제 스토리지 장애로 인한 것인지 여부를 확인할 수 없기 때문입니다.

중재자는 각 클러스터에서 파트너 상태를 확인할 수 있는 대체 경로를 제공합니다. 시나리오는 다음과 같습니다.

- 클러스터가 파트너에게 직접 연락할 수 있는 경우 복제 서비스가 작동합니다. 별도의 조치가 필요 없습니다.
- 기본 사이트가 파트너에게 직접 연락하거나 중재자를 통해 연락할 수 없는 경우, 해당 파트너가 실제로 사용할 수 없거나 격리되어 해당 LUN 경로를 오프라인으로 설정한 것으로 간주됩니다. 그러면 기본 사이트가 RPO=0 상태를 해제하고 읽기 및 쓰기 입출력을 계속 처리합니다.
- 비선호 사이트가 해당 파트너에 직접 연락할 수 없지만 중재자를 통해 연락할 수 있는 경우 해당 경로가 오프라인 상태가 되고 복제 연결이 반환될 때까지 기다립니다.
- 비선호 사이트가 파트너에게 직접 연락하거나 운영 중재자를 통해 연락할 수 없는 경우, 파트너는 실제로 파트너를 사용할 수 없거나 격리되어 LUN 경로를 오프라인으로 전환했다고 가정합니다. 그러면 비기본 사이트가 RPO=0 상태를 해제하고 읽기 및 쓰기 입출력을 계속 처리합니다. 복제 소스의 역할을 가정하고 새로운 기본 사이트가 됩니다.

중재자를 완전히 사용할 수 없는 경우:

- 기본 설정되지 않은 사이트 또는 스토리지 시스템의 장애를 포함하여 어떠한 이유로든 복제 서비스에 장애가 발생하면 기본 사이트에서 RPO=0 상태를 해제하고 읽기 및 쓰기 입출력 처리를 재개합니다. 기본 사이트가 아닌 사이트는 해당 경로를 오프라인으로 전환합니다.
- 기본 사이트에 장애가 발생하면 기본 사이트가 반대 사이트가 실제로 오프라인 상태인지 확인할 수 없으므로 기본 사이트가 서비스를 다시 시작하는 것이 안전하지 않으므로 운영 중단이 발생합니다.

서비스를 복원하는 중입니다

사이트 간 연결 복원 또는 장애 시스템의 전원 켜기와 같은 장애가 해결되면 SnapMirror 활성 동기화 엔드포인트는 장애가 있는 복제 관계의 존재를 자동으로 감지하여 RPO=0 상태로 되돌립니다. 동기식 복제가 다시 설정되면 장애가 발생한 경로가 다시 온라인 상태가 됩니다.

대부분의 경우, 클러스터된 애플리케이션은 장애가 발생한 경로의 반환을 자동으로 감지하여 다시 온라인 상태로 돌아갑니다. 호스트 레벨 SAN 검사가 필요하거나 애플리케이션을 수동으로 다시 온라인으로 전환해야 하는 경우도 있습니다. 애플리케이션 및 구성 방법에 따라 다르며 일반적으로 이러한 작업을 쉽게 자동화할 수 있습니다. ONTAP 자체는 자동 복구이므로 RPO=0 스토리지 작업을 재개하기 위해 사용자 개입이 필요하지 않습니다.

수동 폐일오버

기본 사이트를 변경하려면 간단한 작업이 필요합니다. 클러스터 간 복제 동작 전환에 대한 권한으로 입출력이 1-2초 동안 일시 중지되지만, 그렇지 않으면 입출력이 영향을 받지 않습니다.

## ASA R2 시스템의 스토리지 구성

개요

NetApp ASA R2는 미션 크리티컬 워크로드를 실행하는 SAN 전용 고객을 위한 간편하고 강력한 솔루션입니다. ONTAP 스토리지 솔루션과 Microsoft SQL Server를 실행하는 ASA R2

플랫폼을 결합하면 오늘날의 가장 까다로운 애플리케이션 요구 사항을 충족할 수 있는 엔터프라이즈급 데이터베이스 스토리지 설계를 구현할 수 있습니다.

다음 ASA 플랫폼은 모든 SAN 프로토콜(iSCSI, FC, NVMe/FC, NVMe/TCP)을 지원하는 ASA R2 시스템으로 분류됩니다. iSCSI, FC, NVMe/FC 및 NVMe/TCP 프로토콜은 다중 경로를 위해 대칭 액티브-액티브 아키텍처를 지원하므로 호스트와 스토리지 간의 모든 경로가 액티브/최적화됩니다.

- ASA A1K
- ASA A90
- ASA A70
- ASA A50
- ASA A30
- ASA A20

자세한 내용은을 참조하십시오 ["NetApp ASA"](#)

SQL Server on ONTAP 솔루션을 최적화하려면 SQL Server I/O 패턴과 특성을 이해해야 합니다. SQL Server 데이터베이스용 스토리지 레이아웃은 SQL Server의 성능 요구사항을 지원하는 동시에 인프라 전체를 관리할 수 있는 기능을 극대화해야 합니다. 또한 우수한 스토리지 레이아웃을 통해 초기 구축을 성공적으로 수행할 수 있으며 비즈니스 성장에 따라 환경이 원활하게 확장될 수 있습니다.

#### 데이터 스토리지 설계

데이터 및 로그 파일을 별도의 드라이브에 저장하는 것이 좋습니다. 데이터를 동시에 업데이트하고 요청하는 응용 프로그램의 경우 로그 파일은 쓰기 작업이 많고 데이터 파일(응용 프로그램에 따라 다름)은 읽기/쓰기 작업이 많이 사용됩니다. 데이터 검색을 위해 로그 파일이 필요하지 않습니다. 따라서 자체 드라이브에 있는 데이터 파일에서 데이터 요청을 처리할 수 있습니다.

새 데이터베이스를 만들 때는 데이터와 로그에 대해 별도의 드라이브를 지정하는 것이 좋습니다. 데이터베이스를 만든 후 파일을 이동하려면 데이터베이스를 오프라인으로 전환해야 합니다. Microsoft 권장 사항에 대한 자세한 내용은을 참조하십시오 ["데이터 및 로그 파일을 별도의 드라이브에 저장합니다"](#).

#### 스토리지 유닛 고려 사항

ASA의 스토리지 유닛은 SCSI/FC 호스트의 LUN이나 NVMe 호스트의 NVMe 네임스페이스를 참조합니다. 지원되는 프로토콜에 따라 LUN, NVMe 네임스페이스 또는 둘 모두를 생성하라는 메시지가 표시됩니다. 데이터베이스 구축을 위한 스토리지 유닛을 생성하기 전에 SQL Server 입출력 패턴과 특성이 워크로드와 백업 및 복구 요구 사항에 따라 어떻게 다른지 이해해야 합니다. 스토리지 유닛에 대한 다음 NetApp 권장 사항을 참조하십시오.

- 복잡한 관리를 방지하려면 동일한 호스트에서 실행되는 여러 SQL Server 간에 동일한 스토리지 유닛을 공유하지 마십시오. 동일한 호스트에서 여러 SQL Server 인스턴스를 실행하는 경우, 노드의 스토리지 유닛 제한에 근접하지 않은 경우 공유를 피하고, 데이터 관리가 용이하도록 호스트당 개별 스토리지 유닛을 인스턴스당 별도로 보유하는 것이 좋습니다.
- Windows의 26개 드라이브 문자 제한을 능가하려면 드라이브 문자 대신 NTFS 마운트 지점을 사용하십시오.
- 스냅샷 스케줄 및 보존 정책을 해제합니다. 대신 SnapCenter를 사용하여 SQL Server 데이터 스토리지 유닛의 스냅샷 복사본을 조정하세요.
- SQL Server 시스템 데이터베이스를 전용 스토리지 유닛에 배치합니다.
- tempdb는 SQL Server가 임시 작업 공간으로 사용하는 시스템 데이터베이스로, 특히 I/O를 많이 사용하는 DBCC CHECKDB 작업에 사용됩니다. 따라서 이 데이터베이스를 전용 스토리지 유닛에 배치하십시오. 스토리지 유닛

수가 까다로워진 대규모 환경에서는 신중한 계획 후에 tempdb를 동일한 스토리지 유닛에 있는 시스템 데이터베이스와 통합할 수 있습니다. SQL Server를 다시 시작할 때마다 이 데이터베이스가 다시 생성되므로 tempdb에 대한 데이터 보호는 높은 우선 순위가 아닙니다.

- 사용자 데이터 파일은 (.mdf`랜덤 읽기/쓰기 워크로드이므로 별도의 스토리지 유닛에 배치합니다. 일반적으로 트랜잭션 로그 백업은 데이터베이스 백업보다 더 자주 생성됩니다. 따라서 트랜잭션 로그 파일을 (.ldf 데이터 파일에서 별도의 스토리지 유닛 또는 VMDK에 배치하여 각각에 대해 독립적인 백업 일정을 생성할 수 있습니다. 또한 이 분리 방식은 로그 파일의 순차적 쓰기 I/O를 데이터 파일의 랜덤 읽기/쓰기 I/O에서 격리하고 SQL Server 성능을 크게 향상시킵니다.
- 볼트 정책이 SnapMirror 기능과 함께 사용될 때 보존 정책이 스냅샷을 덮어쓰지 않도록 하려면 로그 백업을 저장할 사용자 데이터베이스 파일과 로그 디렉토리가 별도의 저장소 유닛에 있어야 합니다.
- 전체 텍스트 검색 관련 파일과 같은 데이터베이스 파일과 데이터베이스가 아닌 파일을 동일한 저장소 유닛에 혼합하지 마십시오.
- 데이터베이스 보조 파일(파일 그룹의 일부로)을 별도의 스토리지 유닛에 배치하면 SQL Server 데이터베이스의 성능이 향상됩니다. 이 분리는 데이터베이스의 파일이 다른 파일과 해당 스토리지 유닛을 공유하지 않는 .mdf 경우에만 .mdf 유효합니다.
- Windows 서버에서 디스크 관리자를 사용하여 디스크를 포맷하는 동안 할당 단위 크기가 파티션에 대해 64K로 설정되어 있는지 확인하십시오.
- 사용자 데이터베이스 또는 시스템 데이터베이스를 마운트 지점을 호스팅하는 스토리지 유닛에 배치하지 마십시오.
- 를 참조하십시오 ["최신 SAN에 대한 ONTAP 모범 사례 하의 Microsoft Windows 및 네이티브 MPIO"](#) Windows에서 MPIO 속성의 iSCSI 장치에 다중 경로 지원을 적용하려면 다음을 수행합니다.
- Always On Failover 클러스터 인스턴스를 사용하는 경우 Windows Server Failover 클러스터 노드 전체에서 공유되는 스토리지 유닛에 사용자 데이터베이스를 배치해야 하며 SQL Server 인스턴스와 연결된 클러스터 그룹에 물리적 디스크 클러스터 리소스가 할당됩니다.

## 데이터베이스 파일 및 파일 그룹

초기 구축 단계에서는 ONTAP에 SQL Server 데이터베이스 파일을 적절하게 배치하는 것이 중요합니다. 따라서 비즈니스 요구 사항에 맞게 구성할 수 있는 최적의 성능, 공간 관리, 백업 및 복원 시간이 보장됩니다.

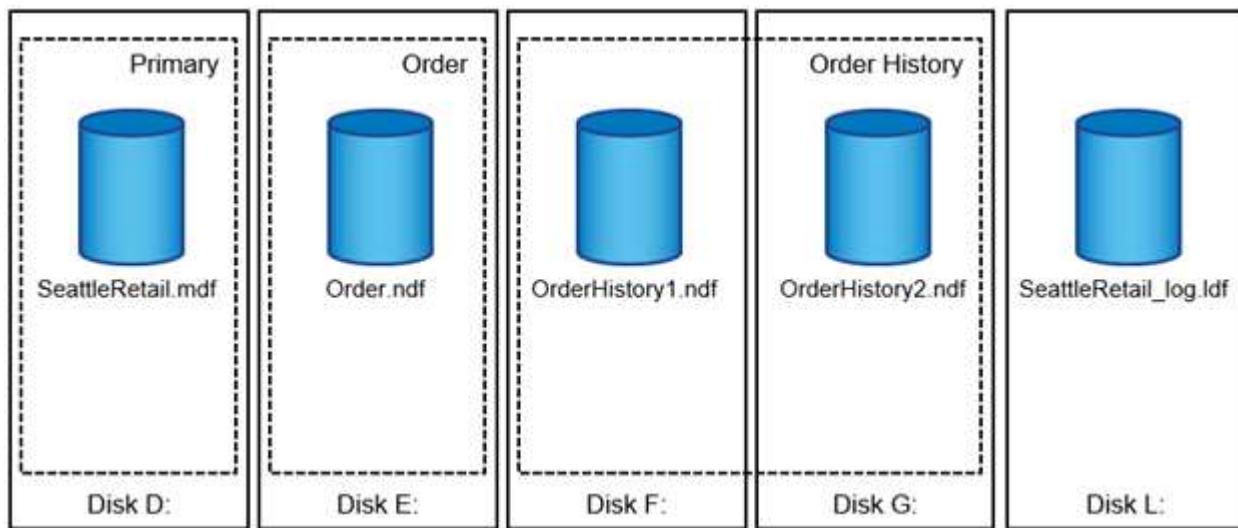
이론적으로 SQL Server(64비트)는 인스턴스당 32,767개의 데이터베이스와 524,272TB의 데이터베이스 크기를 지원하지만, 일반적인 설치에는 일반적으로 여러 개의 데이터베이스가 있습니다. 그러나 SQL Server에서 처리할 수 있는 데이터베이스 수는 로드 및 하드웨어에 따라 다릅니다. SQL Server 인스턴스가 수십, 수백 또는 수천 개의 소규모 데이터베이스를 호스팅하는 것은 드문 일이 아닙니다.

## 데이터베이스 파일 및 파일 그룹

각 데이터베이스는 하나 이상의 데이터 파일과 하나 이상의 트랜잭션 로그 파일로 구성됩니다. 트랜잭션 로그에는 데이터베이스 트랜잭션에 대한 정보와 각 세션에서 수행한 모든 데이터 수정에 대한 정보가 저장됩니다. 데이터가 수정될 때마다 SQL Server는 작업을 실행 취소(롤백)하거나 다시 실행(재생)할 수 있는 충분한 정보를 트랜잭션 로그에 저장합니다. SQL Server 트랜잭션 로그는 데이터 무결성과 견고성에 대한 SQL Server의 평판에 필수적인 부분입니다. 트랜잭션 로그는 SQL Server의 원자성, 일관성, 격리 및 내구성(ACID) 기능에 매우 중요합니다. SQL Server는 데이터 페이지가 변경되는 즉시 트랜잭션 로그에 기록합니다. 모든 DML(Data Manipulation Language) 문(예: SELECT, INSERT, UPDATE 또는 DELETE)은 완전한 트랜잭션이며, 트랜잭션 로그에서는 전체 집합 기반 작업이 수행되도록 하여 트랜잭션의 원자성을 확인합니다.

각 데이터베이스에는 기본 데이터 파일이 하나 있으며 기본적으로 확장명은 .mdf입니다. 또한 각 데이터베이스에는 보조 데이터베이스 파일이 있을 수 있습니다. 이러한 파일의 확장명은 기본적으로 .ndf입니다.

모든 데이터베이스 파일은 파일 그룹으로 그룹화됩니다. 파일 그룹은 논리적 단위로, 데이터베이스 관리를 간소화합니다. 논리 객체 배치와 물리적 데이터베이스 파일 간의 구분이 가능합니다. 데이터베이스 개체 테이블을 만들 때 기본 데이터 파일 구성에 대해 걱정하지 않고 파일 그룹을 배치할 파일 그룹을 지정합니다.



파일 그룹 내에 여러 데이터 파일을 배치할 수 있으므로 여러 스토리지 디바이스에 로드를 분산시킬 수 있으므로 시스템의 입출력 성능을 향상시킬 수 있습니다. 반면 SQL Server는 트랜잭션 로그에 순차적으로 기록하므로 트랜잭션 로그는 여러 파일의 이점을 얻지 못합니다.

파일 그룹에서 논리적 객체 배치와 물리적 데이터베이스 파일 간의 구분을 통해 데이터베이스 파일 레이아웃을 세밀하게 조정하여 스토리지 서브시스템에서 최대한 활용할 수 있습니다. 제공 워크로드를 지원하는 데이터 파일의 수는 애플리케이션에 영향을 주지 않고 I/O 요구사항과 예상 용량을 지원하기 위해 다양할 수 있습니다. 데이터베이스 레이아웃의 이러한 변형은 데이터베이스 파일이 아닌 파일 그룹에 데이터베이스 개체를 배치하는 응용 프로그램 개발자에게 영향을 주지 않습니다.

 \* NetApp는 \* 시스템 객체를 제외한 모든 항목에 대해 기본 파일 그룹을 사용하지 않을 것을 권장합니다.

사용자 객체에 대해 별도의 파일 그룹 또는 파일 집합을 만들면 특히 대규모 데이터베이스의 경우 데이터베이스 관리 및 재해 복구가 간소화됩니다.

#### 데이터베이스 인스턴스 파일 초기화

데이터베이스를 만들거나 기존 데이터베이스에 새 파일을 추가할 때 초기 파일 크기 및 자동 증가 매개 변수를 지정할 수 있습니다. SQL Server는 데이터를 기록할 데이터 파일을 선택할 때 비례 채우기 알고리즘을 사용합니다. 파일에서 사용할 수 있는 여유 공간에 비례하여 데이터의 양을 기록합니다. 파일의 여유 공간이 많을수록 처리하는 쓰기 횟수가 많아집니다.

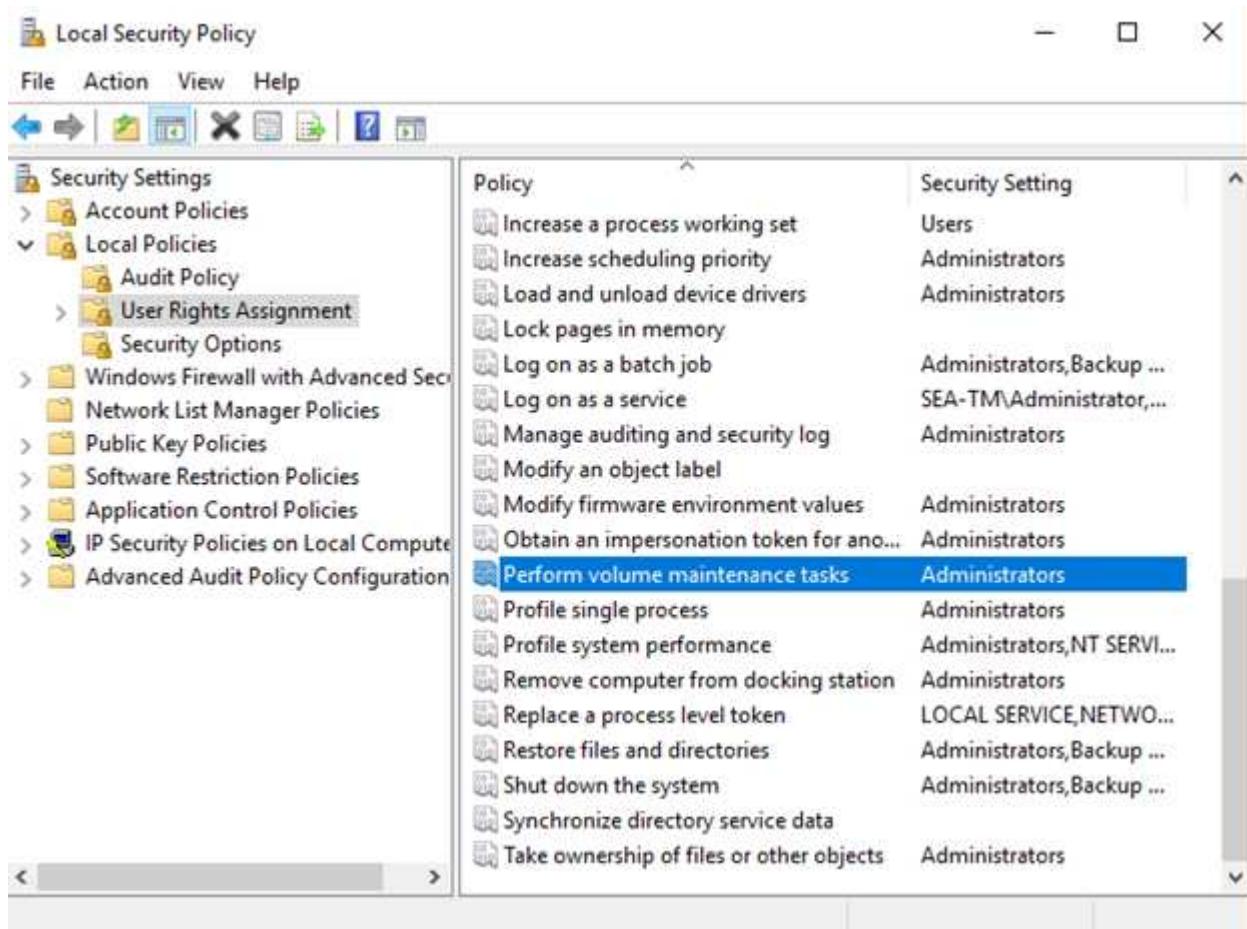
 \* NetApp는 단일 파일 그룹에 있는 모든 파일의 초기 크기 및 자동 증가 매개 변수가 같고 증가 크기가 백분율이 아닌 메가바이트로 정의됨을 \* 권장합니다. 이렇게 하면 비례 채우기 알고리즘이 데이터 파일 간에 쓰기 작업의 균형을 고르게 유지할 수 있습니다.

SQL Server는 파일을 늘릴 때마다 새로 할당된 공간을 0으로 채웁니다. 이 프로세스는 해당 파일에 기록해야 하는 모든 세션을 차단하거나 트랜잭션 로그가 증가하는 경우 트랜잭션 로그 레코드를 생성합니다.

SQL Server는 항상 트랜잭션 로그를 0으로 설정하며 이 동작은 변경할 수 없습니다. 그러나 인스턴트 파일 초기화를 사용하거나 사용하지 않도록 설정하여 데이터 파일의 제로화 여부를 제어할 수 있습니다. 즉각적인 파일 초기화를 사용하면 데이터 파일 증가 속도를 높이고 데이터베이스를 만들거나 복원하는 데 필요한 시간을 줄일 수 있습니다.

즉각적인 파일 초기화와 관련된 보안 위험이 있습니다. 이 옵션을 활성화하면 데이터 파일의 할당되지 않은 부분에 이전에 삭제된 OS 파일의 정보가 포함될 수 있습니다. 데이터베이스 관리자가 이러한 데이터를 검토할 수 있습니다.

SQL Server 시작 계정에 "볼륨 유지 관리 작업 수행"이라고도 하는 SA\_MANAGE\_VOLUME\_NAME 권한을 추가하여 즉각적인 파일 초기화를 활성화할 수 있습니다. 이 작업은 다음 그림과 같이 로컬 보안 정책 관리 응용 프로그램(secpol.msc)에서 수행할 수 있습니다. "볼륨 유지 관리 작업 수행" 권한에 대한 속성을 열고 SQL Server 시작 계정을 사용자 목록에 추가합니다.



사용 권한이 설정되어 있는지 확인하려면 다음 예제의 코드를 사용합니다. 이 코드는 SQL Server가 오류 로그에 추가 정보를 쓰고, 작은 데이터베이스를 만들고, 로그 내용을 읽도록 하는 두 개의 추적 플래그를 설정합니다.

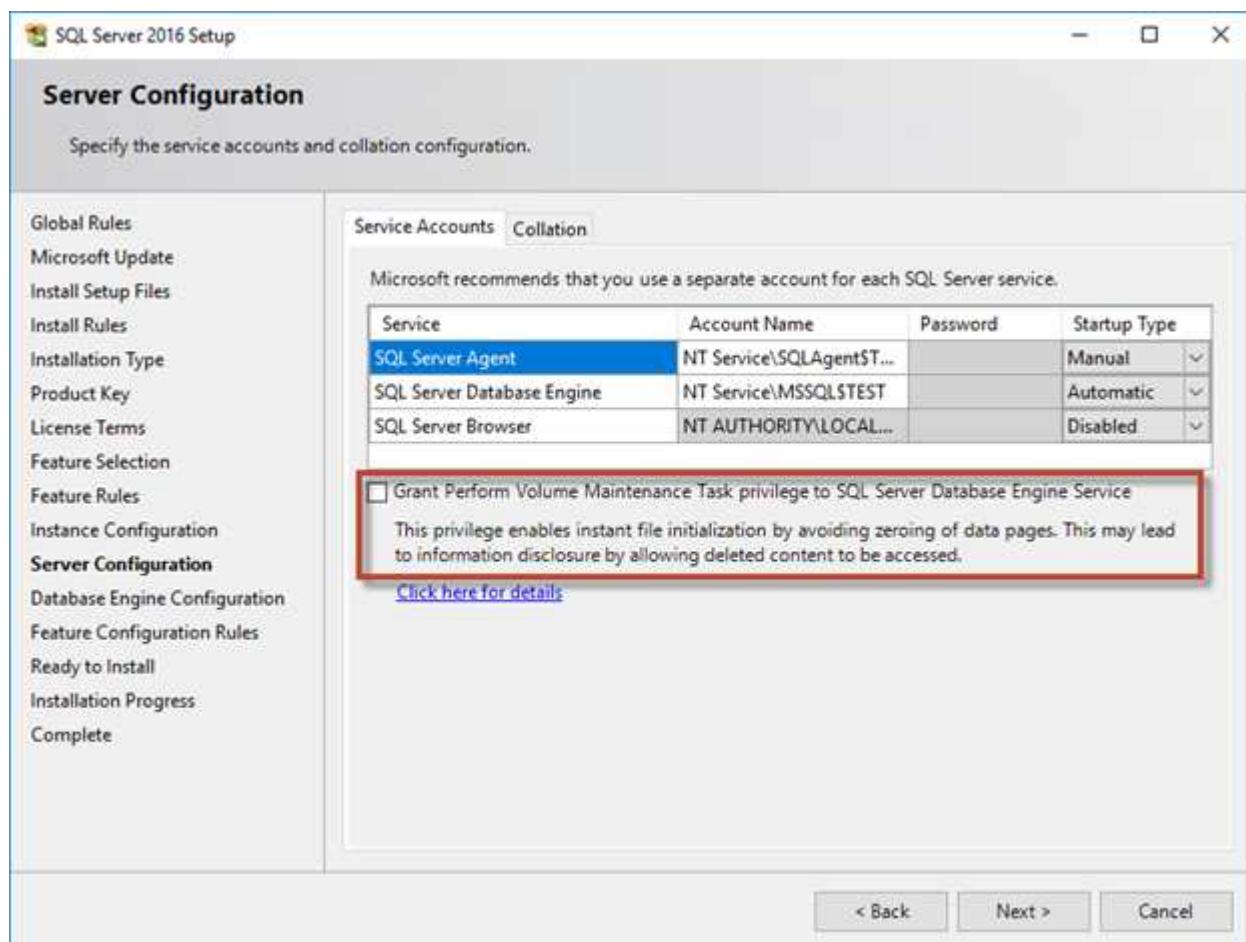
```
DBCC TRACEON (3004, 3605, -1)
GO
CREATE DATABASE DelMe
GO
EXECUTE sp_readerrorlog
GO
DROP DATABASE DelMe
GO
DBCC TRACEOFF (3004, 3605, -1)
GO
```

인스턴트 파일 초기화가 사용되지 않는 경우 SQL Server 오류 로그는 다음 예와 같이 SQL Server가 MDF 데이터

파일을 제로화하는 것 외에 LDF 로그 파일을 제로화하는 것을 보여 줍니다. 인스턴트 파일 초기화가 설정된 경우 로그 파일의 제로화만 표시됩니다.

	LogDate	ProcessInfo	Text
365	2017-02-09 08:10:07.660	spid53	Ckpt dbid 3 flush delta counts.
366	2017-02-09 08:10:07.660	spid53	Ckpt dbid 3 logging active xact info.
367	2017-02-09 08:10:07.750	spid53	Ckpt dbid 3 phase 1 ended (8)
368	2017-02-09 08:10:07.750	spid53	About to log Checkpoint end.
369	2017-02-09 08:10:07.880	spid53	Ckpt dbid 3 complete
370	2017-02-09 08:10:08.130	spid53	Starting up database 'DelMe'.
371	2017-02-09 08:10:08.150	spid53	Fixup Log tail(progress) zeroing C:\Program Files\Microsoft\SQL Server\M... 372 2017-02-09 08:10:08.160 spid53 Zeroing C:\Program Files\Microsoft SQL Server\MSSQL... 373 2017-02-09 08:10:08.170 spid53 Zeroing completed on C:\Program Files\Microsoft SQL... 374 2017-02-09 08:10:08.710 spid53 Ckpt dbid 6 started 375 2017-02-09 08:10:08.710 spid53 About to log Checkpoint begin.

볼륨 유지 관리 수행 작업은 SQL Server 2016에서 간소화되며 나중에 설치 프로세스 중에 옵션으로 제공됩니다. 다음 그림에서는 SQL Server 데이터베이스 엔진 서비스에 볼륨 유지 관리 작업을 수행할 수 있는 권한을 부여하는 옵션을 보여 줍니다.



데이터베이스 파일 크기를 제어하는 또 다른 중요한 데이터베이스 옵션은 자동 축소입니다. 이 옵션을 사용하면 SQL Server에서 정기적으로 데이터베이스 파일을 축소하고 크기를 줄이며 운영 체제에 공간을 해제합니다. 이 작업은 리소스를 많이 사용하며 새 데이터가 시스템에 유입될 때 일정 시간이 지난 후에 데이터베이스 파일이 다시 증가하기 때문에 거의 유용하지 않습니다. 데이터베이스에서 자동 축소를 사용하지 않아야 합니다.

## 로그 디렉토리

로그 디렉토리는 트랜잭션 로그 백업 데이터를 호스트 레벨에서 저장하기 위해 SQL Server에 지정됩니다. SnapCenter를 사용하여 로그 파일을 백업하는 경우 SnapCenter에서 사용하는 각 SQL Server 호스트에 로그 백업을 수행하도록 구성된 호스트 로그 디렉토리가 있어야 합니다.

로그 디렉토리를 전용 스토리지 유닛에 배치합니다. 호스트 로그 디렉토리의 데이터 양은 백업 크기 및 백업 보존 일수에 따라 달라집니다. SnapCenter는 SQL Server 호스트당 하나의 호스트 로그 디렉토리만 허용합니다. 호스트 로그 디렉토리는 SnapCenter → 호스트 → 플러그인 구성에서 구성할 수 있습니다.

- NetApp는 호스트 로그 디렉토리에 대해 다음을 권장합니다 \*.
- 호스트 로그 디렉토리가 백업 스냅샷 데이터를 손상시킬 수 있는 다른 유형의 데이터와 공유되지 않도록 하십시오.
-  SnapCenter에서 트랜잭션 로그를 복제할 전용 스토리지 유닛에 호스트 로그 디렉토리를 생성합니다.
- Always On Failover 클러스터 인스턴스를 사용하는 경우 호스트 로그 디렉토리에 사용되는 스토리지 유닛은 SnapCenter에서 백업되는 SQL Server 인스턴스와 동일한 클러스터 그룹의 클러스터 디스크 리소스여야 합니다.

## 데이터 보호

데이터베이스 백업 전략은 이론적인 기능이 아닌 식별된 비즈니스 요구 사항을 기반으로 해야 합니다. ONTAP의 스냅샷 기술을 결합하고 Microsoft SQL Server API를 활용하면 사용자 데이터베이스 크기에 관계없이 애플리케이션 정합성이 보장되는 백업을 신속하게 수행할 수 있습니다. 고급 또는 스케일아웃 데이터 관리 요구사항을 충족하기 위해 NetApp에서는 SnapCenter를 제공합니다.

### SnapCenter

SnapCenter는 엔터프라이즈 애플리케이션을 위한 NetApp 데이터 보호 소프트웨어입니다. SQL Server용 SnapCenter 플러그인과 Microsoft Windows용 SnapCenter 플러그인으로 관리되는 OS 작업을 통해 SQL Server 데이터베이스를 빠르고 쉽게 보호할 수 있습니다.

SQL Server 인스턴스는 독립 실행형 설치, 장애 조치 클러스터 인스턴스이거나 항상 가용성 그룹일 수 있습니다. 그 결과, 단일 창에서 데이터베이스를 보호, 클론 복제, 운영 또는 2차 복사본으로부터 복원할 수 있습니다. SnapCenter는 온프레미스, 클라우드 및 하이브리드 구성 모두에서 SQL Server 데이터베이스를 관리할 수 있습니다. 개발 또는 보고 용도로 데이터베이스 복사본을 조직 또는 대체 호스트에서 몇 분 내에 생성할 수도 있습니다.

또한 SQL Server는 생성 시 스냅샷에 올바른 데이터가 존재하도록 OS와 스토리지 간의 조정이 필요합니다. 대부분의 경우 이 작업을 수행하는 유일한 안전한 방법은 SnapCenter 또는 T-SQL을 사용하는 것입니다. 이 추가 조정 없이 생성된 스냅샷은 안정적으로 복구할 수 없습니다.

SnapCenter용 SQL Server 플러그인에 대한 자세한 내용은 ["TR-4714: NetApp SnapCenter를 사용하는 SQL Server 모범 사례 가이드"](#)를 참조하십시오.

### T-SQL 스냅샷을 사용하여 데이터베이스 보호

Microsoft는 SQL Server 2022에서 백업 작업의 스크립팅 및 자동화 경로를 제공하는 T-SQL 스냅샷을 도입했습니다. 전체 크기 복제본을 수행하는 대신 스냅샷을 위해 데이터베이스를 준비할 수 있습니다. 데이터베이스를 백업할 준비가 되면 ONTAP REST API를 활용하여 스냅샷을 생성할 수 있습니다.

다음은 샘플 백업 워크플로우입니다.

1. alter 명령으로 데이터베이스를 고정합니다. 이렇게 하면 기본 스토리지에서 정합성이 보장되는 스냅샷을 위해 데이터베이스가 준비됩니다. 고정 후에는 데이터베이스를 해제하고 backup 명령을 사용하여 스냅샷을 기록할 수 있습니다.
2. 새 백업 그룹 및 백업 서버 명령을 사용하여 스토리지 유닛에서 여러 데이터베이스의 스냅샷을 동시에 수행합니다.
3. 데이터베이스 워크로드가 여러 스토리지 유닛에 걸쳐 있는 경우 정합성 보장 그룹을 생성하여 관리 작업을 간소화합니다. 정합성 보장 그룹은 단일 유닛으로 관리되는 스토리지 유닛의 모음입니다.
4. 전체 백업 또는 copy\_only 전체 백업을 수행합니다. 이러한 백업은 msdb에도 기록됩니다.
5. 스냅샷 전체 백업 후 일반 스트리밍 접근 방식으로 생성된 로그 백업을 사용하여 시점 복구를 수행합니다. 원하는 경우 스트리밍 차등 백업도 지원됩니다.

자세한 내용은 ["T-SQL 스냅샷에 대한 Microsoft 설명서를 제공합니다"](#).

 \* NetApp는 SnapCenter를 사용하여 스냅샷 복사본을 생성할 것을 권장합니다 \*. 위에서 설명한 T-SQL 방법도 작동하지만 SnapCenter는 백업, 복원 및 복제 프로세스를 완벽하게 자동화합니다. 또한 검색을 수행하여 올바른 스냅샷이 생성되도록 합니다.

## 재해 복구

### 재해 복구

엔터프라이즈 데이터베이스 및 애플리케이션 인프라에서는 다운타임을 최소화하면서 자연 재해 또는 예상치 못한 비즈니스 운영 중단으로부터 보호하기 위해 복제를 필요로 하는 경우가 많습니다.

SQL Server 상시 가용성 그룹 복제 기능은 탁월한 옵션일 수 있으며, NetApp는 데이터 보호를 상시 가동과 통합하는 옵션을 제공합니다. 그러나 다음 옵션을 사용하여ONTAP 복제 기술을 고려할 수도 있습니다.

### SnapMirror를 참조하십시오

SnapMirror 기술은 LAN 및 WAN을 통한 데이터 복제를 위한 빠르고 유연한 엔터프라이즈 솔루션을 제공합니다. SnapMirror 기술은 초기 미러가 생성된 후에 변경된 데이터 블록만 타겟으로 전송하므로 네트워크 대역폭 요구사항을 크게 줄여줍니다. 동기 또는 비동기 모드로 구성할 수 있습니다. NetApp ASA의 SnapMirror 동기식 복제는 SnapMirror 액티브 동기화를 사용하여 구성됩니다.

### SnapMirror 활성 동기화

많은 고객의 경우 비즈니스 연속성에 단순히 데이터의 원격 복사본을 소유하는 것이 아니라 SnapMirror 액티브 동기화를 사용하여 NetAppONTAP에서 가능한 데이터를 빠르게 사용할 수 있는 기능이 필요합니다.

SnapMirror가 활성 동기화에서는 기본적으로 LUN 데이터의 독립적인 복사본을 유지하지만 해당 LUN의 단일 인스턴스를 제공하기 위해 협력하는 두 개의ONTAP 시스템이 있습니다. 호스트 관점에서 보면 단일 LUN 엔티티입니다. SnapMirror 활성 동기화는 iSCSI/FC 기반 LUN에 대해 지원됩니다.

SnapMirror 액티브 동기화는 RPO=0 복제를 제공할 수 있으며 두 개의 독립 클러스터 간에 쉽게 구현할 수 있습니다. 두 개의 데이터 복사본이 동기화되면 두 클러스터에서 쓰기 작업만 미러링하면 됩니다. 한 클러스터에서 쓰기가 발생하면 다른 클러스터에 복제됩니다. 쓰기가 양쪽 사이트에서 완료된 경우에만 호스트에 인식됩니다. 이 프로토콜 분할 동작 이외에 두 클러스터는 정상적인ONTAP 클러스터입니다.

SM-AS의 주요 활용 사례 중 하나는 세분화된 복제입니다. 모든 데이터를 단일 유닛으로 복제하기 원하지 않거나 특정 워크로드를 선택적으로 페일오버할 수 있어야 하는 경우가 있습니다.

SM-AS의 또 다른 주요 활용 사례는 Active-Active 작업이며, 성능 특성이 동일한 두 위치에 있는 서로 다른 두 클러스터에서 완전히 사용할 수 있는 데이터 복제본을 사용할 수 있도록 하고, 원하는 경우 사이트 간에 SAN을 확장할 필요가 없습니다. 애플리케이션이 지원되면 두 사이트에서 애플리케이션이 이미 실행되고 있을 수 있으므로 페일오버 작업 중 전체 RTO가 감소합니다.

### **SnapMirror을 참조하십시오**

다음은 SQL Server용 SnapMirror에 대한 권장사항입니다.

- 빠른 데이터 복구에 대한 수요가 높은 SnapMirror 액티브 동기화로 동기식 복제를 사용하고 RPO에 유연성을 제공하는 비동기식 솔루션을 사용합니다.
- SnapCenter를 사용하여 데이터베이스를 백업하고 원격 클러스터로 스냅샷을 복제하는 경우 일관성을 위해 컨트롤러에서 SnapMirror 업데이트를 예약하지 마십시오. 대신, 전체 또는 로그 백업이 완료된 후 SnapCenter에서 SnapMirror를 업데이트하도록 SnapMirror를 활성화합니다.
- 모든 클러스터 노드가 SnapMirror 복제 작업을 공유할 수 있도록 클러스터의 서로 다른 노드에서 SQL Server 데이터를 포함하는 스토리지 유닛의 균형을 조정합니다. 이러한 분산은 노드 리소스 사용을 최적화합니다.

SnapMirror에 대한 자세한 내용은 를 참조하십시오 ["TR-4015:ONTAP 9용 SnapMirror 구성 및 모범 사례 가이드"](#).

### **SnapMirror 활성 동기화**

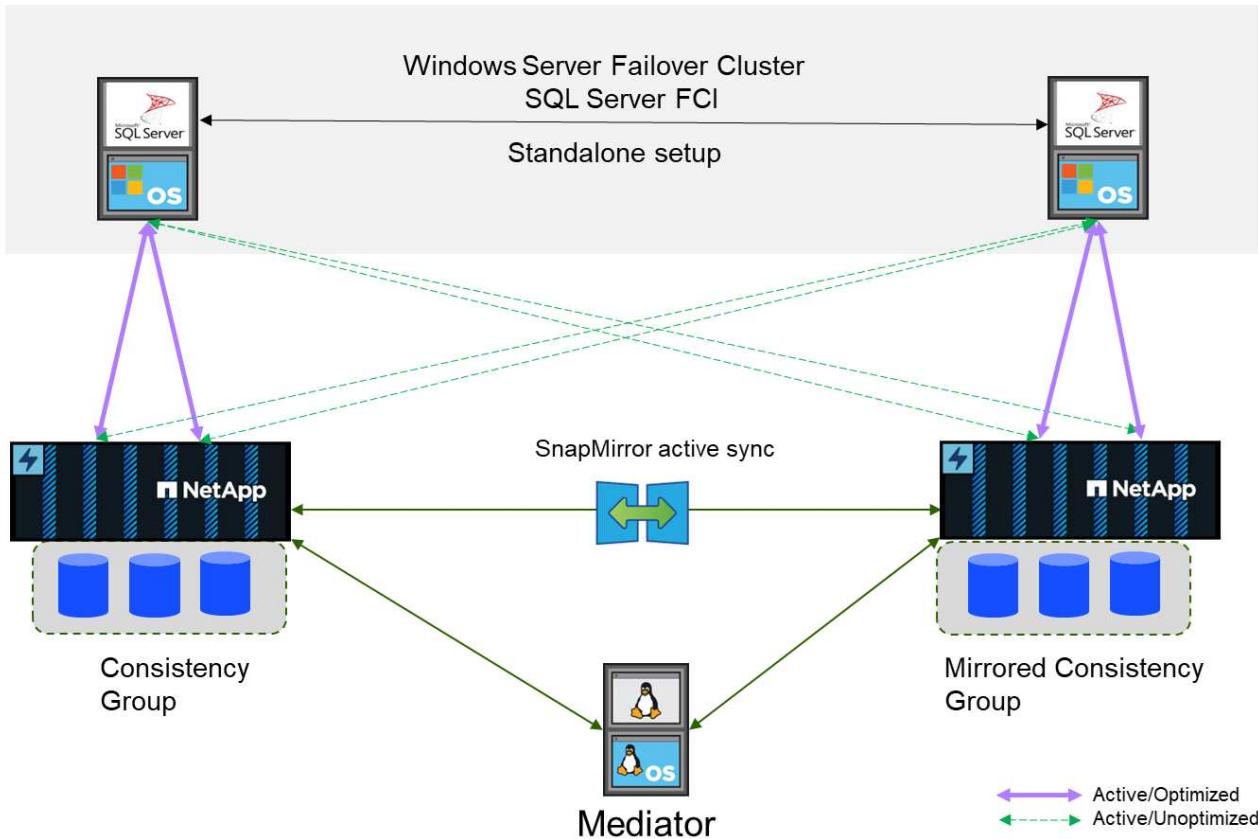
개요

SnapMirror Active Sync를 사용하면 개별 SQL Server 데이터베이스 및 애플리케이션이 스토리지 및 네트워크가 중단되어도 작업을 계속할 수 있으며, 수동 작업 없이 투명한 스토리지 페일오버를 수행할 수 있습니다.

SnapMirror 액티브 동기화는 비즈니스 연속성 및 재해 복구를 위해 동기식 양방향 복제를 제공하는 대칭 액티브/액티브 아키텍처를 지원합니다. 또한 여러 장애 도메인에 걸쳐 데이터에 대한 동시 읽기 및 쓰기 액세스를 통해 중요한 SAN 워크로드에 대한 데이터 액세스를 보호함으로써 무중단 운영을 보장하고 재해 또는 시스템 장애 시 다운타임을 최소화할 수 있습니다.

SQL Server 호스트는 FC(Fibre Channel) 또는 iSCSI LUN을 사용하여 스토리지에 액세스합니다. 복제된 데이터의 복제본을 호스팅하는 각 클러스터 간의 복제입니다. 이 기능은 스토리지 수준 복제이므로 독립 실행형 호스트 또는 장애 조치 클러스터 인스턴스에서 실행되는 SQL Server 인스턴스는 클러스터 중 하나에서 읽기/쓰기 작업을 수행할 수 있습니다. 계획 및 구성 단계는 를 ["SnapMirror 활성 동기화에 대한 ONTAP 문서"](#) 참조하십시오.

### **대칭 액티브/액티브 방식의 SnapMirror 아키텍처**



- 동기 복제\*\*

정상 작동 시 각 복제본은 항상 RPO=0 동기식 복제본이며 한 가지 예외가 있습니다. 데이터를 복제할 수 없는 경우 ONTAP는 데이터를 복제해야 하는 요구 사항을 해제하고 다른 사이트의 LUN이 오프라인 상태가 되는 동안 한 사이트에서 입출력 서비스를 재개합니다.

- 스토리지 하드웨어\*\*

다른 스토리지 재해 복구 솔루션과 달리 SnapMirror Active Sync는 비대칭적 플랫폼 유연성을 제공합니다. 각 사이트의 하드웨어는 동일할 필요가 없습니다. 이 기능을 사용하면 SnapMirror 액티브 동기화를 지원하는 데 사용되는 하드웨어를 적절한 크기로 조정할 수 있습니다. 전체 운영 워크로드를 지원해야 하는 경우 원격 스토리지 시스템이 기본 사이트와 동일할 수 있지만 재해로 인해 I/O가 감소할 경우 원격 사이트의 소규모 시스템보다 비용 효율적입니다.

- ONTAP 중재자\*\*

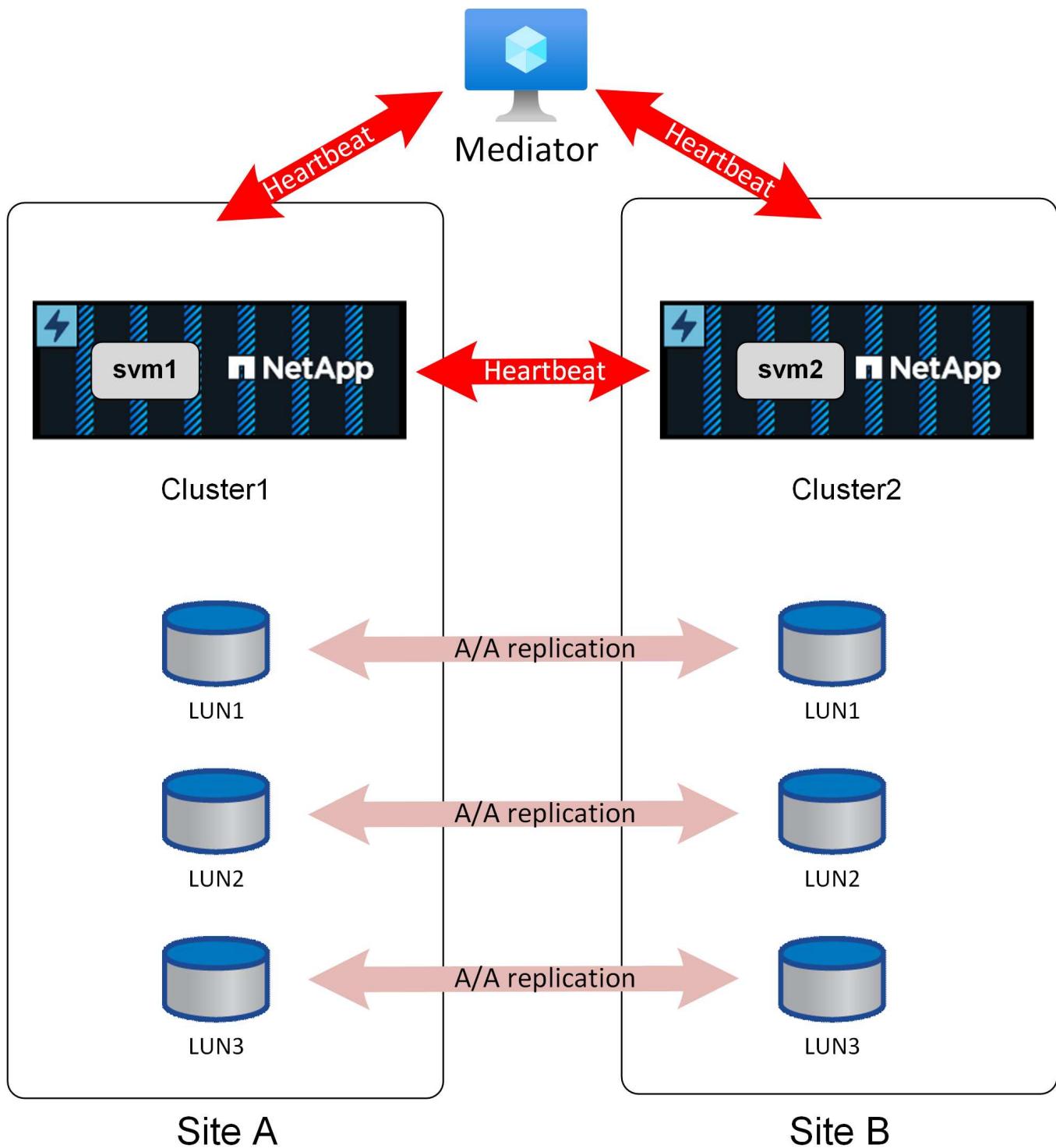
ONTAP 중재자는 NetApp 지원에서 다운로드되는 소프트웨어 응용 프로그램이며 일반적으로 작은 가상 컴퓨터에 구축됩니다. ONTAP 중재자는 타이브레이커가 아닙니다. SnapMirror 활성 동기화 복제에 참여하는 두 클러스터에 대한 대체 통신 채널입니다. 자동화된 운영은 직접 연결과 중재자를 통해 파트너로부터 받은 응답에 따라 ONTAP에 의해 주도됩니다.

#### ONTAP 중재자

장애 조치를 안전하게 자동화하려면 중재자가 필요합니다. 이상적으로는 독립적인 3차 사이트에 배치되지만 복제에 참여하는 클러스터 중 하나와 같은 위치에 배치하면 대부분의 요구 사항에 대해 여전히 기능을 수행할 수 있습니다.

중재자는 실제로 타이브레이커가 아니지만 실제로 제공하는 기능을 제공합니다. 어떤 작업도 수행하지 않고 클러스터 간

통신을 위한 대체 통신 채널을 제공합니다.



자동 장애 조치의 가장 큰 과제는 브레인 분할 문제이며, 두 사이트가 서로 연결이 끊어지면 이 문제가 발생합니다. 어떻게 해야 합니까? 서로 다른 두 사이트가 자신을 데이터의 정상적인 복제본으로 지정하도록 하고 싶지 않지만, 단일 사이트가 상대 사이트의 실제 손실과 반대쪽 사이트와 통신할 수 없는 간의 차이를 어떻게 알 수 있습니까?

중재자가 사진을 입력하는 위치입니다. 세 번째 사이트에 배치되고 각 사이트에 해당 사이트에 대한 별도의 네트워크 연결이 있는 경우 각 사이트에 대한 추가 경로를 통해 다른 사이트의 상태를 확인할 수 있습니다. 위의 그림을 다시 보고 다음 시나리오를 고려하십시오.

- 중재자가 한 사이트 또는 두 사이트에서 작동하지 않거나 연결할 수 없는 경우 어떻게 됩니까?
  - 두 클러스터는 복제 서비스에 사용되는 동일한 링크를 통해 서로 계속 통신할 수 있습니다.
  - RPO=0 보호로 데이터를 계속 제공합니다
- 사이트 A에 장애가 발생하면 어떻게 됩니까?
  - 사이트 B는 두 통신 채널이 모두 다운되는 것을 볼 수 있습니다.
  - 사이트 B가 데이터 서비스를 인수하지만 RPO = 0 미러링이 없습니다
- 사이트 B에 장애가 발생하면 어떻게 됩니까?
  - 사이트 A는 두 통신 채널이 모두 다운되는 것을 볼 수 있습니다.
  - 사이트 A가 데이터 서비스를 인수하지만 RPO = 0 미러링이 없음

고려해야 할 다른 시나리오가 있습니다: 데이터 복제 링크의 손실. 사이트 간 복제 링크가 손실되면 RPO=0 미러링이 불가능할 것입니다. 그러면 어떻게 됩니까?

이는 선호 사이트 상태에 의해 제어됩니다. SM-AS 관계에서 사이트 중 하나가 다른 사이트에 대한 보조 사이트입니다. 이 작업은 일반 작업에는 영향을 주지 않으며 모든 데이터 액세스는 대칭이지만 복제가 중단되면 작업을 재개하려면 연결을 끊어야 합니다. 그 결과 기본 사이트가 미러링 없이 운영을 계속하고 복제 통신이 복구될 때까지 보조 사이트가 입출력 처리를 중지합니다.

기본 사이트

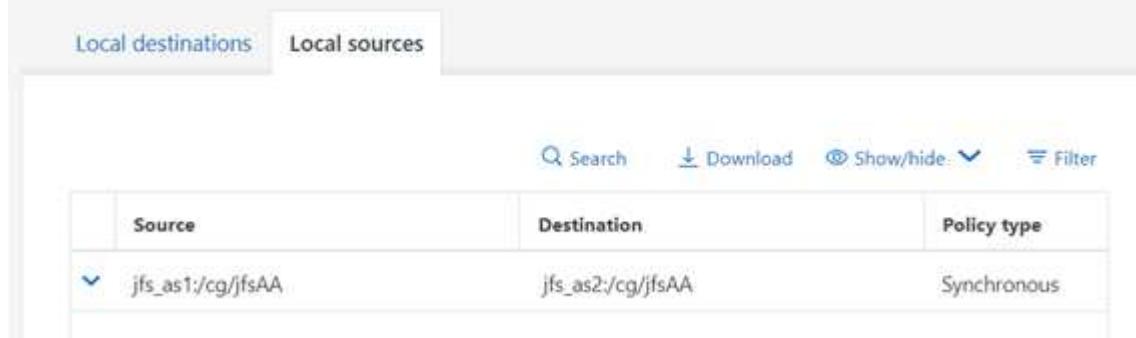
SnapMirror 액티브 동기화 동작은 대칭이며 한 가지 중요한 예외 기본 사이트 구성이 있습니다.

SnapMirror 활성 동기화는 한 사이트를 "소스"로 간주하고 다른 사이트는 "대상"으로 간주합니다. 이는 단방향 복제 관계를 의미하지만 입출력 동작에는 적용되지 않습니다. 복제는 양방향이고 대칭이며 입출력 응답 시간은 미러의 양쪽에서 동일합니다.

``source`` 지정은 기본 사이트를 제어합니다. 복제 링크가 손실되면 소스 복제본의 LUN 경로는 계속 데이터를 제공하고 대상 복제본의 LUN 경로는 복제가 다시 설정되고 SnapMirror가 동기식 상태로 다시 전환될 때까지 사용할 수 없게 됩니다. 그러면 경로가 데이터 제공을 재개합니다.

소스/대상 구성은 SystemManager를 통해 볼 수 있습니다.

## Relationships



The screenshot shows the 'Relationships' page in the SystemManager. The top navigation bar has tabs for 'Local destinations' and 'Local sources', with 'Local sources' currently selected. Below the tabs are buttons for 'Search', 'Download', 'Show/hide', and 'Filter'. The main area is a table with the following columns: 'Source', 'Destination', and 'Policy type'. There is one visible row: 'jfs\_as1:/cg/jfsAA' is the 'Source' and 'jfs\_as2:/cg/jfsAA' is the 'Destination', with 'Synchronous' as the 'Policy type'.

Source	Destination	Policy type
jfs_as1:/cg/jfsAA	jfs_as2:/cg/jfsAA	Synchronous

또는 CLI에서:

```
Cluster2::> snapmirror show -destination-path jfs_as2:/cg/jfsAA

          Source Path: jfs_as1:/cg/jfsAA
          Destination Path: jfs_as2:/cg/jfsAA
          Relationship Type: XDP
          Relationship Group Type: consistencygroup
          SnapMirror Schedule: -
          SnapMirror Policy Type: automated-failover-duplex
          SnapMirror Policy: AutomatedFailOverDuplex
          Tries Limit: -
          Throttle (KB/sec): -
          Mirror State: Snapmirrored
          Relationship Status: InSync
```

핵심은 소스는 클러스터 1의 SVM입니다. 위에서 언급한 바와 같이 "원본" 및 "대상"이라는 용어는 복제된 데이터의 흐름을 설명하지 않습니다. 두 사이트 모두 쓰기를 처리하여 반대쪽 사이트로 복제할 수 있습니다. 실제로 두 클러스터 모두 소스와 타겟입니다. 하나의 클러스터를 소스로 지정하면 복제 링크가 손실된 경우 읽기-쓰기 스토리지 시스템으로 존속하는 클러스터를 제어할 수 있습니다.

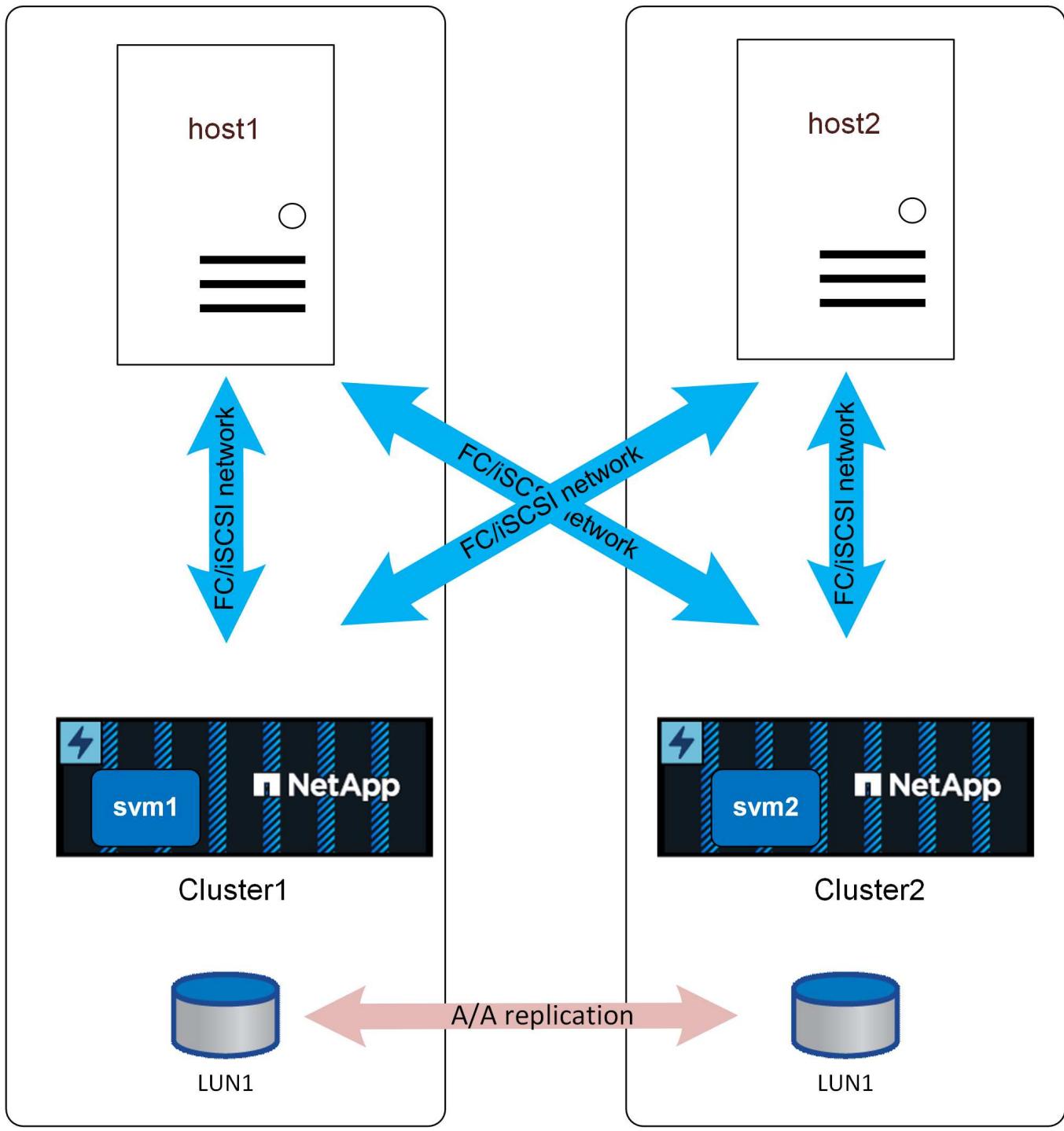
네트워크 토플로지

균일한 액세스

균일한 액세스 네트워킹은 호스트가 두 사이트(또는 동일한 사이트 내의 장애 도메인)의 경로를 액세스할 수 있음을 의미합니다.

SM-AS의 중요한 기능은 호스트의 위치를 알 수 있도록 스토리지 시스템을 구성하는 기능입니다. 특정 호스트에 LUN을 매핑할 때 해당 LUN이 지정된 스토리지 시스템에 근접한지 여부를 지정할 수 있습니다.

NetApp ASA 시스템은 클러스터의 모든 경로에 대해 액티브-액티브 다중 경로를 제공합니다. 이는 SM-AS 구성에도 적용됩니다.



Site A

Site B

## Active/Optimized Path

균일한 액세스의 경우 입출력이 WAN을 통과하게 됩니다. 이 클러스터는 전체 메시 네트워크 클러스터이며 모든 사용 사례에 바람직하지 않을 수도 있습니다.

두 사이트가 파이버 연결을 통해 100m 떨어져 있는 경우 WAN을 통해 추가 지연 시간을 감지할 수 없지만 사이트가 멀리 떨어져 있으면 두 사이트에서 읽기 성능이 저하됩니다. 가변 액세스 네트워크를 사용하는 ASA는 사이트 간 액세스

지연 시간에 영향을 주지 않으면서 ASA의 비용 및 기능 이점을 얻고, 두 사이트에 대해 사이트-로컬 읽기/쓰기 액세스를 허용하기 위한 호스트 근접 기능을 사용할 수 있습니다.

지연 시간이 짧은 구성의 SM-AS를 사용하는 ASA는 두 가지 흥미로운 이점을 제공합니다. 첫째로, IO는 2배 더 많은 경로를 사용하여 2배 더 많은 컨트롤러가 처리할 수 있기 때문에 모든 단일 호스트의 성능을 두 배 \* 제공합니다. 둘째, 단일 사이트 환경에서 호스트 액세스를 중단하지 않고 전체 스토리지 시스템을 손실할 수 있기 때문에 최고의 가용성을 제공합니다.

## 근접 설정

근접성은 특정 호스트 WWN 또는 iSCSI 이니시에이터 ID가 로컬 호스트에 속함을 나타내는 클러스터별 구성을 의미합니다. LUN 액세스를 구성하는 두 번째 선택적 단계입니다.

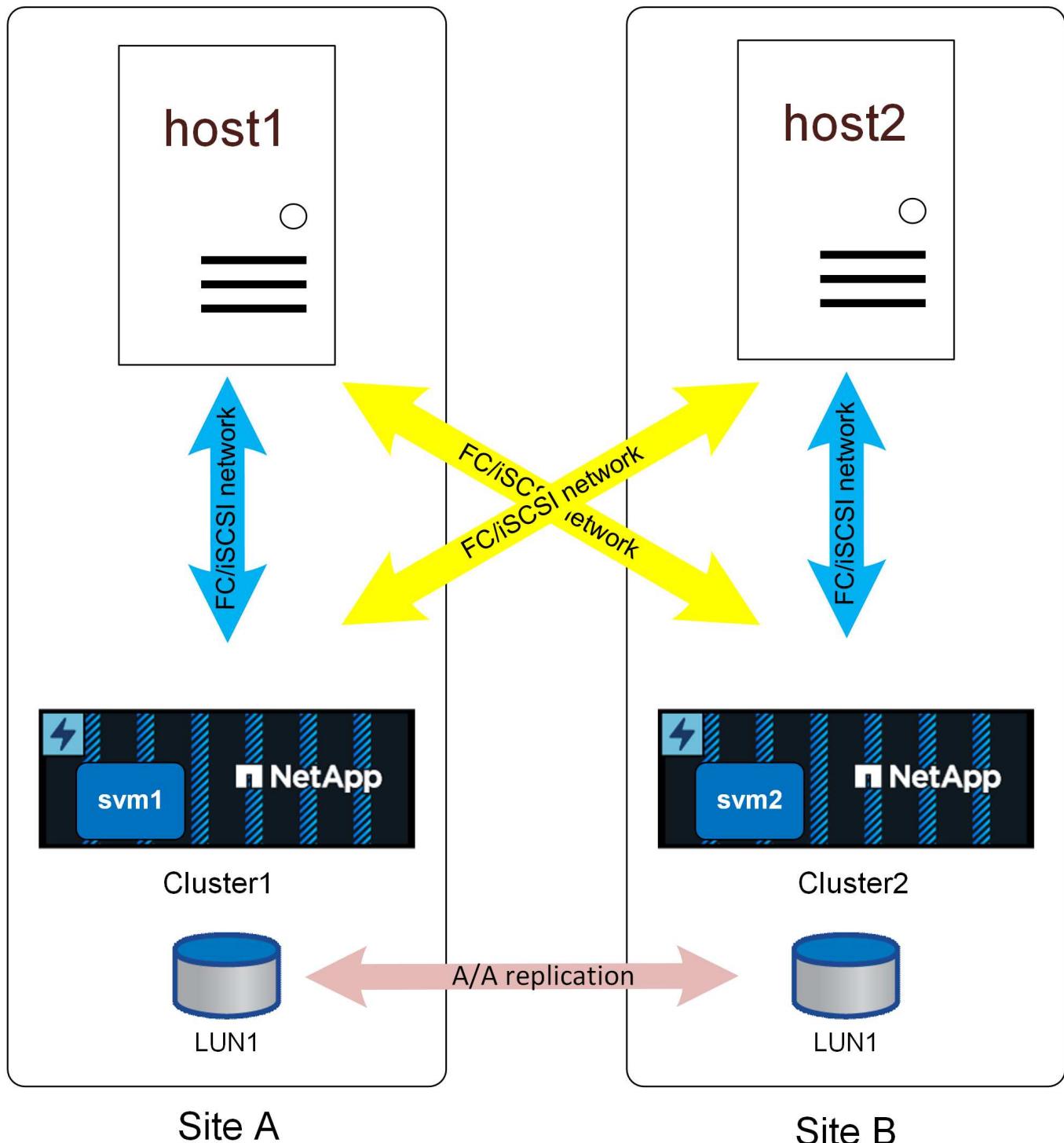
첫 번째 단계는 일반적인 igrup 구성입니다. 각 LUN은 해당 LUN에 액세스해야 하는 호스트의 WWN/iSCSI ID가 포함된 igrup에 매핑되어야 합니다. LUN에 대한 액세스 권한이 있는 호스트를 제어합니다.

두 번째 단계는 호스트 근접성을 구성하는 것입니다. 액세스를 제어하지 않고 \_priority\_를 제어합니다.

예를 들어 사이트 A의 호스트가 SnapMirror 활성 동기화로 보호되는 LUN을 액세스하도록 구성할 수 있고 SAN이 사이트 간에 확장되므로 사이트 A의 스토리지 또는 사이트 B의 스토리지를 사용하여 해당 LUN에 대한 경로를 사용할 수 있습니다

근접 설정이 없으면 두 스토리지 시스템 모두 활성/최적화된 경로를 홍보하기 때문에 해당 호스트는 두 스토리지 시스템을 동일하게 사용합니다. 사이트 간 SAN 대기 시간 및/또는 대역폭이 제한된 경우 이를 원하지 않을 수 있으며, 정상적인 작업 중에 각 호스트가 로컬 스토리지 시스템에 대한 경로를 우선적으로 사용하도록 할 수 있습니다. 호스트 WWN/iSCSI ID를 로컬 클러스터에 근위부 호스트로 추가하여 구성합니다. 이 작업은 CLI 또는 SystemManager에서 수행할 수 있습니다.

호스트 근접성이 구성된 경우 경로가 아래와 같이 나타납니다.

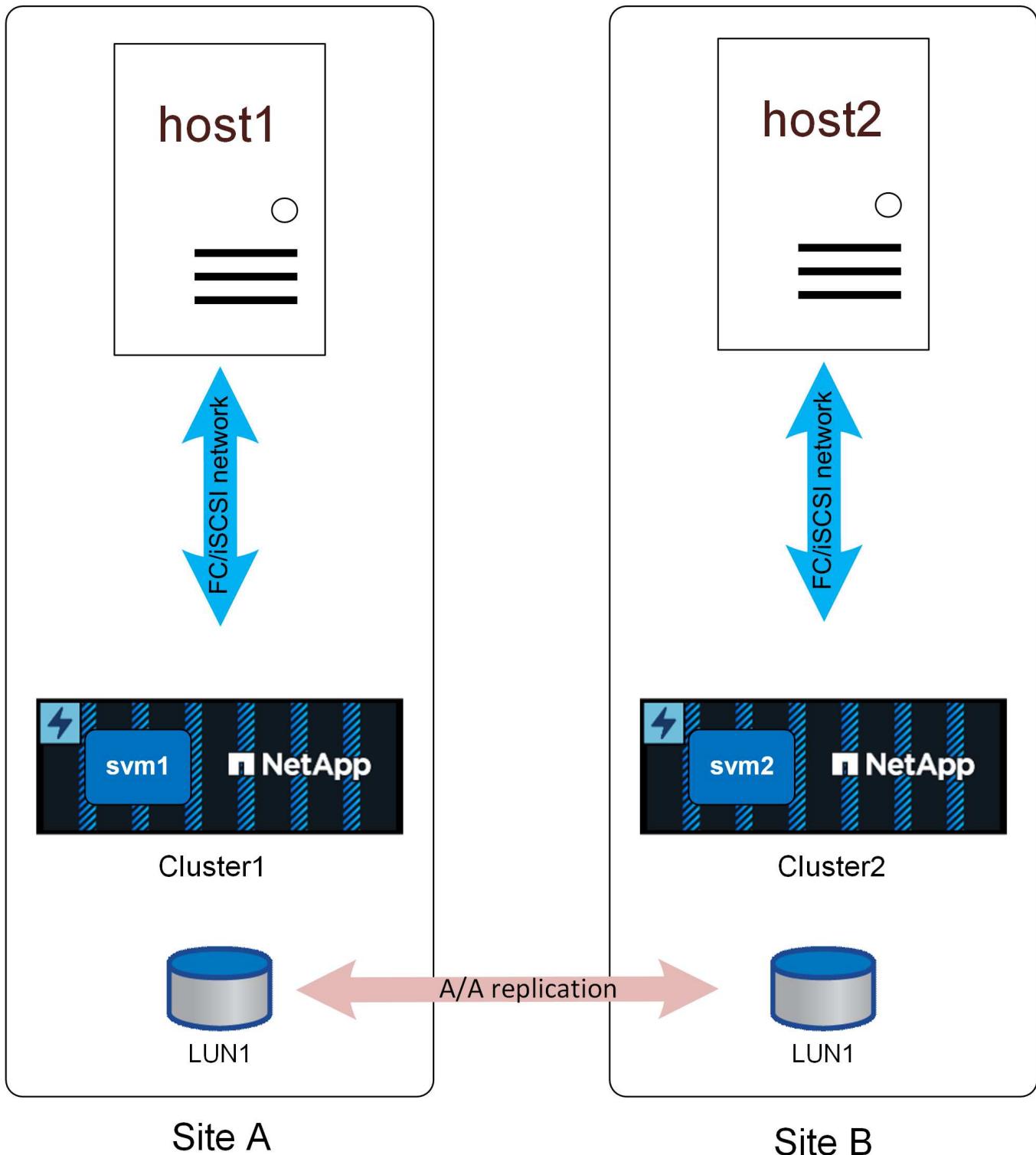


Active/Optimized Path

Active Path

## 비균일 액세스

비균일 액세스 네트워킹은 각 호스트가 로컬 스토리지 시스템의 포트에만 액세스할 수 있음을 의미합니다. SAN은 사이트(또는 동일한 사이트 내의 장애 도메인)에 걸쳐 확장되지 않습니다.



## Active/Optimized Path

이 접근 방식의 주요 이점은 SAN의 단순성입니다. 네트워크를 통해 SAN을 확장할 필요가 없어졌습니다. 일부 고객은 사이트 간에 지연 시간이 충분히 짧거나 사이트 간 네트워크를 통해 FC SAN 트래픽을 터널링할 수 있는 인프라가 부족합니다.

비균일 액세스의 단점은 복제 링크 손실을 비롯한 특정 장애 시나리오로 인해 일부 호스트가 스토리지에 액세스할 수 없게 된다는 것입니다. 클러스터 이외의 데이터베이스와 같이 단일 인스턴스로 실행되는 애플리케이션은 로컬 스토리지 연결이 끊긴 경우 해당 마운트의 단일 호스트에서만 실행 중이므로 실패합니다. 데이터는 여전히 보호되지만 데이터베이스 서버는 더 이상 액세스할 수 없습니다. 원격 사이트에서 자동 프로세스를 통해 재시작해야 합니다. 예를 들어 VMware HA는 한 서버에서 모든 경로 다운 상황을 감지하고 경로를 사용할 수 있는 다른 서버에서 VM을 다시 시작할 수 있습니다.

반면, Oracle RAC와 같은 클러스터된 애플리케이션은 두 개의 사이트에서 동시에 사용할 수 있는 서비스를 제공할 수 있습니다. 사이트를 잃는다고 해서 애플리케이션 서비스 전체가 손실되는 것은 아닙니다. 인스턴스는 계속 사용할 수 있으며 정상적인 사이트에서 실행됩니다.

대부분의 경우, 사이트 간 링크를 통해 스토리지에 액세스하는 애플리케이션의 지연 시간 오버헤드가 허용할 수 없는 경우가 많습니다. 즉, 사이트의 스토리지 손실로 인해 장애가 발생한 사이트의 서비스를 종료해야 하기 때문에 일관된 네트워킹의 가용성 향상은 최소화됩니다.

로컬 클러스터를 통한 중복 경로가 있습니다. 이러한 경로는 단순성을 위해 이러한 다이어그램에 표시되지 않습니다. ONTAP 스토리지 시스템은 HA 자체이므로 컨트롤러 장애가 사이트 장애로 이어질 수 없습니다. 영향을 받는 사이트에서 사용되는 로컬 경로가 변경될 뿐입니다.

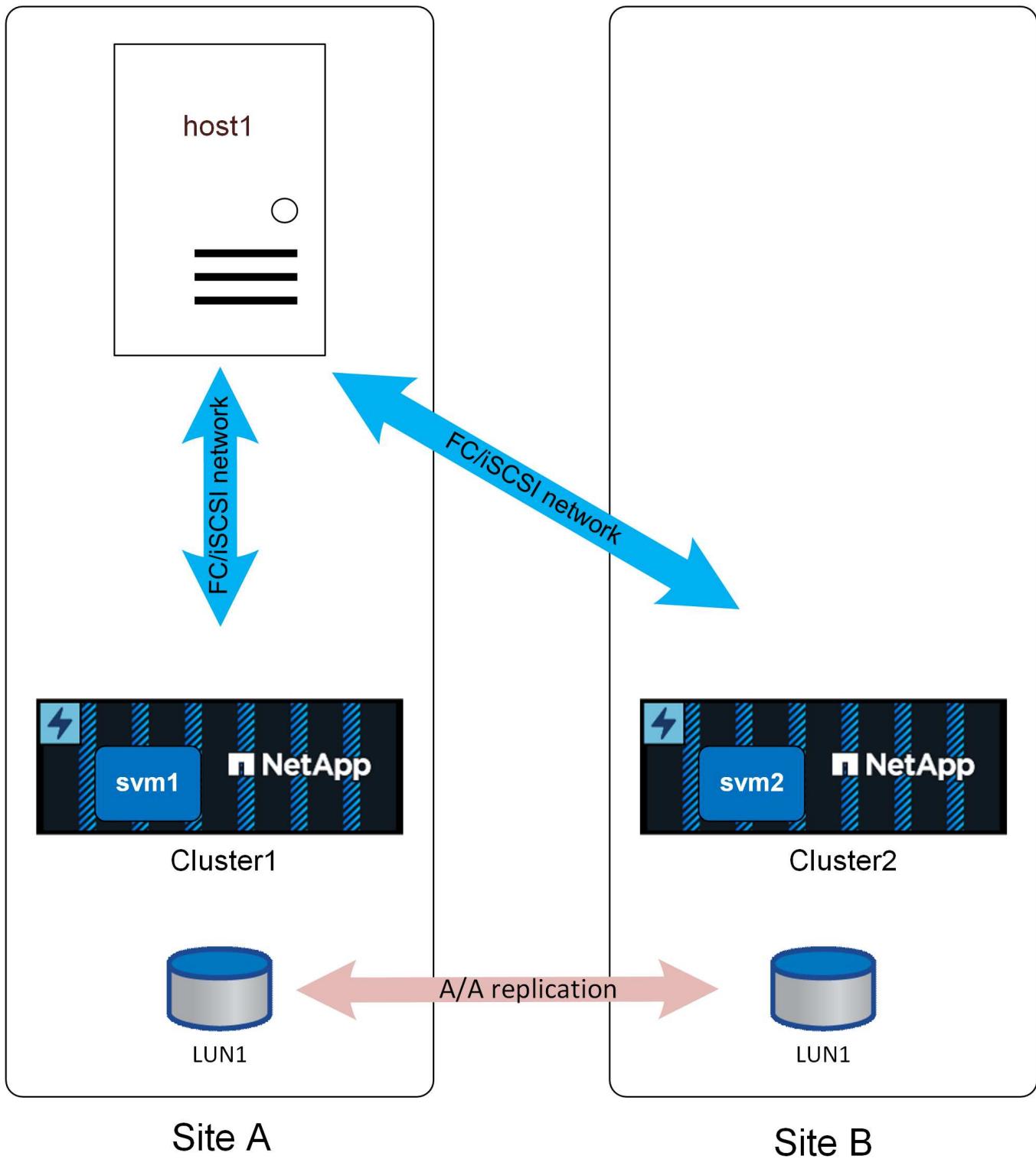
개요

여러 가지 방법으로 SnapMirror 활성 동기화에서 작동하도록 SQL Server를 구성할 수 있습니다. 정답은 사용 가능한 네트워크 연결, RPO 요구사항 및 가용성 요구사항에 따라 다릅니다.

**SQL Server**의 독립 실행형 인스턴스입니다

파일 레이아웃 및 서버 구성에 대한 Best Practice는 설명서에서 권장하는 방법과 동일합니다 "[ONTAP 기반 SQL Server](#)".

독립 실행형 설정을 사용하면 한 사이트에서만 SQL Server를 실행할 수 있습니다. 아마도 "균일" access가 사용될 것입니다.



균일한 액세스를 사용하더라도 두 사이트의 스토리지 장애로 인해 데이터베이스 작업이 중단되지 않습니다. 데이터베이스 서버가 포함된 사이트의 전체 사이트 장애는 물론 중단이 발생할 수 있습니다.

일부 고객은 사전 구성된 SQL Server 설정으로 원격 사이트에서 실행 중인 OS를 구성할 수 있으며, 운영 인스턴스의 빌드 버전과 동일한 버전으로 업데이트할 수 있습니다. 장애 조치에는 대체 사이트에서 해당 독립 실행형 SQL Server 인스턴스를 활성화하고, LUN을 검색하고, 데이터베이스를 시작해야 합니다. 스토리지 측에서 별도의 작업이 필요하지 않으므로 Windows Powershell cmdlet을 사용하여 전체 프로세스를 자동화할 수 있습니다.

"비균일" 액세스를 사용할 수도 있지만 데이터베이스에 스토리지에 대한 가용 경로가 없기 때문에 데이터베이스 서버가 있는 스토리지 시스템에 장애가 발생하면 데이터베이스가 중단될 수 있습니다. 일부 경우에는 이 방법이 여전히 허용됩니다. SnapMirror 활성 동기화는 여전히 RPO=0 데이터 보호를 제공하며, 사이트 장애 발생 시 나머지 복제본은 활성 상태이며 위에서 설명한 것과 동일한 액세스 절차를 사용하여 작업을 재개할 수 있습니다.

단순하고 자동화된 페일오버 프로세스는 가상화 호스트를 사용하여 더욱 쉽게 구성할 수 있습니다. 예를 들어, SQL Server 데이터 파일이 부팅 VMDK와 함께 보조 스토리지에 동기식으로 복제되면 재해 발생 시 전체 환경을 대체 사이트에서 활성화할 수 있습니다. 관리자는 정상적인 사이트에서 호스트를 수동으로 활성화하거나 VMware HA와 같은 서비스를 통해 프로세스를 자동화할 수 있습니다.

### **SQL Server** 장애 조치 클러스터 인스턴스입니다

SQL Server 장애 조치 인스턴스는 물리적 서버 또는 가상 서버에서 게스트 운영 체제로 실행되는 Windows 장애 조치 클러스터에서 호스팅될 수도 있습니다. 이 다중 호스트 아키텍처는 SQL Server 인스턴스와 스토리지 복구 기능을 제공합니다. 이러한 배포는 향상된 성능을 유지하면서 강력한 장애 조치 프로세스를 원하는 수요가 많은 환경에 유용합니다. 장애 조치 클러스터 설정에서 호스트 또는 운영 스토리지가 영향을 받으면 SQL 서비스가 보조 호스트로 페일오버되고, 동시에 보조 스토리지를 사용하여 IO를 제공할 수 있습니다. 자동화 스크립트나 관리자 개입이 필요하지 않습니다.

### 실패 시나리오

전체 SnapMirror 액티브 동기화 애플리케이션 아키텍처를 계획하려면 계획된 페일오버 및 예상치 못한 다양한 페일오버 시나리오에서 SM-AS가 어떻게 반응하는지 이해해야 합니다.

다음 예에서는 사이트 A가 기본 사이트로 구성되어 있다고 가정합니다.

### 복제 접속이 끊어졌습니다

SM-AS 복제가 중단되면 클러스터에서 변경 내용을 반대편 사이트로 복제할 수 없기 때문에 쓰기 입출력을 완료할 수 없습니다.

### 사이트 A(기본 사이트)

기본 사이트에서 복제 링크 실패의 결과는 ONTAP가 복제된 쓰기 작업을 다시 시도하기 때문에 쓰기 입출력 처리가 약 15초 동안 일시 중지되는 것입니다. 이 경우 복제 링크에 도달할 수 없는 것으로 판단됩니다. 15초가 지나면 사이트 A 시스템이 읽기 및 쓰기 IO 처리를 재개합니다. SAN 경로는 변경되지 않으며 LUN은 온라인 상태로 유지됩니다.

### 사이트 B

사이트 B는 SnapMirror 활성 동기화 기본 사이트가 아니므로 약 15초 후에 해당 LUN 경로를 사용할 수 없게 됩니다.

### 스토리지 시스템 장애

스토리지 시스템 장애의 결과는 복제 링크 손실의 결과와 거의 동일합니다. 정상적인 사이트에서 약 15초의 입출력 일시 중지 시간이 발생합니다. 15초가 지나면 평소와 같이 해당 사이트에서 입출력이 재개됩니다.

### 중재자의 상실

중재자 서비스는 스토리지 운영을 직접 제어하지 않습니다. 클러스터 간 대체 제어 경로 역할을 합니다. 이는 주로 브레인 분할 시나리오의 위험 없이 장애 조치를 자동화하는 데 있습니다. 정상 작동 시 각 클러스터가 파트너에 변경 사항을 복제하고 있으므로 각 클러스터가 온라인 상태이고 데이터를 제공하고 있는지 확인할 수 있습니다. 복제 링크가 실패하면 복제가 중지됩니다.

안전한 자동 폐일오버를 위해 중재자가 필요한 이유는 스토리지 클러스터에서 양방향 통신 손실이 네트워크 중단이나 실제 스토리지 장애로 인한 것인지 여부를 확인할 수 없기 때문입니다.

중재자는 각 클러스터에서 파트너 상태를 확인할 수 있는 대체 경로를 제공합니다. 시나리오는 다음과 같습니다.

- 클러스터가 파트너에게 직접 연락할 수 있는 경우 복제 서비스가 작동합니다. 별도의 조치가 필요 없습니다.
- 기본 사이트가 파트너에게 직접 연락하거나 중재자를 통해 연락할 수 없는 경우, 해당 파트너가 실제로 사용할 수 없거나 격리되어 해당 LUN 경로를 오프라인으로 설정한 것으로 간주됩니다. 그러면 기본 사이트가 RPO=0 상태를 해제하고 읽기 및 쓰기 입출력을 계속 처리합니다.
- 비선호 사이트가 해당 파트너에 직접 연락할 수 없지만 중재자를 통해 연락할 수 있는 경우 해당 경로가 오프라인 상태가 되고 복제 연결이 반환될 때까지 기다립니다.
- 비선호 사이트가 파트너에게 직접 연락하거나 운영 중재자를 통해 연락할 수 없는 경우, 파트너는 실제로 파트너를 사용할 수 없거나 격리되어 LUN 경로를 오프라인으로 전환했다고 가정합니다. 그러면 비기본 사이트가 RPO=0 상태를 해제하고 읽기 및 쓰기 입출력을 계속 처리합니다. 복제 소스의 역할을 가정하고 새로운 기본 사이트가 됩니다.

중재자를 완전히 사용할 수 없는 경우:

- 기본 설정되지 않은 사이트 또는 스토리지 시스템의 장애를 포함하여 어떠한 이유로든 복제 서비스에 장애가 발생하면 기본 사이트에서 RPO=0 상태를 해제하고 읽기 및 쓰기 입출력 처리를 재개합니다. 기본 사이트가 아닌 사이트는 해당 경로를 오프라인으로 전환합니다.
- 기본 사이트에 장애가 발생하면 기본 사이트가 반대 사이트가 실제로 오프라인 상태인지 확인할 수 없으므로 기본 사이트가 서비스를 다시 시작하는 것이 안전하지 않으므로 운영 중단이 발생합니다.

서비스를 복원하는 중입니다

사이트 간 연결 복원 또는 장애 시스템의 전원 켜기와 같은 장애가 해결되면 SnapMirror 활성 동기화 엔드포인트는 장애가 있는 복제 관계의 존재를 자동으로 감지하여 RPO=0 상태로 되돌립니다. 동기식 복제가 다시 설정되면 장애가 발생한 경로가 다시 온라인 상태가 됩니다.

대부분의 경우, 클러스터된 애플리케이션은 장애가 발생한 경로의 반환을 자동으로 감지하여 다시 온라인 상태로 돌아갑니다. 호스트 레벨 SAN 검사가 필요하거나 애플리케이션을 수동으로 다시 온라인으로 전환해야 하는 경우도 있습니다. 애플리케이션 및 구성 방법에 따라 다르며 일반적으로 이러한 작업을 쉽게 자동화할 수 있습니다. ONTAP 자체는 자동 복구이므로 RPO=0 스토리지 작업을 재개하기 위해 사용자 개입이 필요하지 않습니다.

수동 폐일오버

기본 사이트를 변경하려면 간단한 작업이 필요합니다. 클러스터 간 복제 동작 전환에 대한 권한으로 입출력이 1-2초 동안 일시 중지되지만, 그렇지 않으면 입출력이 영향을 받지 않습니다.

## 데이터 보안

SQL Server 데이터베이스 환경의 보안은 데이터베이스 자체 관리 이상의 다차원적인 작업입니다. ONTAP은 데이터베이스 인프라의 스토리지 측면을 보호하기 위해 설계된 몇 가지 고유한 기능을 제공합니다.

### Snapshot 복사본

스토리지 스냅샷은 타겟 데이터의 시점 복제본입니다. ONTAP은 다양한 정책을 설정하고 볼륨당 최대 1024개의

스냅샷을 저장하는 기능을 제공합니다. ONTAP의 스냅샷은 공간 효율적입니다. 공간은 원래 데이터 세트의 변경이 있을 때만 사용됩니다. 또한 읽기 전용입니다. 스냅샷은 삭제할 수 있지만 변경할 수는 없습니다.

경우에 따라 ONTAP에서 스냅샷을 직접 예약할 수 있습니다. 스냅샷을 생성하기 전에 SnapCenter와 같은 소프트웨어가 애플리케이션 또는 OS 작업을 조율해야 할 수도 있습니다. 워크로드에 가장 적합한 접근 방식이 무엇이든 공격적인 스냅샷 전략은 부팅 LUN에서 업무상 중요한 데이터베이스에 이르는 모든 항목의 백업에 자주 쉽게 액세스할 수 있도록 하여 데이터 보안을 제공할 수 있습니다.

 ONTAP 유연한 볼륨 또는 더 간단히 말하면 볼륨은 LUN과 동의어가 아닙니다. 볼륨은 파일 또는 LUN 같은 데이터를 위한 관리 컨테이너입니다. 예를 들어, 모든 LUN이 단일 볼륨에 포함된 8개의 LUN 스트라이프 세트에 데이터베이스를 배치할 수 있습니다.

스냅샷에 대한 자세한 내용은 를 참조하십시오 ["ONTAP 설명서."](#)

## 변조 방지 스냅샷

ONTAP 9.12.1부터 스냅샷은 읽기 전용일 뿐 아니라 실수로 또는 의도적으로 삭제되지 않도록 보호할 수 있습니다. 이 기능을 변조 방지 스냅샷이라고 합니다. 스냅샷 정책을 통해 보존 기간을 설정하고 적용할 수 있습니다. 생성된 스냅샷은 만료 날짜에 도달할 때까지 삭제할 수 없습니다. 관리 또는 지원 센터 재정의가 없습니다.

이렇게 하면 침입자, 악의적 내부자 또는 랜섬웨어 공격으로 인해 ONTAP 시스템 자체에 대한 액세스 결과가 발생했더라도 백업이 손상될 수 있습니다. 빈번한 스냅샷 스케줄과 결합하면 RPO가 매우 낮은 매우 강력한 데이터 보호 기능을 얻을 수 있습니다.

 패브릭 풀을 사용하여 변조 방지 스냅샷은 계층화할 수 없습니다.

변조 방지 스냅샷에 대한 자세한 내용은 를 참조하십시오 ["ONTAP 설명서."](#)

 새 ASA 플랫폼에서 `_vault_policy`를 사용하여 스냅샷을 원격 클러스터로 복제하고 대상을 잠가 스냅샷을 무단 변경하지 못하도록 합니다.

## SnapMirror 복제

스냅샷을 원격 시스템에 복제할 수도 있습니다. 여기에는 원격 시스템에 보존 기간이 적용되고 적용되는 변조 방지 스냅샷이 포함됩니다. 그 결과 로컬 스냅샷과 동일한 데이터 보호 이점을 얻을 수 있지만 데이터는 두 번째 스토리지 시스템에 위치합니다. 이렇게 하면 원래 스토리지를 폐기해도 백업이 손상되지 않습니다.

두 번째 시스템은 관리 보안을 위한 새로운 옵션도 엽니다. 예를 들어, 일부 NetApp 고객은 운영 스토리지 시스템과 2차 스토리지 시스템의 인증 자격 증명을 분리합니다. 한 명의 관리자가 두 시스템에 액세스할 수 없으므로 악의적인 관리자가 모든 데이터 복사본을 삭제할 수 없습니다.

SnapMirror에 대한 자세한 내용은 를 참조하십시오 ["ONTAP 설명서."](#)

## 스토리지 가상 머신

새로 구성된 ONTAP 스토리지 시스템은 가상 시스템을 생성할 때까지 어떤 사용자도 지원할 수 없기 때문에 새로 프로비저닝된 VMware ESX Server와 유사합니다. ONTAP를 사용하면 스토리지 관리의 가장 기본적인 장치가 되는 SVM(스토리지 가상 시스템)을 만들 수 있습니다. 각 SVM에는 자체 스토리지 리소스, 프로토콜 구성, IP 주소 및 FCP WWN이 있습니다. 이것이 ONTAP 멀티 테넌시의 토대입니다.

예를 들어, 중요 운영 워크로드에 1개의 SVM을, 개발 활동을 위해 다른 네트워크 세그먼트에 2차 SVM을 구성할 수

있습니다. 그런 다음 운영 SVM에 대한 액세스를 특정 관리자로 제한하고, 개발자에게는 개발 SVM에서 스토리지 리소스를 더욱 포괄적으로 제어할 수 있습니다. 특히 중요한 눈 전용 데이터를 저장하려면 재무 및 HR 팀에 세 번째 SVM을 제공해야 할 수도 있습니다.

SVM에 대한 자세한 내용은 [를 참조하십시오 "ONTAP 설명서."](#)

## 관리 RBAC

ONTAP는 관리 로그인을 위한 강력한 RBAC(역할 기반 액세스 제어)를 제공합니다. 일부 관리자는 전체 클러스터 액세스가 필요할 수 있고 일부 관리자는 특정 SVM에 대해서만 액세스해야 할 수도 있습니다. 고급 헬프데스크 직원은 볼륨 크기를 늘릴 수 있어야 합니다. 그 결과, 관리 사용자에게 업무 수행에 필요한 액세스 권한을 부여할 수 있으며 그 이상의 권한은 없습니다. 또한 다양한 공급업체의 PKI를 사용하여 이러한 로그인을 보호하고 ssh 키에만 대한 액세스를 제한하며 실패한 로그인 시도 잠금을 적용할 수 있습니다.

관리 액세스 제어에 대한 자세한 내용은 [를 참조하십시오 "ONTAP 설명서."](#)

## 다단계 인증(MFA)

이제 ONTAP 및 특정 기타 NetApp 제품은 다양한 방법을 사용하여 다단계 인증(MFA)을 지원합니다. 결과적으로 손상된 사용자 이름/암호만으로는 FOB 또는 스마트폰 앱과 같은 두 번째 요소의 데이터가 없는 보안 스레드가 아닙니다.

MFA(다중 요소 인증)에 대한 자세한 내용은 [를 참조하십시오 "ONTAP 설명서."](#)

## API RBAC입니다

자동화를 위해서는 API 호출이 필요하지만, 모든 툴에 전체 관리 액세스가 필요한 것은 아닙니다. 자동화 시스템의 보안을 유지하기 위해 RBAC는 API 레벨에서도 사용할 수 있습니다. 자동화 사용자 계정을 필요로 하는 API 호출로 제한할 수 있습니다. 예를 들어, 모니터링 소프트웨어는 변경 액세스가 필요하지 않고 읽기 액세스만 있으면 됩니다. 스토리지를 프로비저닝하는 워크플로우에는 스토리지를 삭제하는 기능이 필요하지 않습니다.

API RBAC에 대한 자세한 내용은 [를 참조하십시오 "ONTAP 설명서."](#)

## MAV(Multi-admin verification)

여러 "요소" 인증은 각각 고유한 자격 증명을 가진 두 명의 관리자가 특정 작업을 승인하도록 요구함으로써 더욱 더 나아가 수행할 수 있습니다. 여기에는 로그인 권한 변경, 진단 명령 실행, 데이터 삭제가 포함됩니다.

MAV(다중 관리자 확인)에 대한 자세한 내용은 [를 참조하십시오 "ONTAP 설명서."](#)

## 저작권 정보

Copyright © 2026 NetApp, Inc. All Rights Reserved. 미국에서 인쇄됨 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그레픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이센스와 고지사항이 적용됩니다.

본 소프트웨어는 NetApp에 의해 '있는 그대로' 제공되며 상품성 및 특정 목적에의 적합성에 대한 명시적 또는 묵시적 보증을 포함하여(이에 제한되지 않음) 어떠한 보증도 하지 않습니다. NetApp은 대체품 또는 대체 서비스의 조달, 사용 불능, 데이터 손실, 이익 손실, 영업 중단을 포함하여(이에 국한되지 않음), 이 소프트웨어의 사용으로 인해 발생하는 모든 직접 및 간접 손해, 우발적 손해, 특별 손해, 징벌적 손해, 결과적 손해의 발생에 대하여 그 발생 이유, 책임론, 계약 여부, 엄격한 책임, 불법 행위(과실 또는 그렇지 않은 경우)와 관계없이 어떠한 책임도 지지 않으며, 이와 같은 손실의 발생 가능성이 통지되었다 하더라도 마찬가지입니다.

NetApp은 본 문서에 설명된 제품을 언제든지 예고 없이 변경할 권리를 보유합니다. NetApp은 NetApp의 명시적인 서면 동의를 받은 경우를 제외하고 본 문서에 설명된 제품을 사용하여 발생하는 어떠한 문제에도 책임을 지지 않습니다. 본 제품의 사용 또는 구매의 경우 NetApp에서는 어떠한 특허권, 상표권 또는 기타 지적 재산권이 적용되는 라이센스도 제공하지 않습니다.

본 설명서에 설명된 제품은 하나 이상의 미국 특허, 해외 특허 또는 출원 중인 특허로 보호됩니다.

제한적 권리 표시: 정부에 의한 사용, 복제 또는 공개에는 DFARS 252.227-7013(2014년 2월) 및 FAR 52.227-19(2007년 12월)의 기술 데이터-비상업적 품목에 대한 권리(Rights in Technical Data -Noncommercial Items) 조항의 하위 조항 (b)(3)에 설명된 제한사항이 적용됩니다.

여기에 포함된 데이터는 상업용 제품 및/또는 상업용 서비스(FAR 2.101에 정의)에 해당하며 NetApp, Inc.의 독점 자산입니다. 본 계약에 따라 제공되는 모든 NetApp 기술 데이터 및 컴퓨터 소프트웨어는 본질적으로 상업용이며 개인 비용만으로 개발되었습니다. 미국 정부는 데이터가 제공된 미국 계약과 관련하여 해당 계약을 지원하는 데에만 데이터에 대한 전 세계적으로 비독점적이고 양도할 수 없으며 재사용이 불가능하며 취소 불가능한 라이센스를 제한적으로 가집니다. 여기에 제공된 경우를 제외하고 NetApp, Inc.의 사전 서면 승인 없이는 이 데이터를 사용, 공개, 재생산, 수정, 수행 또는 표시할 수 없습니다. 미국 국방부에 대한 정부 라이센스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

## 상표 정보

NETAPP, NETAPP 로고 및 <http://www.netapp.com/TM>에 나열된 마크는 NetApp, Inc.의 상표입니다. 기타 회사 및 제품 이름은 해당 소유자의 상표일 수 있습니다.