



# Microsoft SQL Server

## Enterprise applications

NetApp

January 02, 2026

# 目录

|                                |    |
|--------------------------------|----|
| Microsoft SQL Server .....     | 1  |
| 概述 .....                       | 1  |
| Microsoft SQL Server工作负载 ..... | 1  |
| 数据库配置 .....                    | 1  |
| CPU配置 .....                    | 1  |
| 内存配置 .....                     | 4  |
| 共享实例与专用实例 .....                | 7  |
| tempdb文件 .....                 | 8  |
| AFF或FAS系统上的存储配置 .....          | 9  |
| 概述 .....                       | 9  |
| 数据库文件和文件组 .....                | 11 |
| 存储效率 .....                     | 15 |
| 数据保护 .....                     | 19 |
| 灾难恢复 .....                     | 20 |
| ASA R2系统上的存储配置 .....           | 38 |
| 概述 .....                       | 38 |
| 数据库文件和文件组 .....                | 40 |
| 数据保护 .....                     | 43 |
| 灾难恢复 .....                     | 44 |
| 数据安全性 .....                    | 58 |
| Snapshot 副本 .....              | 58 |
| 防篡改快照 .....                    | 59 |
| SnapMirror 复制 .....            | 59 |
| Storage Virtual Machine .....  | 59 |
| 管理RBAC .....                   | 59 |
| 多因素身份验证(MFA) .....             | 59 |
| API RBAC .....                 | 60 |
| 多管理员验证(MAV) .....              | 60 |

# Microsoft SQL Server

## 概述

ONTAP为您的Microsoft SQL Server数据库提供企业级安全性和性能解决方案、同时还提供世界一流的工具来管理您的环境。



本文档将取代先前发布的技术报告\_TR-4590：《采用ONTAP的Microsoft SQL Server最佳实践指南》

NetApp假定读者具备以下方面的实践知识：

- ONTAP
- Microsoft SQL Server架构和管理
- NetApp SnapCenter作为备份软件、其中包括：
  - 适用于 Microsoft Windows 的 SnapCenter 插件
  - 适用于SQL Server的SnapCenter插件

本最佳实践部分的范围仅限于根据NetApp为存储基础架构建议的原则和首选标准进行技术设计。端到端实施已超出范围。

有关NetApp产品的兼容性信息，请参见"[NetApp 互操作性表工具（IMT）](#)"。

## Microsoft SQL Server工作负载

在部署SQL Server之前、您必须了解SQL Server数据库实例支持的应用程序的工作负载要求。每个应用程序对容量、性能和可用性有不同的要求、因此每个数据库的设计都应以最佳方式支持这些要求。许多组织使用应用程序要求定义SLA、将数据库分为多个管理层。SQL Server工作负载通常按如下所述进行分类：

- OLTP、通常是组织中最关键的数据库。这些数据库通常为面向客户的应用程序提供支持、并被视为公司核心运营的关键。任务关键型OLTP数据库及其支持的应用程序通常具有SLA、这些SLA要求高性能、容易受到性能下降的影响、并要求最大可用性。它们也可能是"无中断故障转移集群"或"无中断可用性"组的候选对象。这些类型数据库上的I/O混合通常表现为75%到90%的随机读取和25%到10%的写入。
- 决策支持系统(DSS)数据库、有时称为数据仓库。这些数据库对于许多依靠分析来开展业务的企业来说至关重要。在运行查询时、这些数据库对CPU利用率和磁盘读取操作非常敏感。在许多组织中、DSS数据库在月份、季度和年终都是最重要的此工作负载的读取I/O混合比例通常接近100%、并且IO吞吐量通常比IOPS更重要。

## 数据库配置

### CPU配置

SQL Server的性能与CPU和核心配置有多种依赖性。

## 超线程

超线程是指同时多线程(SMT)实施、可提高x86处理器上执行计算的并行性。SMT在Intel和AMD处理器上均可用。

超线程会导致逻辑CPU在操作系统中显示为物理CPU。然后、SQL Server会看到这些额外的CPU、并将其用作核心数多于物理内核数的情况。这可以通过提高并行处理能力显著提高性能。

此处需要注意的是、每个SQL Server版本都有自己的计算能力限制。有关详细信息，请参见 ["计算容量限制\(按SQL Server版本\)"](#)。

## 核心和许可

SQL Server许可有两种选择。第一种称为服务器+客户端访问许可证(CAL)模式；第二种是按处理器核心模式。尽管使用server + CAL策略可以访问SQL Server中提供的所有产品功能、但每个插槽的硬件限制为20个CPU核心。即使每个插槽的CPU核心数超过20个的服务器具有SQL Server Enterprise Edition + CAL、应用程序也无法在该实例上一次使用所有这些核心。

下图显示了启动后的SQL Server日志消息、指示核心限制的强制实施。

```
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:
-a "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
```

因此、要使用所有CPU、应使用每个处理器的核心许可证。有关SQL Server许可的详细信息、请参见 ["SQL Server 2022：您的现代数据平台"](#)。

## CPU关联性

除非遇到性能问题、否则您不太可能需要更改处理器关联性默认值、但仍然值得了解它们是什么以及它们的工作原理。

SQL Server通过两个选项支持处理器关联性：

- CPU相关性掩码
- 相关性I/O掩码

SQL Server使用操作系统中可用的所有CPU (如果选择了按处理器核心许可证)。此外、它还会为每个CPU创建计划程序for、以便充分利用任何给定工作负载的资源。多任务处理时、操作系统或服务上的其他应用程序可以将进程线程从一个处理器切换到另一个处理器。SQL Server是一个资源密集型应用程序、发生这种情况时、性能可能会受到影响。为了最大限度地减少影响、您可以对处理器进行配置、使所有SQL Server负载都定向到一组预先选择的处理器。这可通过使用CPU相关性掩码来实现。

相关性I/O掩码选项可将SQL Server磁盘I/O绑定到一部分CPU。在SQL Server OLTP环境中、此扩展可以显著提高发出I/O操作的SQL Server线程的性能。

## 最大并行度(MAXDOP)

默认情况下、如果选择了按处理器核心许可证、则SQL Server会在查询执行期间使用所有可用的CPU。

尽管这对于大型查询很有用、但它可能会出现发生原因性能问题并限制并发性。一种更好的方法是、将并行处理限制为单个CPU插槽中的物理核心数。例如、在具有两个物理CPU插槽且每个插槽具有12个核心的服务器上、无论是否使用超线程、均`MAXDOP`应设置为12。`MAXDOP`不能限制或指定要使用的CPU。相反、它会限制单个批处理查询可使用的CPU数量。



\* NetApp建议\*对于数据仓库等DSS，从50开始，`MAXDOP`并根据需要进行调整。请确保在进行更改时测量应用程序中的关键查询。

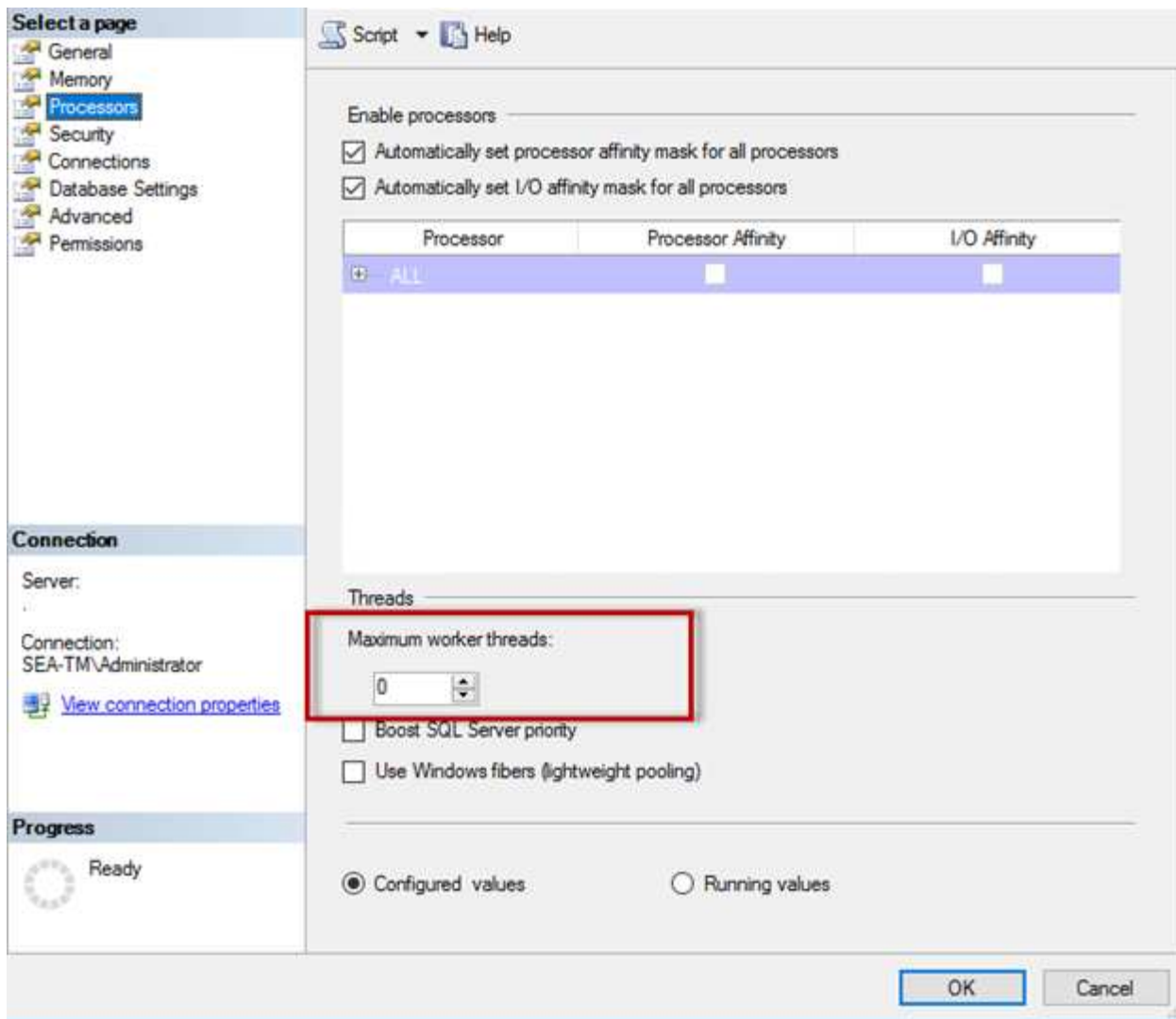
## 最大工作线程数

最大工作线程数选项有助于在大量客户端连接到SQL Server时优化性能。

通常、系统会为每个查询创建一个单独的操作系统线程。如果与SQL Server同时建立了数百个连接、则每个查询一个线程的配置可能会占用过多的系统资源。该`max worker threads`选项可以使SQL Server创建一个工作线程池、从而共同为大量查询请求提供服务、从而有助于提高性能。

默认值为0、这允许SQL Server在启动时自动配置工作线程数。这适用于大多数系统。最大工作线程数是一个高级选项、没有经验丰富的数据库管理员(Database Administrator、DBA)的帮助、不得更改。

应在何时配置SQL Server以使用更多工作线程？如果每个计划程序的平均工作队列长度超过1、则向系统添加更多线程可能会更有益、但前提是负载不受CPU制约或遇到任何其他繁重等待。如果出现上述任一情况、添加更多线程将毫无帮助、因为它们最终会等待其他系统瓶颈。有关最大工作线程数的详细信息，请参见["配置最大工作线程数服务器配置选项"](#)。



使用SQL Server Management Studio配置最大工作线程数。

以下示例显示了如何使用T-SQL配置最大工作线程数选项。

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

## 内存配置

下一节将介绍优化数据库性能所需的SQL Server内存设置。

## 最大服务器内存

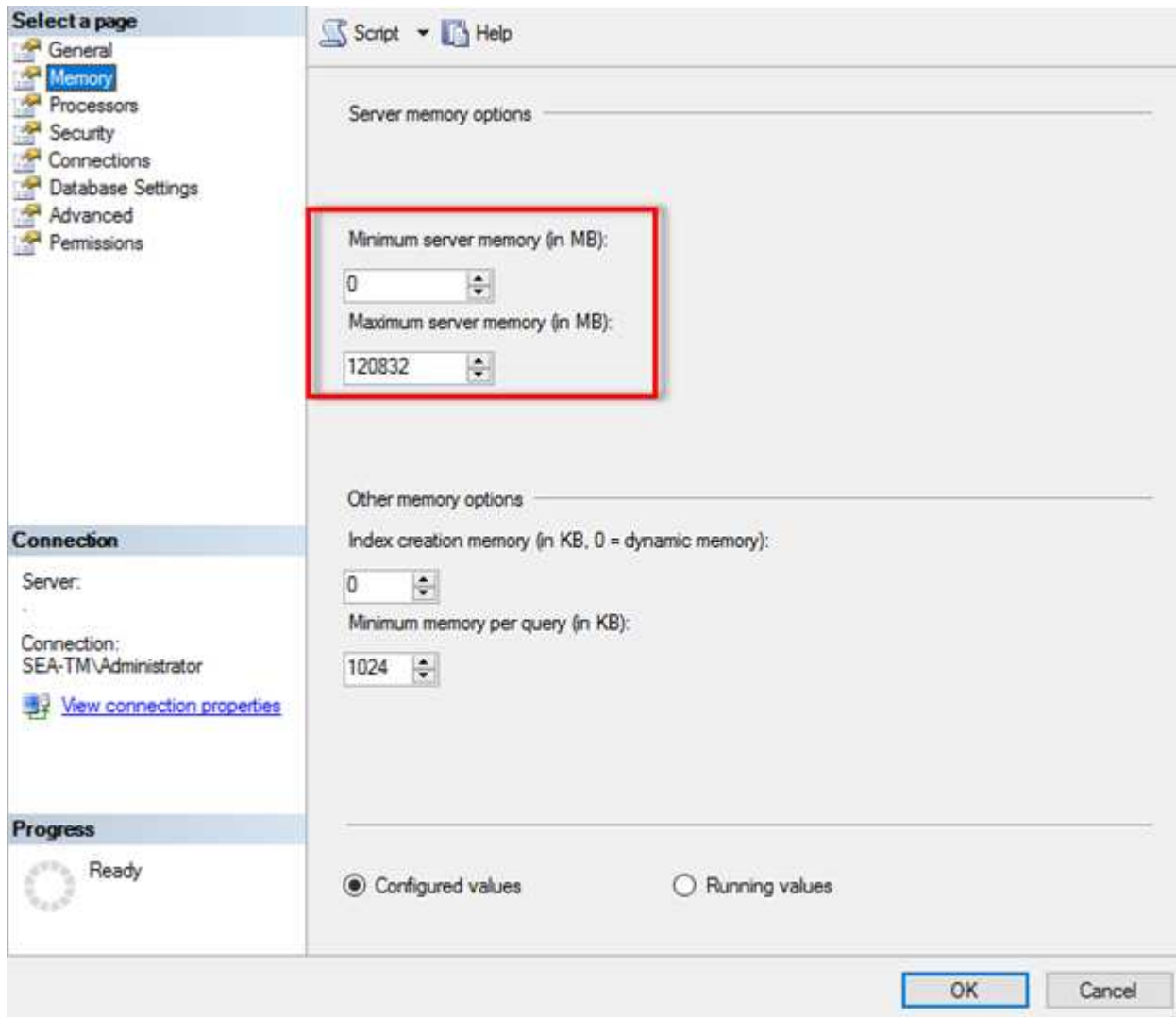
最大服务器内存选项用于设置SQL Server实例可以使用的最大内存量。通常、如果多个应用程序运行在运行SQL Server的同一服务器上、并且您希望保证这些应用程序具有足够的内存来正常运行、则使用此方法。

某些应用程序仅在启动时使用可用的任何内存、而不会请求额外内存、即使它们受到内存压力也是如此。这就是最大服务器内存设置的作用所在。

在包含多个SQL Server实例的SQL Server集群上、每个实例可能会争用资源。为每个SQL Server实例设置内存限制有助于保证每个实例的最佳性能。



\*RAM NetApp建议\*为操作系统至少保留4GB到6GB的内存，以避免出现性能问题。



使用SQL Server Management Studio调整最小和最大服务器内存。

使用SQL Server Management Studio调整最小或最大服务器内存需要重新启动SQL Server服务。您还可以使用Transact SQL (T-SQL)使用以下代码调整服务器内存：



```
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
```

## 非一致内存访问

非一致内存访问(Nonuniform Memory Access, NMA)是一种内存访问优化技术、有助于避免处理器总线上的额外负载。

如果在安装了SQL Server的服务器上配置了NUMA、则无需进行其他配置、因为SQL Server可识别NUMA、并且在NUMA硬件上性能良好。

## 索引创建内存

index create memory选项是另一个高级选项、通常不需要更改默认值。

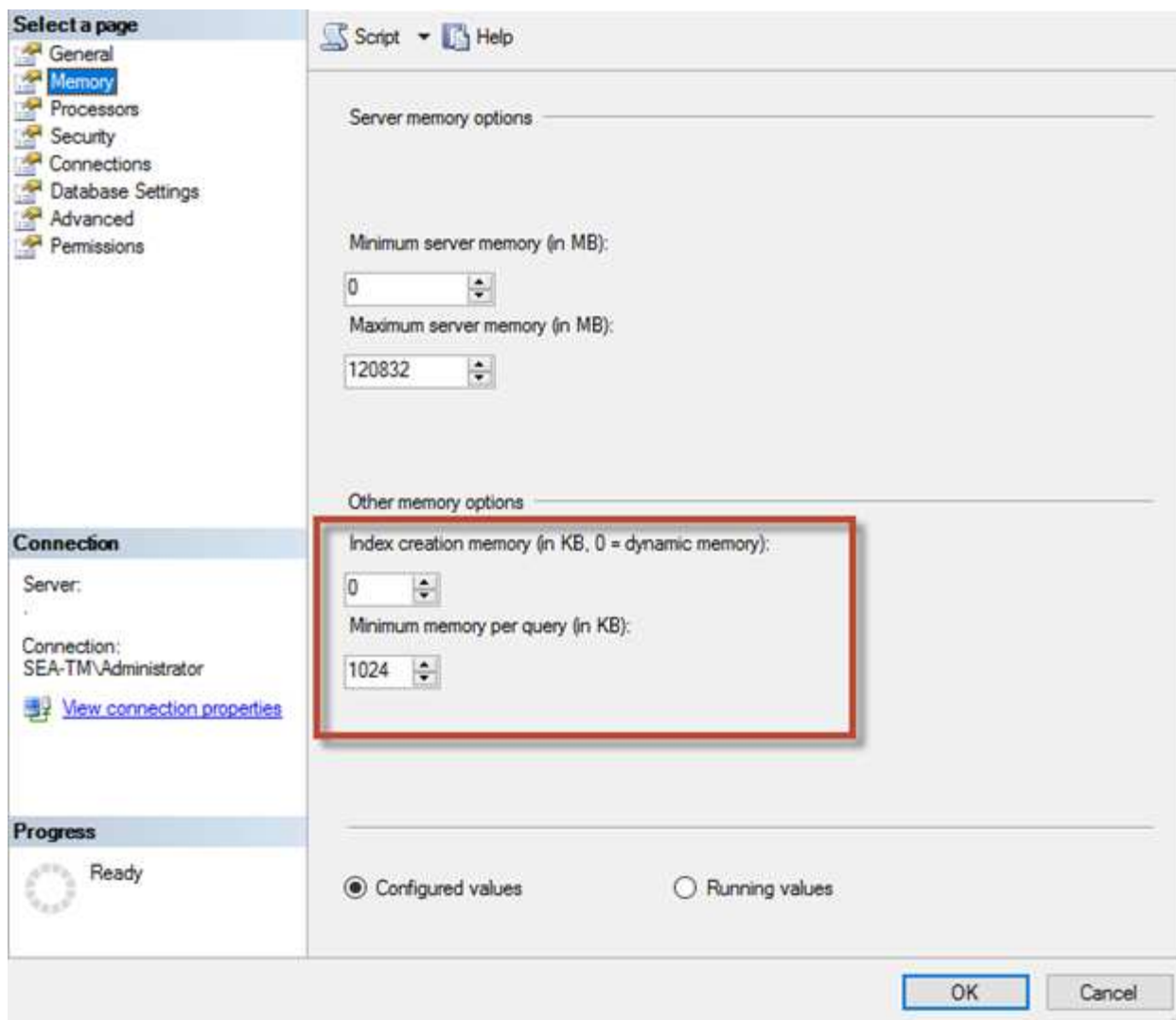
它控制最初为创建索引而分配的最大RAM量。此选项的默认值为0、这意味着它由SQL Server自动管理。但是，如果在创建索引时遇到困难，请考虑增加此选项的值。

## 每个查询的最小内存

运行查询时、SQL Server会尝试分配最佳内存量、以便高效运行。

默认情况下、"每个查询的最小内存"设置会为要运行的每个查询分配>=到1024KB的内存。最佳做法是、将此设置保留为默认值、以便SQL Server能够动态管理为索引创建操作分配的内存量。但是、如果SQL Server的RAM超过高效运行所需的RAM、则提高此设置可以提高某些查询的性能。因此、只要SQL Server、任何其他应用程序或操作系统未使用服务器上的可用内存、则提高此设置就可以提高SQL Server的整体性能。如果没有可用内存，则增加此设置可能会影响整体性能。





## 共享实例与专用实例

SQL Server可以配置为每个服务器一个实例、也可以配置为多个实例。正确的决策通常取决于多种因素、例如服务器是用于生产还是开发、实例是否被视为对业务运营和性能目标至关重要。

共享实例配置最初可能更易于配置、但可能会导致资源被划分或锁定的问题、进而导致数据库托管在共享SQL Server实例上的其他应用程序出现性能问题。

对性能问题进行故障排除可能非常复杂、因为您必须确定哪个实例是根发生原因。此问题会与操作系统许可证和SQL Server许可证的成本进行权衡。如果应用程序性能至关重要、则强烈建议使用专用实例。

Microsoft在服务器级别为每个核心(而不是每个实例)授予SQL Server许可证。因此、为了节省许可成本、数据库管理员尝试尽可能多地安装服务器可以处理的SQL Server实例、这可能会导致以后出现重大性能问题。



\*SQL NetApp建议\*尽可能选择专用的SQL Server实例，以获得最佳性能。

## tempdb文件

tempdb数据库的利用率可能很高。除了在ONTAP上以最佳方式放置用户数据库文件之外、tempdb数据文件放置对于减少分配争用也至关重要。tempdb应放置在单独的磁盘上、不应与用户数据文件共享。

如果SQL Server必须向特殊系统页面写入数据以分配新对象、则全局分配映射(GAM)、共享全局分配映射(SGAM)或页面可用空间(PFS)页面可能会发生页面争用。页锁将这些页面锁定在内存中。在繁忙的SQL Server实例上、可能需要很长时间才能在tempdb中的系统页面上获得锁定。这会导致查询运行时间变慢、并称为页锁争用。请参见以下创建tempdb数据文件的最佳实践：

- 对于<或=最多8个核心：tempdb数据文件=核心数
- 对于超过8个核心：8个tempdb数据文件
- 应创建大小相等的tempdb数据文件

以下示例脚本通过创建八个大小相等的tempdb文件并将tempdb移动到SQL Server 2012及更高版本的挂载点来修改tempdb C:\MSSQL\tempdb。

```
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
```

从SQL Server 2016开始、安装期间会自动检测操作系统可见的CPU核心数、SQL Server会根据该数量计算和配置获得最佳性能所需的tempdb文件数。

## AFF或FAS系统上的存储配置

### 概述

ONTAP存储解决方案与Microsoft SQL Server相结合、可以实现企业级数据库存储设计、从而满足当今最苛刻的应用程序要求。

要优化基于ONTAP的SQL Server解决方案、需要了解SQL Server I/O模式和特征。为SQL Server数据库精心设计的存储布局必须支持SQL Server的性能要求、同时还必须在整体上最大限度地提高基础架构的可管理性。良好的存储布局还可以成功完成初始部署、并随着业务的增长逐步平稳地扩展环境。

### 数据存储设计

对于不使用SnapCenter 执行备份的SQL Server数据库、Microsoft建议将数据和日志文件放置在不同的驱动器上。对于同时更新和请求数据的应用程序、日志文件会占用大量写入资源、而数据文件(取决于您的应用程序)会占用大量读/写资源。对于数据检索、不需要日志文件。因此、可以通过放置在其自己驱动器上的数据文件来满足数据请求。

创建新数据库时、Microsoft建议为数据和日志指定单独的驱动器。要在创建数据库后移动文件、必须使数据库脱机。有关Microsoft的更多建议、请参见 ["将数据和日志文件放在不同的驱动器上"](#)。

## 聚合

聚合是NetApp存储配置的最低级别存储容器。互联网上存在一些旧文档、建议将IO分隔到不同的底层驱动器组上。不建议在ONTAP中使用此选项。NetApp已使用数据文件和事务日志文件分开的共享和专用聚合执行了各种I/O工作负载特征化测试。测试表明、一个具有更多RAID组和驱动器的大型聚合可以优化和提高存储性能、并使管理员更易于管理、原因有两个：

- 一个大型聚合可使所有驱动器的I/O功能对所有文件可用。
- 一个大型聚合可以最高效地利用磁盘空间。

对于高可用性(HA)、请将SQL Server Always On可用性组二级同步副本放在聚合中的一个单独Storage Virtual Machine (SVM)上。为了实现灾难恢复、请将异步副本放置在灾难恢复站点中单独存储集群中的聚合上、并使用NetApp SnapMirror技术复制内容。NetApp建议在聚合中至少留出10%的可用空间、以获得最佳存储性能。

## Volumes

卷会创建并驻留在聚合中。此术语有时会引起混淆、因为ONTAP卷不是LUN。ONTAP卷是数据的管理容器。卷可以包含文件、LUN甚至S3对象。卷不会占用空间、它仅用于管理包含的数据。

### 卷设计注意事项

在创建数据库卷设计之前、请务必了解SQL Server I/O模式和特征如何根据工作负载以及备份和恢复要求而有所不同。请参见以下适用于灵活卷的NetApp建议：

- 避免在主机之间共享卷。例如、虽然可以在一个卷中创建2个LUN并将每个LUN共享到不同的主机、但应避免这种情况、因为它会使管理复杂化。如果在同一主机上运行多个SQL Server实例、则除非某个节点上的卷接近限制、否则请避免卷共享、而是为每个主机上的每个实例创建一个单独的卷、以简化数据管理。
- 使用NTFS挂载点而不是驱动器盘符、以超过Windows中26个驱动器盘符的限制。使用卷挂载点时、一般建议为卷标签指定与挂载点相同的名称。
- 适当时、配置卷自动调整策略以帮助防止出现空间不足情况。
- 如果在SMB共享上安装SQL Server、请确保在SMB卷上启用Unicode以创建文件夹。
- 将卷中的Snapshot预留值设置为零、以便于从操作角度进行监控。
- 禁用快照计划和保留策略。请改用SnapCenter来协调SQL Server数据卷的Snapshot副本。
- 将SQL Server系统数据库放在一个专用卷上。
- tempdb是一个系统数据库、SQL Server会使用它作为临时工作空间、尤其是用于I/O密集型DBCC CHECKDB操作。因此、请将此数据库放在具有一组单独磁盘轴的专用卷上。在卷数量是一项挑战的大型环境中、您可以在仔细规划后将tempdb整合到较少的卷中、并将其与其他系统数据库存储在同一个卷中。tempdb的数据保护不是高优先级、因为每次重新启动SQL Server时都会重新创建此数据库。
- 将用户数据文件(.mdf)放在不同的卷上、因为它们是随机读/写工作负载。通常、创建事务日志备份的频率比创建数据库备份的频率更高。因此、请将事务日志文件(.ldf)放在与数据文件不同的卷或VMDK上、以便可以为每个卷或VMDK创建独立的备份计划。这种分离还会将日志文件的顺序写入I/O与数据文件的随机读/写I/O隔离开来、并显著提高SQL Server性能。

## LUN

- 确保用户数据库文件和用于存储日志备份的日志目录位于不同的卷上、以防止保留策略在SnapVault技术中使用快照时覆盖这些快照。
- 不要在同一LUN上混合使用数据库文件和非数据库文件、例如与全文搜索相关的文件。
- 将数据库二级文件(作为文件组的一部分)放在不同的卷上可提高SQL Server数据库的性能。只有当数据库的文件不与任何其他文件共享其LUN .mdf`时、此隔离才有效 ` .mdf。
- 如果使用DiskManager或其他工具创建LUN、请确保在格式化LUN时将分区的分配单元大小设置为64K。
- 请参见 ["Microsoft Windows和本机MPIO、并遵循适用于现代SAN的ONTAP最佳实践"](#) 在MPIO属性中将Windows上的多路径支持应用于iSCSI设备。

## 数据库文件和文件组

在初始部署阶段、在ONTAP上正确放置SQL Server数据库文件至关重要。这样可以确保最佳性能、空间管理、备份和还原时间、并可根据您的业务需求进行配置。

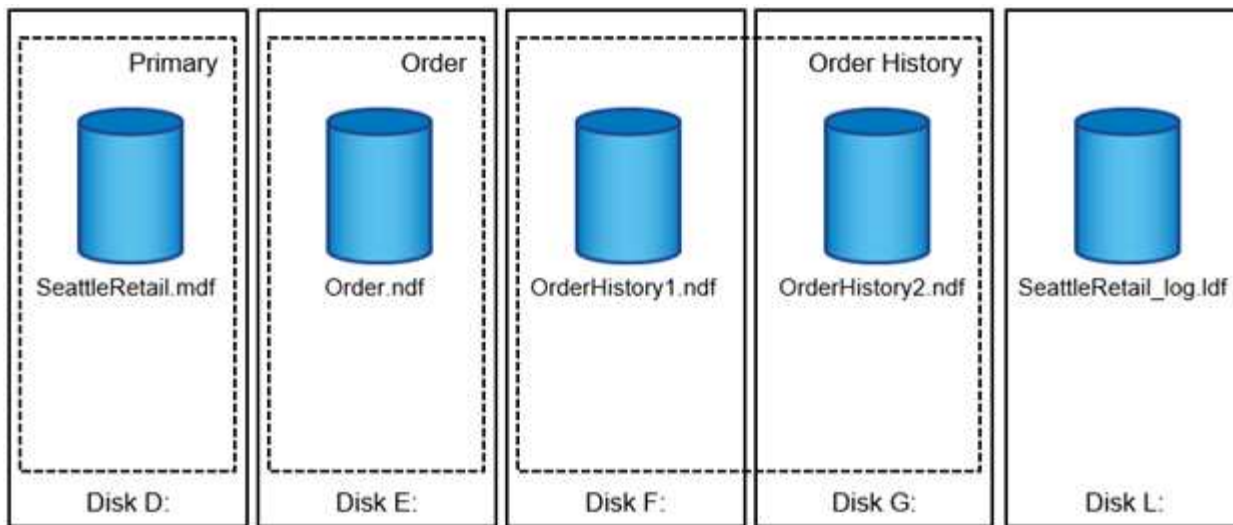
从理论上讲、SQL Server (64位)支持每个实例32、767个数据库以及524、272 TB的数据库大小、尽管典型安装通常包含多个数据库。但是、SQL Server可以处理的数据库数量取决于负载和硬件。SQL Server实例托管数十个、数百个甚至数千个小型数据库的情况并不少见。

### 数据库文件和文件组

每个数据库由一个或多个数据文件和一个或多个事务日志文件组成。事务日志存储有关数据库事务以及每个会话所做的所有数据修改的信息。每次修改数据时、SQL Server都会在事务日志中存储足够的信息、以便撤消(回滚)或重做(重放)操作。SQL Server事务日志是SQL Server在数据完整性和稳定性方面声誉的重要组成部分。事务日志对于SQL Server的运行异常、一致性、隔离和持久性(ACID)功能至关重要。一旦对数据页面进行任何更改、SQL Server就会写入事务日志。每个数据操作语言(DML)语句(例如SELECT、INSERT、UPDATE-DELETE)都是一个完整的事务、事务日志可确保执行整个基于集合的操作、从而确保事务的正常运行。

每个数据库都有一个主数据文件、默认情况下、该文件的扩展名为.mdf。此外、每个数据库都可以具有二级数据库文件。默认情况下、这些文件的扩展名为.ndf。

所有数据库文件都分组到文件组中。文件组是逻辑单元、可简化数据库管理。它们允许在逻辑对象放置和物理数据库文件之间进行隔离。创建数据库对象表时、您可以指定应将其放置在哪个文件组中、而不必担心基础数据文件配置。



通过将多个数据文件放在文件组中、您可以将负载分布在不同的存储设备上、这有助于提高系统的I/O性能。相反、事务日志不会从多个文件中受益、因为SQL Server会按顺序写入事务日志。

通过将文件组中的逻辑对象放置与物理数据库文件分隔开、您可以微调数据库文件布局、从而最大程度地利用存储子系统。支持指定工作负载的数据文件数量可以根据需要进行更改、以支持I/O要求和预期容量、而不会影响应用程序。数据库布局中的这些变化对于应用程序开发人员来说是透明的、他们将数据库对象放在文件组中、而不是数据库文件中。



\* NetApp建议\*避免将主文件组用于除系统对象之外的任何其他对象。为用户对象创建单独的文件组或一组文件组可简化数据库管理和灾难恢复、尤其是在大型数据库的情况下。

#### 数据库实例文件初始化

您可以在创建数据库或向现有数据库添加新文件时指定初始文件大小和自动增长参数。SQL Server在选择应将数据写入哪个数据文件时使用比例填充算法。它会将大量数据按比例写入文件中的可用空间。文件中的可用空间越多、它处理的写入就越多。



\* NetApp建议\*单个文件组中的所有文件都具有相同的初始大小和自动增长参数，并且增长大小以MB而不是百分比为单位定义。这有助于比例填充算法均匀地平衡数据文件之间的写入活动。

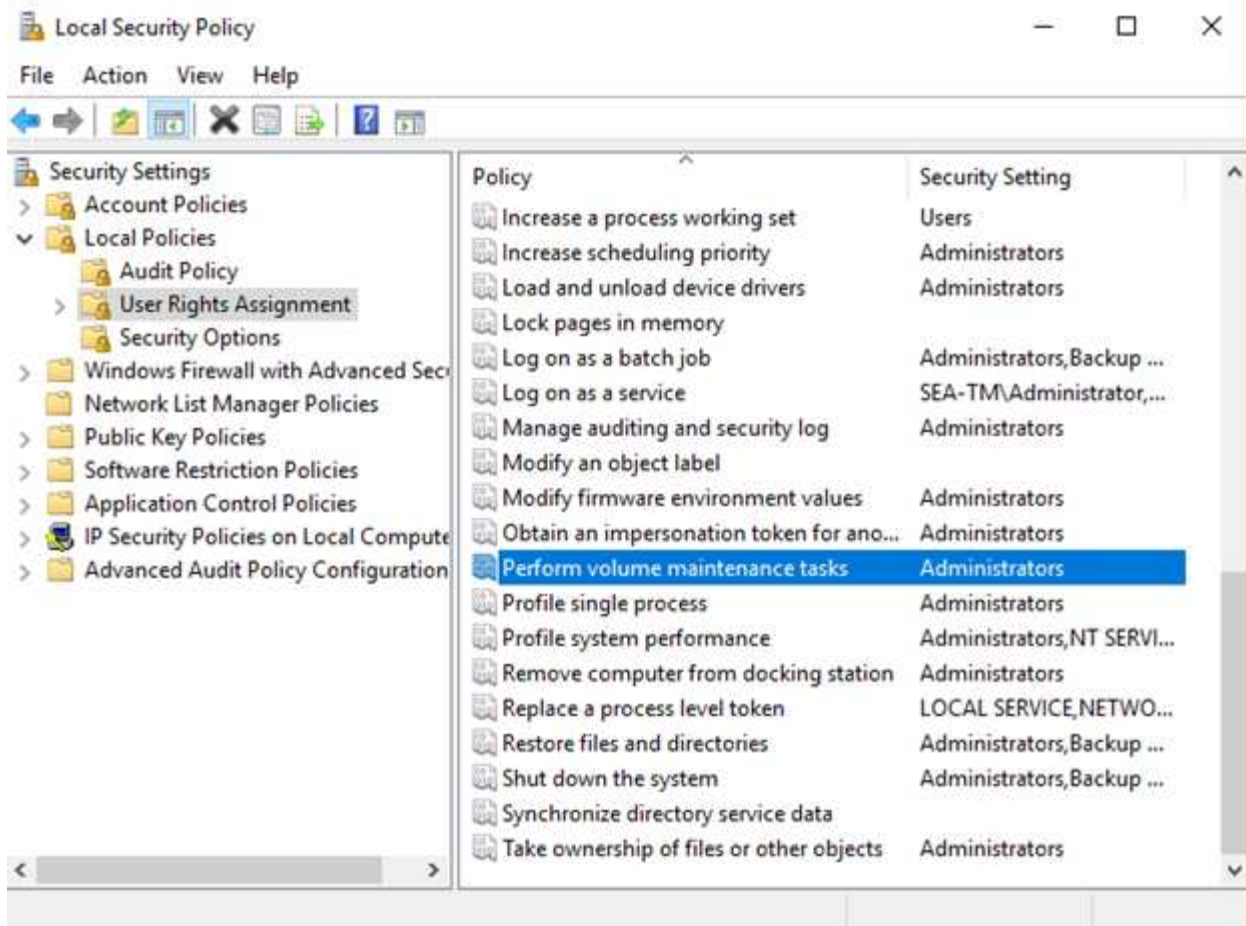
每当SQL Server增加文件大小时、它都会用零填充新分配的空间。该进程会阻止所有需要写入相应文件的会话、或者在事务日志增长时生成事务日志记录。

SQL Server始终将事务日志零、并且此行为无法更改。但是、您可以通过启用或禁用即时文件初始化来控制数据文件是否置零。启用即时文件初始化有助于加快数据文件增长速度、并减少创建或还原数据库所需的时间。

即时文件初始化会带来较小的安全风险。启用此选项后、数据文件的未分配部分可能包含先前删除的操作系统文件中的信息。数据库管理员可以检查此类数据。

您可以通过向SQL Server启动帐户添加SA\_Manage\_volume\_name权限(也称为"执行卷维护任务")来启用即时文件初始化。您可以在本地安全策略管理应用程序(secpol.msc)下执行此操作、如下图所示。打开"Perform volume Maintenance Task (执行卷维护任务)"权限的属性、并将SQL Server启动帐户添加到其中的用户列表中。





要检查权限是否已启用、您可以使用以下示例中的代码。此代码设置了两个跟踪标志、用于强制SQL Server将追加信息写入错误日志、创建小型数据库以及读取日志内容。

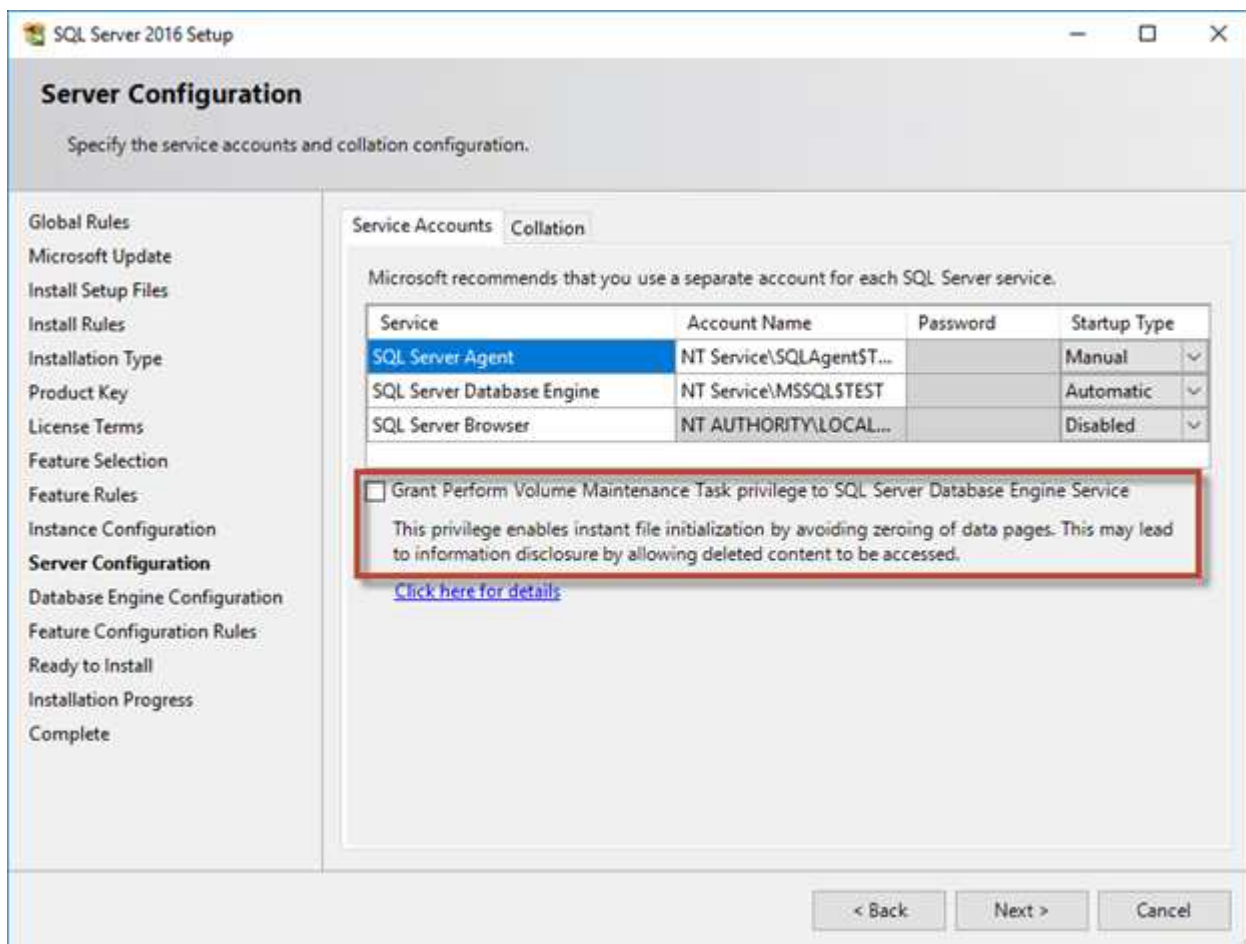
```
DBCC TRACEON(3004,3605,-1)
GO
CREATE DATABASE DelMe
GO
EXECUTE sp_readerrorlog
GO
DROP DATABASE DelMe
GO
DBCC TRACEOFF(3004,3605,-1)
GO
```

如果未启用即时文件初始化、则SQL Server错误日志会显示、除了将ldf日志文件置零之外、SQL Server还会将MDF数据文件置零、如以下示例所示。启用即时文件初始化后、它仅显示日志文件置零。



|     | 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      | FixupLog Tail(progress) zeroing C:\Program Files\Microsoft SQL Server\110\Shared\ |
| 372 | 2017-02-09 08:10:08.160 | spid53      | Zeroing C:\Program Files\Microsoft SQL Server\MSSQL\DATA\                         |
| 373 | 2017-02-09 08:10:08.170 | spid53      | Zeroing completed on C:\Program Files\Microsoft SQL Server\110\Shared\            |
| 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.  |

在SQL Server 2016中、执行卷维护任务得到了简化、稍后在安装过程中作为一个选项提供。此图显示了为SQL Server数据库引擎服务授予执行卷维护任务权限的选项。



控制数据库文件大小的另一个重要数据库选项是自动缩放。启用此选项后、SQL Server会定期缩减数据库文件、减小其大小并向操作系统释放空间。此操作会占用大量资源、并且很少有用、因为新数据进入系统一段时间后、数据库文件会再次增长。不应在数据库上启用自动缩放。

## 日志目录

日志目录在SQL Server中指定、用于在主机级别存储事务日志备份数据。如果使用SnapCenter备份日志文件、则SnapCenter使用的每个SQL Server主机都必须配置一个主机日志目录、才能执行日志备份。SnapCenter 具

有一个数据库存储库、因此与备份、还原或克隆操作相关的元数据存储在一个中央数据库存储库中。

主机日志目录的大小计算如下：

主机日志目录大小=((最大数据库LDF大小x每日日志更改率%) x (快照保留))÷(1 - LUN开销空间%)

主机日志目录规模估算公式假定LUN开销空间为10%

将日志目录放在专用卷或LUN上。主机日志目录中的数据量取决于备份的大小以及备份的保留天数。SnapCenter只允许每个SQL Server主机有一个主机日志目录。您可以在SnapCenter → Host → Configure Plug-in中配置主机日志目录。



- NetApp建议\*对主机日志目录执行以下操作：
- 确保主机日志目录不会被任何可能损坏备份快照数据的其他类型的数据共享。
- 请勿将用户数据库或系统数据库放置在托管挂载点的LUN上。
- 在SnapCenter将事务日志复制到的专用卷上创建主机日志目录。
- 使用SnapCenter向导将数据库迁移到NetApp存储、以便数据库存储在有效位置、从而成功执行SnapCenter备份和还原操作。请注意、迁移过程会造成系统中断、并且在迁移过程中、发生原因会使数据库脱机。
- 对于SQL Server的故障转移集群实例(FCIS)、必须满足以下条件：
  - 如果您使用的是故障转移集群实例、则主机日志目录LUN必须是与要备份的SQL Server实例位于同一集群组中的集群磁盘资源SnapCenter。
  - 如果您使用的是故障转移集群实例、则必须将用户数据库置于共享LUN上、这些共享LUN是分配给与SQL Server实例关联的集群组的物理磁盘集群资源。

## 存储效率

ONTAP存储效率已进行优化、可存储和管理SQL Server数据、所用存储空间量最低、不会对性能产生任何影响。

数据压缩、数据缩减和重复数据删除等空间效率功能旨在增加给定物理存储量所需的逻辑数据量。这样可以降低成本和管理开销。

从较高层面来看、数据压缩是一个数学过程、通过该过程、可以检测数据模式并对其进行编码、从而减少空间需求。相反、重复数据删除会检测实际重复的数据块并删除无关的副本。数据缩减允许多个逻辑数据块共享介质上的同一物理块。



有关存储效率与预留百分比之间交互的说明、请参见以下有关精简配置的章节。

## 压缩

在全闪存存储系统推出之前、基于阵列的数据压缩的价值有限、因为大多数I/O密集型工作负载都需要大量磁盘轴才能提供可接受的性能。由于驱动器数量众多、存储系统所含容量总是远远超出所需容量。随着固态存储的兴起、这种情况发生了变化。不再需要纯粹为了获得良好的性能而大量过度配置驱动器。存储系统中的驱动器空间可以与实际容量需求相匹配。

与旋转驱动器相比、固态驱动器(SSD)的IOPS功能提高几乎始终可以节省成本、但数据压缩可以通过增加固态介质的有效容量来进一步节省成本。

数据压缩方法有多种。许多数据库都具有自己的数据压缩功能、但在客户环境中很少出现这种情况。原因通常是对压缩数据\*进行更改\*会对性能造成影响、此外、对于某些应用程序、数据库级数据压缩的许可成本较高。最后、还会对数据库操作产生整体性能影响。为执行数据压缩和解压缩的CPU支付较高的每CPU许可证成本毫无意义、而不是实际的数据库工作。更好的选择是将压缩工作负载分流到存储系统。

#### 自适应数据压缩

自适应数据压缩已针对企业级工作负载进行了全面测试、未观察到对性能的影响、即使在延迟以微秒为单位的闪存环境中也是如此。一些客户甚至报告说、使用数据压缩后性能会提高、因为数据会在缓存中保持压缩状态、从而有效地增加了控制器中的可用缓存量。

ONTAP以4 KB为单位管理物理块。自适应数据压缩使用默认的压缩块大小8 KB、这意味着数据以8 KB单位进行压缩。这与关系数据库最常使用的8 KB块大小匹配。随着将更多数据作为一个单元进行压缩、数据压缩算法的效率也会提高。32 KB压缩块大小比8 KB压缩块单元更节省空间。这确实意味着、使用默认8 KB块大小的自适应数据压缩确实会使效率略低、但使用更小的数据压缩块大小也会有显著优势。数据库工作负载包含大量覆盖活动。要覆盖经过压缩的32 KB数据块中的8 KB、需要回读整个32 KB逻辑数据、对其进行解压缩、更新所需的8 KB区域、重新压缩、然后将整个32 KB写入驱动器。这对存储系统来说是一项非常昂贵的操作、因此、某些基于较大压缩块大小的竞争存储阵列也会对数据库工作负载的性能造成严重影响。



自适应数据压缩使用的块大小最多可以增加至32 KB。这可能会提高存储效率、对于事务日志和备份文件等不活动的文件、如果阵列上存储了大量此类数据、则应考虑使用此方法。在某些情况下、使用16 KB或32 KB块大小的活动数据库也可以通过增加要匹配的自适应数据压缩的块大小来受益。请咨询NetApp或合作伙伴代表、了解这是否适合您的工作负载。



在流式备份目标上、不应同时使用大于8 KB的数据压缩块大小和重复数据删除。原因是、对备份的数据所做的微小更改会影响32 KB数据压缩窗口。如果窗口发生变化、则生成的压缩数据会在整个文件中有所不同。重复数据删除在数据压缩后进行、这意味着重复数据删除引擎对每个压缩备份的看法不同。如果需要对流式备份进行重复数据删除、则只应使用8 KB块自适应数据压缩。最好使用自适应数据压缩、因为它的块大小较小、不会影响重复数据删除的效率。出于类似的原因、主机端压缩也会影响重复数据删除效率。

#### 数据压缩对齐

数据库环境中的自适应数据压缩需要在一定程度上考虑数据压缩块对齐问题。对于随机覆盖非常特定的块的数据来说、这样做只是一个问题。这种方法在概念上类似于整体文件系统对齐、即文件系统的起点必须与4 k设备边界对齐、文件系统的块大小必须是4 k的倍数。

例如、只有当8 KB写入文件与文件系统本身内的8 KB边界对齐时、才会对其进行压缩。这一点意味着它必须位于文件的前8 KB、文件的后8 KB、依此类推。要确保正确对齐、最简单的方法是使用正确的LUN类型、创建的任何分区都应设备起始位置偏移8K的倍数、并使用数据库块大小的倍数作为文件系统块大小。

备份或事务日志等数据是跨多个块按顺序写入的操作、所有这些块都会进行压缩。因此、无需考虑对齐。唯一关注的I/O模式是随机覆盖文件。

#### 数据缩减

数据缩减是一项可提高数据压缩效率的技术。如前文所述、自适应数据压缩本身最多可节省2:1的空间、因为它仅限于在4 KB WAFL块中存储8 KB I/O。块大小越大、压缩方法的效率越高。但是、它们不适用于受到小块覆盖的数据。解压缩32 KB数据单元、更新8 KB部分、重新压缩以及回写驱动器会产生开销。

数据缩减的工作原理是、允许将多个逻辑块存储在物理块中。例如、具有高度可压缩数据(例如文本或部分全满块)的数据库可以从8 KB压缩到1 KB。如果不进行数据缩减、这1 KB的数据仍会占用整个4 KB块。实时数据缩减

允许将1 KB的压缩数据与其他压缩数据一起存储在仅1 KB的物理空间中。它不是一种压缩技术；它只是一种在驱动器上分配空间的更高效的方式、因此不会产生任何可检测的性能影响。

节省的资金数额各不相同。已压缩或加密的数据通常无法进一步压缩、因此、数据集无法从数据缩减中受益。相比之下、新初始化的数据文件包含的块元数据和零数据略多、数据压缩率高达80：1。

#### 对温度敏感的存储效率

温度敏感型存储效率(TSSE)可在ONTAP 9.8及更高版本中使用。它依靠块访问热图来识别不常访问的块、并以更高的效率对其进行压缩。

#### 重复数据删除

重复数据删除是指从数据集中删除重复的块大小。例如、如果10个不同文件中存在相同的4 KB块、则重复数据删除会将所有10个文件中的4 KB块重定向到相同的4 KB物理块。结果是、这些数据的效率将提高10：1。

VMware子系统启动LUN等数据的重复数据删除效果通常非常好、因为它们包含同一操作系统文件的多个副本。我们观察到的效率为100：1甚至更高。

某些数据不包含重复数据。例如、Oracle块包含数据库全局唯一的标头和几乎唯一的尾部。因此、对Oracle数据库进行重复数据删除很少能节省超过1%的空间。对MS SQL数据库执行重复数据删除略有改进、但块级别的唯一元数据仍是一个限制。

在少数情况下、使用16 KB和大型块的数据库可节省多达15%的空间。每个块的初始4 KB包含全局唯一标头、而最后4 KB块包含接近唯一的尾部。内部块是重复数据删除的候选数据、但实际上、这几乎完全是由于对置零数据进行重复数据删除。

许多争用资源的阵列都声称可以根据数据库被复制多次的假设对数据库进行重复数据删除。在这方面、也可以使用NetApp重复数据删除、但ONTAP提供了一个更好的选择：NetApp FlexClone技术。最终结果是相同的；系统会为一个数据库创建多个副本、这些副本共享大多数底层物理块。与花时间复制数据库文件并对其进行重复数据删除相比、使用FlexClone的效率要高得多。实际上、它是无重复数据删除、而不是重复数据删除、因为从一开始就不会创建重复数据。

#### 效率和精简配置

效率功能是精简配置的一种形式。例如、占用100 GB卷的100 GB LUN可能会压缩到50 GB。由于卷仍为100 GB、因此尚未实现实际节省。必须先减小卷大小、以便节省的空间可用于系统上的其他位置。如果稍后更改100 GB LUN会导致数据的可压缩性降低、则LUN大小会增大、卷可能会填满。

强烈建议使用精简配置、因为它可以简化管理、同时显著提高可用容量并节省相关成本。原因很简单—数据库环境通常包含大量空闲空间、大量卷和LUN以及可压缩数据。厚配置会为卷和LUN预留存储空间、以防它们最终达到100%全满并包含100%不可压缩数据。这种情况不大可能发生。通过精简配置、可以回收这些空间并将其用于其他位置、并可以基于存储系统本身进行容量管理、而不是基于许多较小的卷和LUN。

有些客户更喜欢对特定工作负载使用厚配置、或者通常根据既定的运营和采购实践使用厚配置。



如果卷配置为厚配置、则必须注意完全禁用该卷的所有效率功能、包括使用命令解压缩和删除重复数据删除 `sis undo`。此卷不应显示在输出中 `volume efficiency show`。如果配置了效率功能、则仍会为卷部分配置效率功能。因此、覆盖保证的工作方式有所不同、这会增加配置疏忽导致卷意外用尽空间的可能性、从而导致数据库I/O错误。



## 效率最佳实践

- NetApp建议\*：

### AFF默认值

在纯闪存AFF系统上运行的ONTAP上创建的卷经过精简配置、并启用了所有实时效率功能。尽管数据库通常不会从重复数据删除中受益、并且可能包含不可压缩的数据、但默认设置适用于几乎所有工作负载。ONTAP旨在高效处理所有类型的数据和I/O模式、无论它们是否可节省空间。只有在完全了解原因且有优势的情况下、才应更改默认值。

### 一般建议

- 如果卷和(或) LUN未进行精简配置、则必须禁用所有效率设置、因为使用这些功能不会节省空间、并且将厚配置与空间效率结合使用可能会导致意外行为、包括空间不足错误。
- 如果数据不会被覆盖(例如使用备份或数据库事务日志)、则可以通过在较低的冷却期启用TSSE来提高效率。
- 某些文件可能包含大量不可压缩数据、例如、在应用程序级别已启用数据压缩时、文件已加密。如果出现上述任一情况、请考虑禁用数据压缩、以便在包含可压缩数据的其他卷上执行更高效的操作。
- 不要在数据库备份中同时使用32 KB数据压缩和重复数据删除。请参见一节 [\[自适应数据压缩\]](#) 了解详细信息。

### 数据库压缩

SQL Server本身还具有压缩和高效管理数据的功能。SQL Server目前支持两种类型的数据压缩：行压缩和页面压缩。

行压缩会更改数据存储格式。例如、它会将整数和小数更改为可变长度格式、而不是其本机固定长度格式。此外、它还会通过消除空格将固定长度字符串更改为可变长度格式。页面压缩可实现行压缩和另外两种压缩策略(前缀压缩和词典压缩)。有关页面压缩的详细信息、请参见 "[页面压缩实施](#)"。

SQL Server 2008及更高版本的企业版、开发人员版和评估版目前支持数据压缩。虽然数据压缩可以由数据库本身执行、但在SQL Server环境中很少会出现这种情况。

下面是管理SQL Server数据文件空间的建议

- 在SQL Server环境中使用精简配置可提高空间利用率、并在使用空间保证功能时降低整体存储需求。
  - 对最常见的部署配置使用自动增长、因为存储管理员只需要监控聚合中的空间使用量。
- 请勿在FAS中包含SQL Server数据文件的任何卷上启用重复数据删除、除非已知该卷包含同一数据的多个副本、例如、将数据库从备份还原到单个卷。

### 空间回收

可以定期启动空间回收、以恢复LUN中未使用的空间。对于SnapCenter、您可以使用以下PowerShell命令启动空间回收。

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

如果需要运行空间回收、则应在活动较少的时段运行此过程、因为它最初会占用主机上的周期。

## 数据保护

数据库备份策略应基于已确定的业务要求、而不是理论能力。通过将ONTAP的Snapshot技术与Microsoft SQL Server API相结合、无论用户数据库大小如何、您都可以快速创建应用程序一致的备份。为了满足更高级或横向扩展的数据管理要求、NetApp提供了SnapCenter。

### SnapCenter

SnapCenter是一款适用于企业级应用程序的NetApp数据保护软件。使用适用于SQL Server的SnapCenter插件以及适用于Microsoft Windows的SnapCenter插件管理的操作系统、可以快速轻松地保护SQL Server数据库。

SQL Server实例可以是独立设置、故障转移集群实例、也可以始终位于可用性组上。这样、可以通过单一管理平台从主副本或二级副本保护、克隆和还原数据库。SnapCenter可以管理内部、云端和混合配置中的SQL Server数据库。此外、还可以在原始主机或备用主机上几分钟内创建数据库副本、以供开发或报告之用。

SQL Server还要求操作系统和存储之间进行协调、以确保创建快照时Snapshot中存在正确的数据。在大多数情况下、执行此操作的唯一安全方法是使用SnapCenter或T-SQL。如果未进行这种额外协调、则创建的快照可能无法可靠地恢复。

有关适用于SnapCenter的SQL Server插件的详细信息、请参见 ["TR-4714：《使用NetApp SnapCenter的SQL Server最佳实践指南》"](#)。

### 使用T-SQL快照保护数据库

在SQL Server 2022中、Microsoft引入了T-SQL快照、可通过它编写脚本并自动执行备份操作。您可以为快照准备数据库、而不是执行全尺寸副本。数据库备份就绪后、您可以利用ONTAP REST API创建快照。

以下是备份工作流示例：

1. 使用alter命令冻结数据库。这样可以使数据库在底层存储上为一致的快照做好准备。冻结后、您可以解冻数据库并使用backup命令记录快照。
2. 使用新的backup group和backup server命令同时对存储卷上的多个数据库执行快照。
3. 执行完整备份或copy\_only完整备份。这些备份也会记录在msdb中。
4. 在快照完整备份之后、使用通过正常流式传输方法创建的日志备份执行时间点恢复。如果需要、还支持流式差异备份。

要了解更多信息，请参见 ["了解T-SQL快照的Microsoft文档"](#)。



\* NetApp建议\*使用SnapCenter创建Snapshot副本。上述T-SQL方法同样有效、但SnapCenter可实现备份、还原和克隆过程的完全自动化。同时、它还会执行发现以确保创建的快照正确无误。无需预配置。

### 使用SnapCenter的SQL Server可用性组

SnapCenter支持备份使用Windows故障转移集群配置的SQL Server可用性组数据库。

适用于Microsoft SQL Server的SnapCenter插件必须安装在Windows服务器故障转移集群的所有节点上。有关设置SnapCenter插件的前提条件和步骤、请参见["文档"](#)。

SnapCenter会发现Windows主机中的所有数据库、实例和可用性组、SnapCenter资源页面上会枚举这些主机和资源。

## 保护始终可用性组中的数据库

可用性组中的数据库可以通过多种方式进行保护。

- 数据库级备份：为数据库资源页面选择可用性数据库、添加包含完整备份/日志备份的策略、并计划备份。无论数据库角色是主副本还是二级副本、SnapCenter都会执行备份。也可以通过将数据库添加到资源组来配置保护。
- 实例级备份：选择实例、实例上运行的所有数据库将根据选定策略受到保护。所有数据库(包括作为主副本或二级副本运行的可用性数据库)都会使用SnapCenter进行备份。也可以通过将实例添加到资源组来配置此保护。
- 可用性组级别备份：配置策略时、SnapCenter提供了一个用于可用性组级别备份的高级选项。策略中的可用性组设置允许用户选择要备份的副本首选项。您可以选择主副本、二级副本或全部副本。默认选项基于SQL Server可用性组配置中的备份副本集。

仅当可用性组级别备份用于保护可用性组数据库而不适用于数据库或实例级别备份时、SnapCenter策略中的可用性组设置才会适用。



\* NetApp建议\*使用可用性级别备份来备份在NetApp ONTAP存储上运行的所有副本。

## 在SnapCenter中配置日志备份

如果在独立SQL Server设置中设置了可用性组、则必须在Windows服务器故障转移集群的每个节点上挂载一个专用磁盘。应使用专用磁盘配置日志目录以保存事务日志备份。

如果在SQL Server故障转移集群上设置了可用性组、则应在SQL Server故障转移集群实例上创建集群磁盘、以托管日志目录。

## 使用SnapCenter在可用性组设置中还原数据库

- SnapCenter提供了reeseed选项、可从二级副本上提供的最新快照自动恢复数据库。重新密封操作将自动还原数据库备份并将其加入可用性组。
- 在可用性组中还原副本数据库的另一种方法是、中断可用性组并执行完整的完整还原和日志还原。使用SnapCenter在恢复模式下还原数据库、然后使用SQL Server Management Studio或T-SQL将数据库重新加入可用性组。
- 要仅恢复部分数据、可以使用SnapCenter的克隆功能创建数据库的克隆副本。使用SnapCenter在几分钟内创建数据库副本、然后使用SQL Server本机工具将数据导出到主副本。

有关设置数据库存储布局以满足RTO和RPO要求的最佳实践，请参见["TR-4714 《使用NetApp SnapCenter的Microsoft SQL Server最佳实践》"](#)。



SnapCenter不支持分布式可用性组和包含的可用性组。

## 灾难恢复



企业数据库和应用程序基础架构通常需要进行复制、以防止发生自然灾害或意外业务中断、同时最大限度地减少停机时间。

SQL Server无中断可用性组复制功能是一个绝佳的选择、NetApp提供了将数据保护与无中断集成的选项。但是、在某些情况下、您可能需要考虑使用ONTAP复制技术。有三个基本选项。

### SnapMirror

SnapMirror技术为通过LAN和广域网复制数据提供了快速灵活的企业解决方案。SnapMirror技术仅在创建初始镜像后将更改后的数据块传输到目标、从而显著降低网络带宽要求。它可以配置为同步或异步模式。

### NetApp MetroCluster和SnapMirror活动同步

对于许多客户来说、灾难恢复不仅需要拥有远程数据副本、还需要能够快速利用这些数据。NetApp提供了两种技术来满足这一需求—MetroCluster和SnapMirror主动同步

MetroCluster是指硬件配置中的ONTAP、其中包括低级同步镜像存储和许多附加功能。MetroCluster等集成解决方案简化了当今复杂的横向扩展数据库、应用程序和虚拟化基础架构。它将多个外部数据保护产品和策略替换为一个简单的中央存储阵列。此外、它还可以在一个集群模式存储系统中提供集成的备份、恢复、灾难恢复和高可用性(HA)功能。

SnapMirror活动同步基于SnapMirror同步。通过MetroCluster、每个ONTAP控制器都负责将其驱动器数据复制到远程位置。使用SnapMirror主动同步时、您实际上拥有两个不同的ONTAP系统、它们会维护LUN数据的独立副本、但会相互协作、为该LUN提供一个实例。从主机角度来看、它是一个LUN实体。

### SM-AS和MCC比较

SM-AS和MetroCluster在整体功能上相似、但在实施RPO = 0复制的方式及其管理方式上存在重要差异。SnapMirror异步和同步也可用作灾难恢复计划的一部分、但它们不是作为HA回配技术而设计的。

- MetroCluster配置更像是一个集成集群、其中的节点分布在各个站点之间。SM-AS的行为类似于两个其他方面独立的集群、它们合作提供选定的RPO = 0同步复制的LUN。
- 在任何给定时间、只能从一个特定站点访问MetroCluster配置中的数据。另一个数据副本位于另一个站点上、但数据是被动的。如果没有存储系统故障转移、则无法访问它。
- MetroCluster和SM-AS执行镜像在不同级别进行。MetroCluster镜像在RAID层执行。使用SyncMirror以镜像格式存储低级别的数据。在LUN、卷和协议层、镜像的使用实际上是不可见的。
- 相反、SM-AS镜像发生在协议层。这两个集群总体上是独立的集群。两个数据副本同步后、这两个集群只需镜像写入即可。在一个集群上进行写入时、该写入会复制到另一个集群。只有在两个站点上的写入均已完成时、才会向主机确认写入。除了此协议拆分行为之外、这两个集群在其他方面都是正常的ONTAP集群。
- MetroCluster的主要角色是大规模复制。您可以复制RPO为0且RTO接近零的整个阵列。这样可以简化故障转移过程、因为故障转移只需执行一项"操作"、而且在容量和IOPS方面扩展得非常好。
- SM-AS的一个关键用例是粒度复制。有时、您不希望将所有数据作为一个单元进行复制、或者您需要能够有选择地对某些工作负载进行故障转移。
- SM-AS的另一个主要用例是主动-主动操作、您希望在位于两个不同位置的两个不同集群上提供完全可用的数据副本、这些集群具有相同的性能特征、如果需要、也不需要站点间延伸SAN。您的应用程序可以同时在这两个站点上运行、这样可以减少故障转移操作期间的整体恢复时间。

## SnapMirror

下面是有关适用于SQL Server的SnapMirror的建议：

- 如果使用SMB、则目标SVM必须是源SVM所属的同一Active Directory域的成员、以便在灾难恢复期间不会破坏NAS文件中存储的访问控制列表(ACL)。
- 使用与源卷名称相同的目标卷名称并不是必需的、但这样可以简化将目标卷挂载到目标的过程、使其易于管理。如果使用SMB、则必须使目标NAS命名空间与源命名空间的路径和目录结构完全相同。
- 为保持一致、请勿从控制器计划SnapMirror更新。而是应启用SnapCenter中的SnapMirror更新、以便在完整备份或日志备份完成后更新SnapMirror。
- 将包含SQL Server数据的卷分布在集群中的不同节点上、以允许所有集群节点共享SnapMirror复制活动。此分布可优化节点资源的使用。
- 如果对快速数据恢复的需求较高、则使用同步复制；如果需要异步解决方案、则可以灵活地执行RPO。

有关SnapMirror的详细信息、请参见 ["TR-4015：《适用于ONTAP 9的SnapMirror配置和最佳实践指南》"](#)。

## MetroCluster

### 架构

在MetroCluster环境中部署Microsoft SQL Server需要对MetroCluster系统的物理设计进行一些说明。

MetroCluster会在位于不同位置或故障域的两个ONTAP集群之间同步镜像数据和配置。MetroCluster通过自动管理两个目标为应用程序提供持续可用的存储：

- 通过同步镜像写入集群的数据实现零恢复点目标(RPO)。
- 通过镜像配置和自动访问第二个站点的数据、实现近乎为零的恢复时间目标(Recovery Time目标、RTO)。

MetroCluster通过在两个站点中的两个独立集群之间自动镜像数据和配置来简化操作。由于存储是在一个集群中配置的，因此它会自动镜像到第二个站点的第二个集群。NetApp SyncMirror®提供所有数据的完整副本，并且没有RPO。这意味着、一个站点的工作负载可以随时切换到另一个站点、并继续提供数据、而不会丢失数据。MetroCluster负责管理切换过程、以便能够访问第二个站点上NAS和SAN配置的数据。将MetroCluster设计为经验证的解决方案、其中包括规模估算和配置、可在协议超时期限内或更短时间(通常少于120秒)执行切换。这样、RPO几乎为零、应用程序可以继续访问数据、而不会发生故障。MetroCluster可通过后端存储网络结构定义的多种变体提供。

### MetroCluster可用于3种不同的配置

- 具有IP连接的HA对
- 具有FC连接的HA对
- 具有FC连接的单个控制器



术语"连接"是指用于跨站点复制的集群连接。它不是指主机协议。无论用于集群间通信的连接类型如何、MetroCluster配置均支持所有主机端协议。

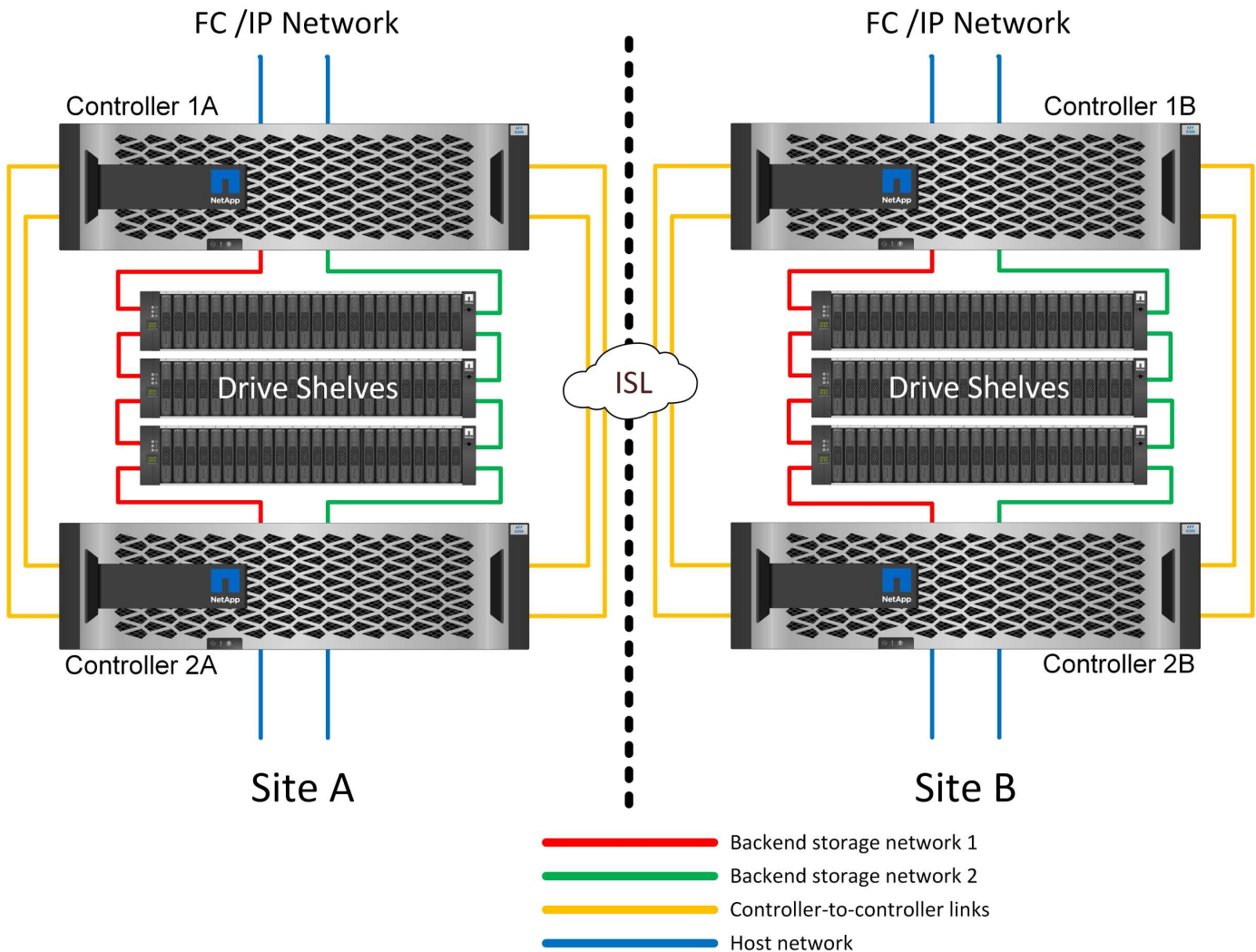
## MetroCluster IP

HA对MetroCluster IP配置会在每个站点上使用两个或四个节点。与双节点选项相比、此配置选项会增加复杂性和成本、但它具有一个重要优势：站点内冗余。简单的控制器故障不需要通过WAN访问数据。数据访问仍通过备用本地控制器保持在本地。

大多数客户选择IP连接是因为基础架构要求更简单。过去、使用暗光纤和FC交换机配置高速跨站点连接通常比较容易、但如今、高速、低延迟IP电路更容易获得。

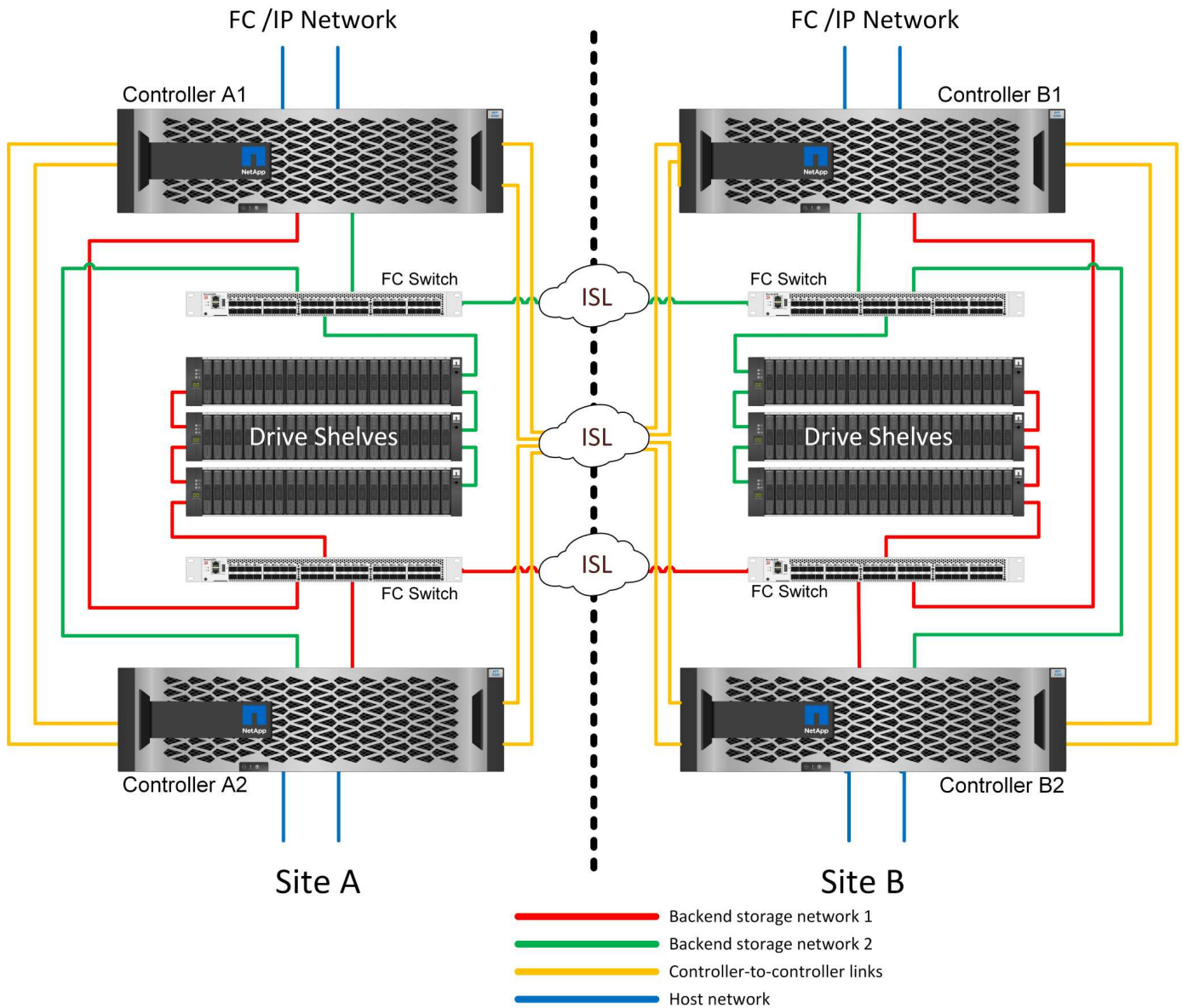
此外、该架构也更加简单、因为只有跨站点连接用于控制器。在FC SAN连接的MetroCluster中、控制器会直接写入另一站点上的驱动器、因此需要更多的SAN连接、交换机和网桥。相反、IP配置中的控制器会通过控制器写入相对的驱动器。

对于追加信息、请参阅ONTAP官方文档和 ["MetroCluster IP 解决方案架构和设计"](#)。



## HA对FC SAN连接的MetroCluster

HA对MetroCluster FC配置会在每个站点上使用两个或四个节点。与双节点选项相比、此配置选项会增加复杂性和成本、但它具有一个重要优势：站点内冗余。简单的控制器故障不需要通过WAN访问数据。数据访问仍通过备用本地控制器保持在本地。



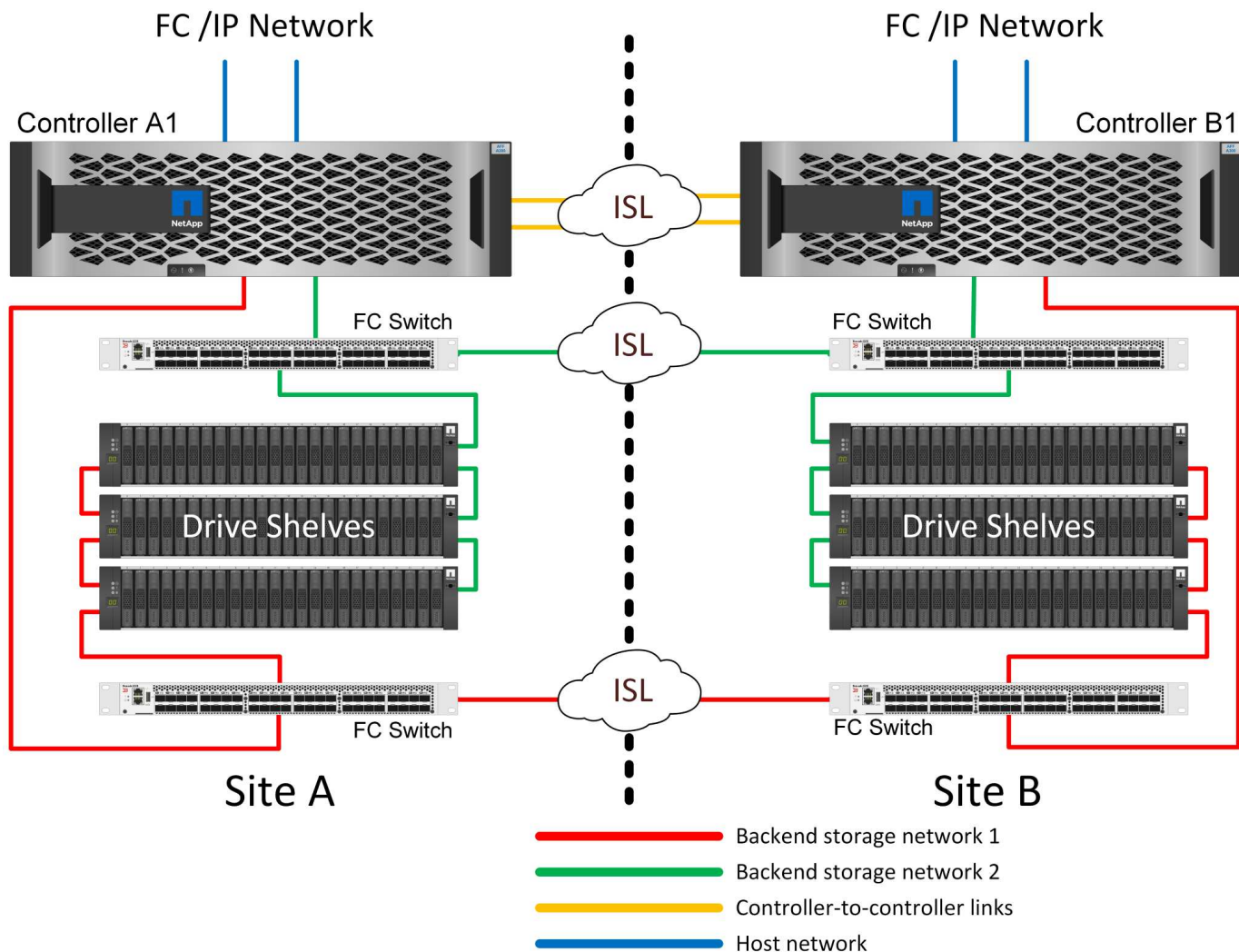
某些多站点基础架构不是为主动-主动操作而设计的、而是更多地用作主站点和灾难恢复站点。在这种情况下、通常最好使用HA对MetroCluster选项、原因如下：

- 尽管双节点MetroCluster集群是一个HA系统、但控制器意外故障或计划内维护要求数据服务必须在相反站点联机。如果站点之间的网络连接无法支持所需的带宽、则性能会受到影响。唯一的选择是同时将各种主机操作系统和相关服务故障转移到备用站点。HA对MetroCluster集群可消除此问题、因为丢失控制器会导致在同一站点内进行简单的故障转移。
- 某些网络拓扑不是为跨站点访问而设计的、而是使用不同的子网或隔离的FC SAN。在这些情况下、双节点MetroCluster集群将不再充当HA系统、因为备用控制器无法向对面站点上的服务器提供数据。要提供完全冗余、需要使用高可用性对MetroCluster选项。
- 如果将双站点基础架构视为一个高可用性基础架构、则适合使用双节点MetroCluster配置。但是、如果系统在站点发生故障后必须长时间运行、则首选HA对、因为它会继续在单个站点中提供HA。

### 双节点FC SAN连接MetroCluster

双节点MetroCluster配置仅为每个站点使用一个节点。这种设计比HA对选项更简单、因为需要配置和维护的组件更少。此外、它还降低了布线和FC交换方面的基础架构需求。最后、它还可以降低成本。





这种设计的明显影响是、单个站点上的控制器故障意味着数据可以从另一个站点访问。这种限制不一定是问题。许多企业都拥有多站点数据中心运营、并采用延伸型高速低延迟网络、这些网络本质上充当一个基础架构。在这些情况下、首选配置是双节点版本的MetroCluster。目前、多家服务提供商以PB级的规模使用双节点系统。

### MetroCluster故障恢复能力功能

MetroCluster 解决方案 中没有单点故障：

- 每个控制器都有两条通往本地站点上的驱动器架的独立路径。
- 每个控制器都有两条通往远程站点上驱动器架的独立路径。
- 每个控制器都有两条独立的路径连接到另一站点上的控制器。
- 在HA对配置中、每个控制器都有两个指向其本地配对节点的路径。

总之、可以删除配置中的任何一个组件、而不会影响MetroCluster提供数据的能力。这两个选项在故障恢复能力方面的唯一区别是、发生站点故障后、HA对版本仍然是整体HA存储系统。

### SyncMirror

使用MetroCluster保护SQL Server以SyncMirror为基础、该技术可提供最高性能的横向扩展同步镜像技术。

## 利用SyncMirror实现数据保护

最简单的一个层面是、同步复制意味着、在确认镜像存储之前、必须对镜像存储的两端进行任何更改。例如、如果数据库正在写入日志、或者VMware子系统正在修补、则写入操作绝不能丢失。作为协议级别、在将写入提交到两个站点上的非易失性介质之前、存储系统不得确认写入。只有这样、才能安全地继续操作、而不会丢失数据。

使用同步复制技术是设计和管理同步复制解决方案的第一步。最重要的注意事项是了解在各种计划内和计划外故障情形下可能发生的情况。并非所有同步复制解决方案都能提供相同的功能。如果您需要的解决方案能够实现零恢复点目标(RPO)、即零数据丢失、则必须考虑所有故障情形。特别是、如果由于站点间连接断开而无法进行复制、则会产生什么预期结果？

### SyncMirror数据可用性

MetroCluster复制基于NetApp SyncMirror技术、该技术旨在高效地切换至同步模式和切换至同步模式之外。此功能可满足需要同步复制、但也需要数据服务高可用性的客户的要求。例如、如果与远程站点的连接断开、则通常最好让存储系统继续在未复制的状态下运行。

许多同步复制解决方案只能在同步模式下运行。这种类型的全或全不复制有时称为Domino模式。此类存储系统将停止提供数据、而不是允许本地和远程数据副本处于不同步状态。如果强制中断复制、重新同步可能会非常耗时、并且可能会使客户在重新建立镜像期间完全丢失数据。

SyncMirror不仅可以在无法访问远程站点时无缝切换出同步模式、还可以在恢复连接后快速重新同步到RPO = 0状态。远程站点上的陈旧数据副本也可以在重新同步期间保留在可用状态、从而确保本地和远程数据副本始终存在。

如果需要Domino模式、则NetApp提供SnapMirror同步(SM-S)。此外、还存在应用程序级选项、例如Oracle DataGuard或SQL Server Always On可用性组。可以选择操作系统级磁盘镜像。有关追加信息和选项、请咨询您的NetApp或合作伙伴客户团队。

### SQL Server与MetroCluster

保护零RPO SQL Server数据库的一个选项是MetroCluster。MetroCluster是一种简单的高性能RPO = 0复制技术、支持您在站点间轻松复制整个基础架构。

SQL Server可在一个MetroCluster系统上扩展到多达数千个数据库。可能存在SQL Server独立实例或故障转移集群实例、MetroCluster系统不一定会添加或更改数据库管理的任何最佳实践。

本文档不会对MetroCluster进行完整说明、但其原则非常简单。MetroCluster可提供具有快速故障转移功能的RPO = 0复制解决方案。在此基础之上构建的内容取决于您的要求。

例如、站点突然丢失后的基本快速灾难恢复过程可以使用以下基本步骤：

- 强制执行MetroCluster切换
- 发现FC/iSCSI LUN (仅限SAN)
- 挂载文件系统
- 启动SQL服务

此方法的主要要求是在远程站点上运行操作系统。它必须预配置SQL Server设置、并应使用等效的内部版本进行更新。SQL Server系统数据库也可以镜像到远程站点、并在发生灾难时进行挂载。

如果在切换之前灾难恢复站点未使用托管虚拟化数据库的卷、文件系统和数据存储库、则无需在关联的卷上设置 `dr-force-nvfailo`。

## SnapMirror活动同步

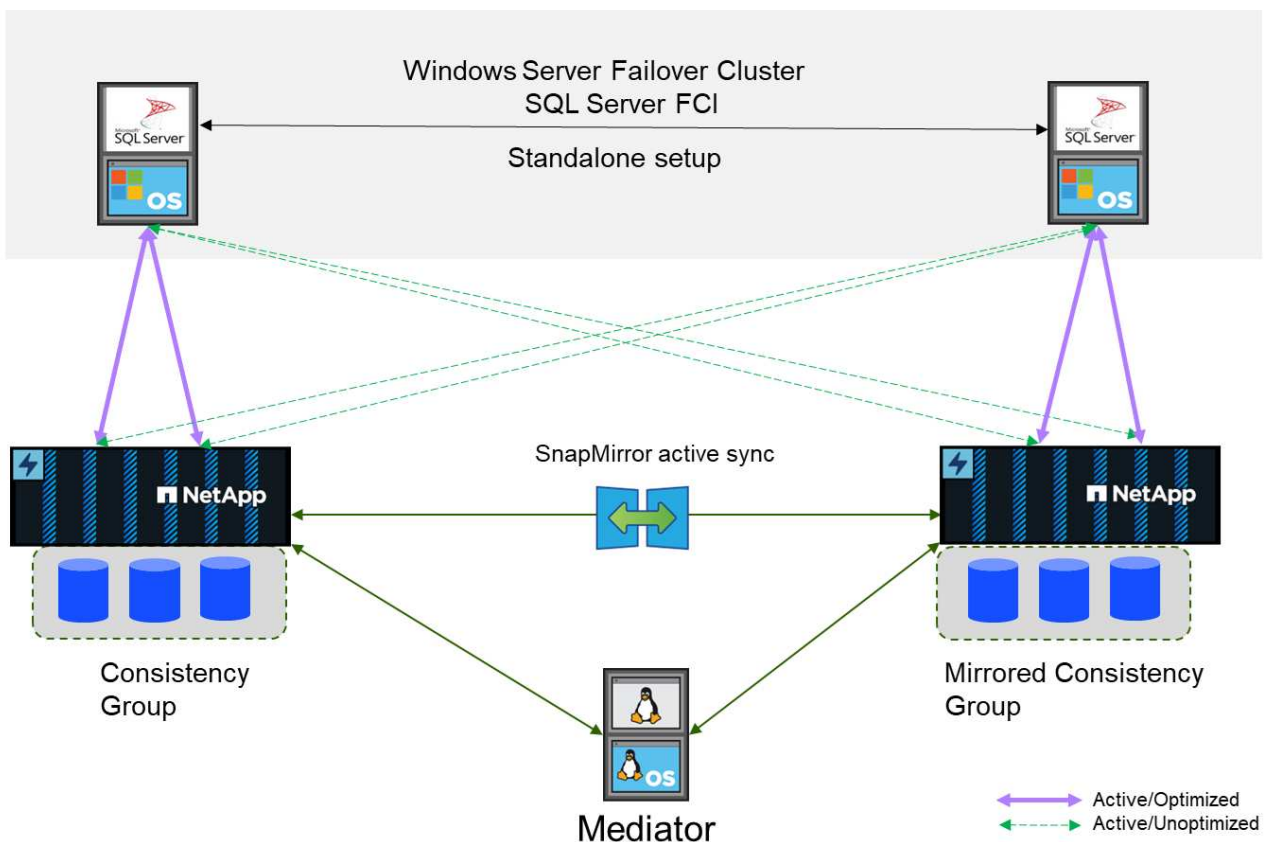
### 概述

通过SnapMirror主动同步、各个SQL Server数据库和应用程序可以在存储和网络中断期间继续运行、并可实现透明的存储故障转移、而无需任何手动干预。

从ONTAP 9 15.1开始、除了现有的非对称配置之外、SnapMirror主动同步还支持对称主动/主动架构。对称主动/主动功能提供同步双向复制、以实现业务连续性和灾难恢复。它可以跨多个故障域同时对数据进行读写访问、从而帮助您保护关键SAN工作负载的数据访问、从而确保无中断运行、并在发生灾难或系统故障时最大限度地减少停机时间。

SQL Server主机使用光纤通道(Fibre Channel、FC)或iSCSI LUN访问存储。在托管已复制数据副本的每个集群之间进行复制。由于此功能是存储级别复制、因此在独立主机或故障转移集群实例上运行的SQL Server实例可以对任一集群执行读/写操作。有关规划和配置步骤,请参阅["有关SnapMirror active sync的ONTAP文档"](#)。

### SnapMirror主动同步架构与对称主动/主动架构



### 同步复制

在正常操作下、每个副本始终是一个RPO = 0的同步副本、但有一个例外。如果无法复制数据、则ONTAP将不再需要复制数据并恢复在一个站点上提供IO、而另一个站点上的LUN将脱机。

### 存储硬件



与其他存储灾难恢复解决方案不同、SnapMirror主动同步可提供非对称平台灵活性。每个站点的硬件不必相同。通过此功能、您可以调整用于支持SnapMirror活动同步的硬件的大小。如果需要支持完整的生产工作负载、远程存储系统可以与主站点完全相同；但是、如果灾难导致I/O减少、则与远程站点上较小的系统相比、可能会更经济高效。

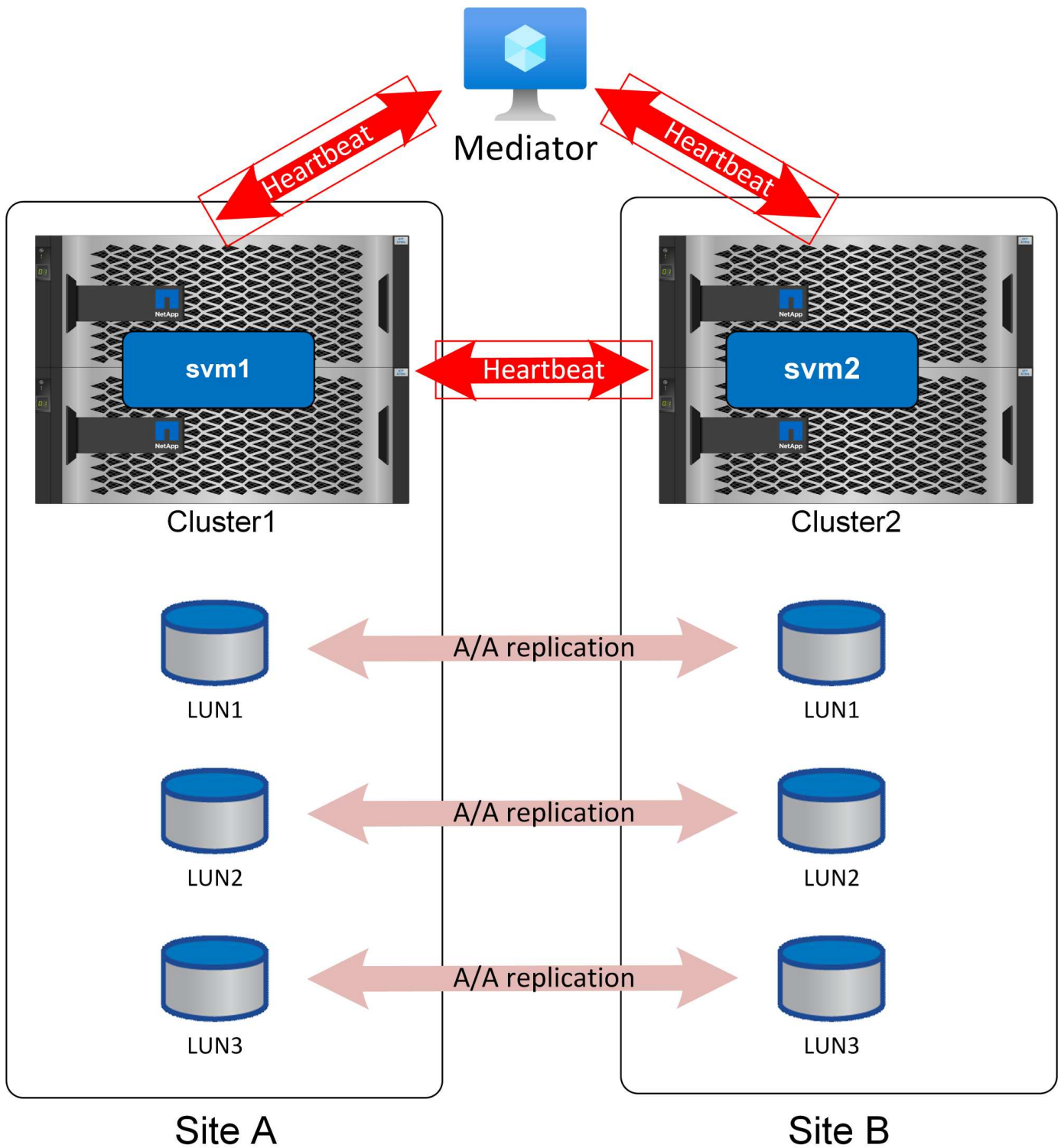
- **ONTAP调解器\*\***

ONTAP调解器是从NetApp支持下载的软件应用程序、通常部署在小型虚拟机上。ONTAP调解器不是Tieb破碎机。它是参与SnapMirror活动同步复制的两个集群的备用通信通道。ONTAP根据通过直接连接和调解器从合作伙伴处收到的响应来推动自动化操作。

#### **ONTAP调解器**

要安全地自动执行故障转移、需要使用调解器。理想情况下、它会放置在独立的第三个站点上、但如果与参与复制的集群之一主机代管、它仍可满足大多数需求。

调解员实际上并不是打破僵局的人，尽管它实际上发挥了这样的作用。中介器有助于确定集群节点的状态，并在站点发生故障时协助自动切换过程。在任何情况下，中介都不会传输数据。



自动化故障转移的第一大挑战是脑裂问题、如果两个站点彼此断开连接、就会出现该问题。应该发生什么？您不希望让两个不同的站点将自己指定为数据的无故障副本、但单个站点如何区分实际丢失相对站点与无法与相反站点通信之间的区别？

这是调解者进入画面的地方。如果放置在第三个站点上、并且每个站点都与该站点建立了单独的网络连接、则每个站点都有一条额外的路径来验证另一个站点的运行状况。再次查看上图、并考虑以下情形。

- 如果调解器发生故障或无法从一个或两个站点访问、会发生什么情况？

- 两个集群仍可通过复制服务所使用的同一链路彼此通信。
- 数据仍会提供RPO = 0保护
- 如果站点A发生故障、会发生什么情况？
  - 站点B将看到两个通信通道关闭。
  - 站点B将接管数据服务、但没有RPO = 0镜像
- 如果站点B发生故障、会发生什么情况？
  - 站点A将看到两个通信通道关闭。
  - 站点A将接管数据服务、但没有RPO = 0镜像

还需要考虑另一种情形：丢失数据复制链路。如果站点之间的复制链路丢失、显然无法执行RPO = 0镜像。那么应该发生什么呢？

这由首选站点状态控制。在SM-AS关系中、其中一个站点是另一个站点的二级站点。这对正常操作没有影响、并且所有数据访问都是对称的、但是如果复制中断、则必须断开连接才能恢复操作。结果是、首选站点将继续操作而不进行镜像、而二级站点将暂停IO处理、直到复制通信恢复为止。

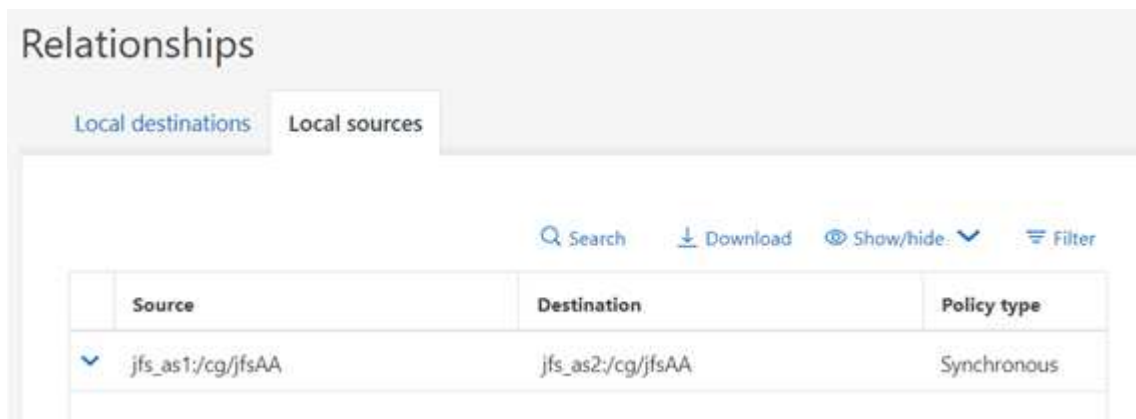
首选站点

SnapMirror主动同步行为是对称的、但有一个重要例外-首选站点配置。

SnapMirror主动同步会将一个站点视为"源"、而将另一个站点视为"目标"。这意味着单向复制关系、但这不适用于IO行为。复制是双向的、对称的、镜像两端的IO响应时间相同。

该 `source` 名称用于控制首选站点。如果复制链路丢失、则源副本上的LUN路径将继续提供数据、而目标副本上的LUN路径将变得不可用、直到重新建立复制并使SnapMirror重新进入同步状态为止。然后、这些路径将恢复提供数据。

可通过SystemManager查看源/目标配置：



The screenshot shows the 'Relationships' section of a management interface. It has two tabs: 'Local destinations' and 'Local sources'. The 'Local sources' tab is selected. Below the tabs is a table with three columns: 'Source', 'Destination', and 'Policy type'. There is one entry in the table with a checkmark icon in the 'Source' column.

| Source              | Destination       | Policy type |
|---------------------|-------------------|-------------|
| ✓ jfs_as1:/cg/jfsAA | jfs_as2:/cg/jfsAA | Synchronous |

或在命令行界面上：

```
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
```

关键在于源是位于第一个Storage Virtual Machine上的SVM。如上所述、术语"源"和"目标"并不表示复制的数据流。两个站点都可以处理写入并将其复制到相反站点。实际上、两个集群都是源和目标。将一个集群指定为源集群的效果只是控制在复制链路丢失时哪个集群作为读写存储系统继续存在。

## 网络拓扑

### 统一访问

统一访问网络意味着主机能够访问两个站点(或同一站点中的故障域)上的路径。

SM-AS的一项重要功能是、可以对存储系统进行配置、使其知道主机所在的位置。将LUN映射到给定主机时、您可以指示它们是否接近给定存储系统。

### 邻近设置

接近是指每个集群的配置、表示特定主机WWN或iSCSI启动程序ID属于本地主机。这是配置LUN访问的第二个可选步骤。

第一步是常规的igrop配置。每个LUN都必须映射到一个igrop、该igrop包含需要访问该LUN的主机的wwn/iSCSI ID。此选项用于控制哪个主机对LUN具有\_access\_访问权限。

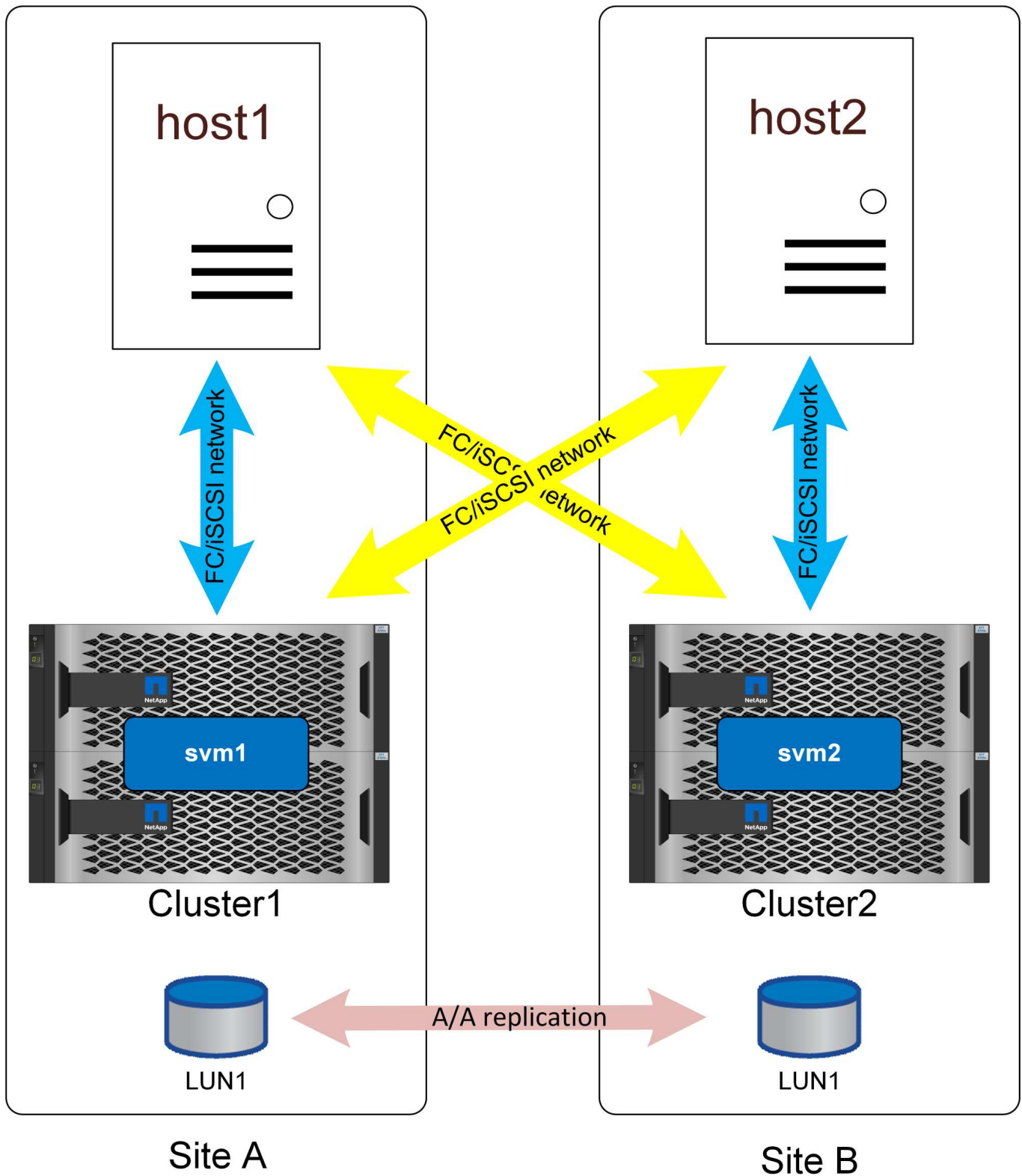
第二个可选步骤是配置主机邻近性。这不控制访问、而是控制\_priority\_。

例如、可以将站点A的主机配置为访问受SnapMirror活动同步保护的LUN、并且由于SAN跨站点扩展、因此可以使用站点A上的存储或站点B上的存储为该LUN提供路径

如果没有邻近设置、则该主机将平等使用这两个存储系统、因为这两个存储系统都会公布主动/优化路径。如果站点之间的SAN延迟和/或带宽有限、则可能无法实现这一点、您可能希望确保在正常操作期间、每个主机优先使用指向本地存储系统的路径。这可通过将主机的wwn/iSCSI ID作为近端主机添加到本地集群来配置。可通过命令行界面或SystemManager完成此操作。

## AFF

对于AFF系统、配置主机邻近性后、路径将如下所示。



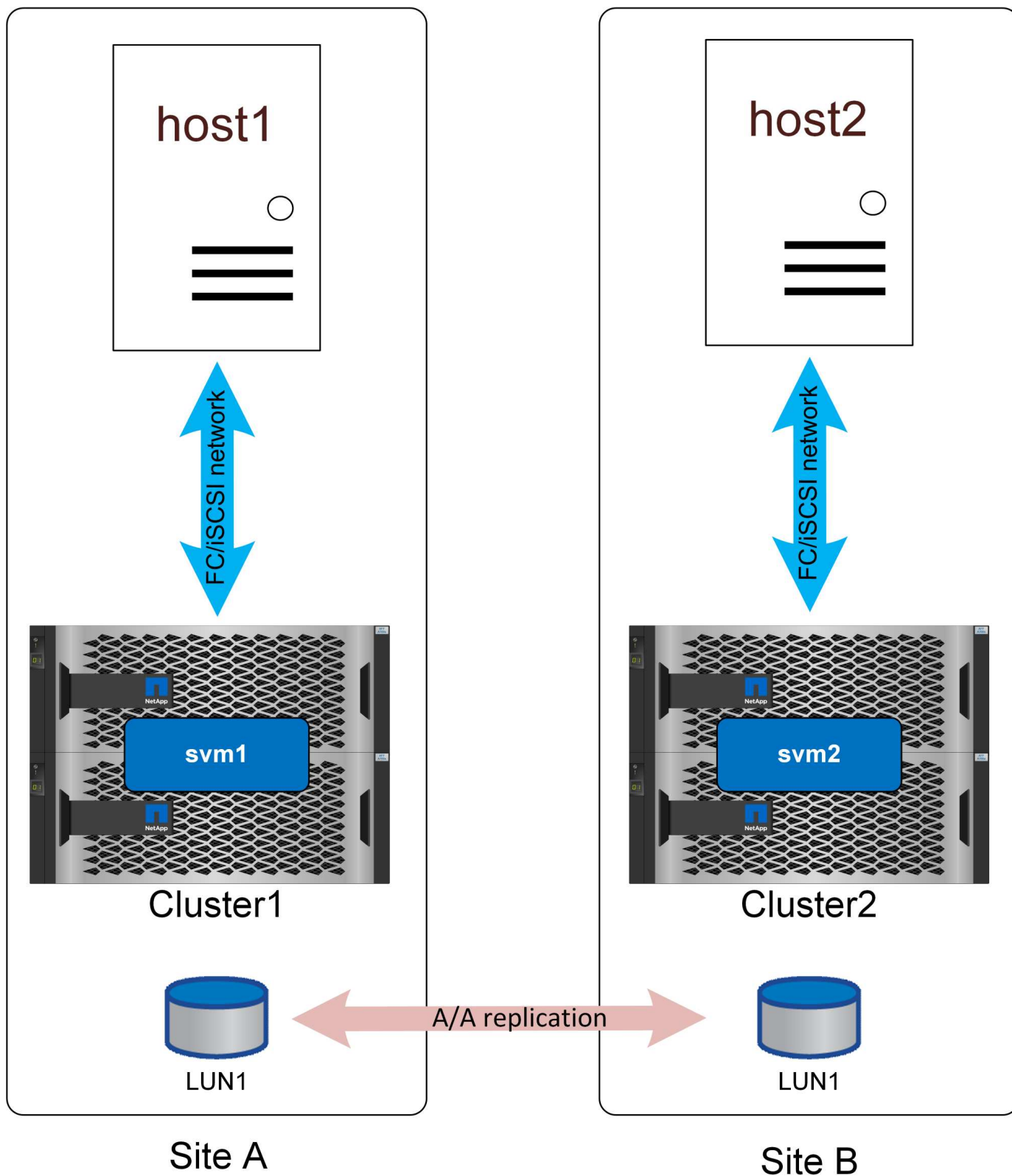
在正常操作下、所有IO均为本地IO。读取和写入操作由本地存储阵列提供。当然、在确认写入IO之前、本地控制器也需要将其复制到远程系统、但所有读取IO都将在本地进行处理、并且不会通过遍历站点间的SAN链路而产生额外延迟。

只有在所有主动/优化路径丢失时、才会使用非优化路径。例如、如果站点A上的整个阵列断电、则站点A上的主机仍可访问站点B上阵列的路径、因此、尽管延迟较长、但仍可保持正常运行。

为了简单起见、这些图中未显示通过本地集群的冗余路径。ONTAP存储系统本身就是HA、因此控制器故障不应导致站点故障。它只会导致受影响站点上使用的本地路径发生更改。

#### 非一致访问

非一致访问网络意味着每个主机只能访问本地存储系统上的端口。SAN不会跨站点(或同一站点内的故障域)进行扩展。



## Active/Optimized Path

这种方法的主要优势是SAN的精简性、您无需在网络上延伸SAN。某些客户的站点间连接延迟不足、或者缺少通过站点间网络传输FC SAN流量的基础架构。



非一致访问的缺点是、某些故障情形(包括丢失复制链路)将导致某些主机无法访问存储。如果本地存储连接丢失、则作为单个实例运行的应用程序(例如、本质上仅在任何给定挂载的单个主机上运行的非集群数据库)将失败。数据仍会受到保护、但数据库服务器将无法再访问。它需要在远程站点上重新启动、最好是通过自动化过程重新启动。例如、VMware HA可以在一台服务器上检测到全路径关闭的情况、并在具有可用路径的另一台服务器上重新启动VM。

相比之下、Oracle RAC等集群应用程序可以提供在两个不同站点上同时提供的服务。丢失站点并不意味着整个应用程序服务都会丢失。实例仍可用、并且在正常运行的站点上运行。

在许多情况下、通过站点间链路访问存储的应用程序所产生的额外延迟开销是不可接受的。这意味着统一网络可用性的提高微乎其微、因为如果站点上丢失存储、则无论如何都需要关闭故障站点上的服务。



为了简单起见、这些图中未显示通过本地集群的冗余路径。ONTAP存储系统本身就是HA、因此控制器故障不应导致站点故障。它只会导致受影响站点上使用的本地路径发生更改。

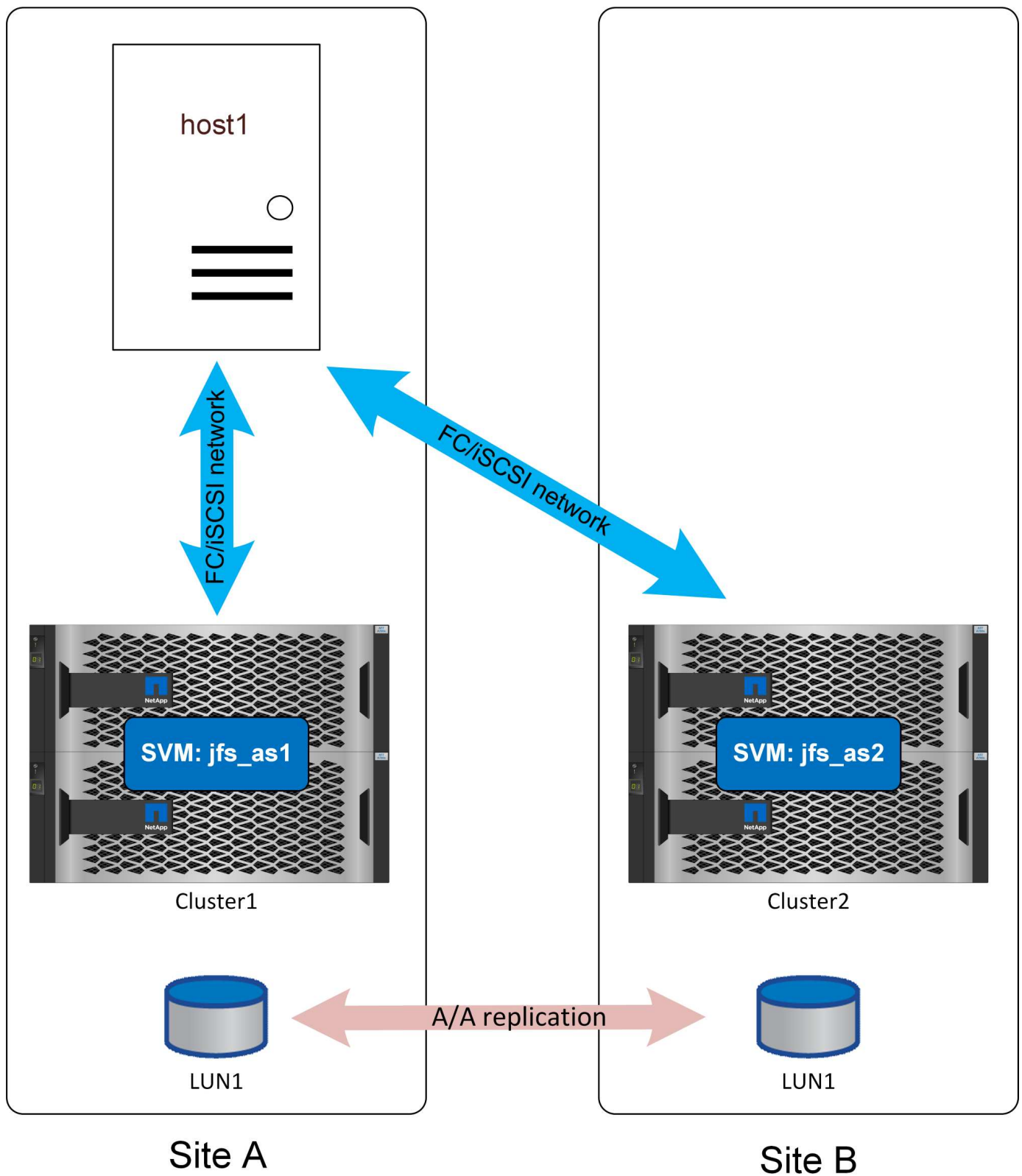
#### 概述

可以将SQL Server配置为以多种方式与SnapMirror活动同步配合使用。正确答案取决于可用的网络连接、RPO要求和可用性要求。

#### SQL Server的独立实例

文件布局和服务器配置的最佳实践与文档中建议的相同["基于ONTAP的SQL Server"](#)。

使用独立设置时、SQL Server只能在一个站点上运行。可能["统一"](#)会使用访问权限。



使用统一访问时、任一站点的存储故障都不会中断数据库操作。当然、如果包含数据库服务器的站点发生完全故障、则会导致中断。

某些客户可以为在远程站点上运行的操作系统配置预先配置的SQL Server设置、并使用与生产实例相同的构建版本进行更新。故障转移需要激活备用站点上的独立SQL Server实例、发现LUN并启动数据库。由于不需要从存储端执行任何操作、因此可以使用Windows PowerShell cmdlet自动完成整个过程。

"非一致性"也可以使用访问、但如果数据库服务器所在的存储系统因数据库没有可用的存储路径而出现故障、则会导致数据库中断。在某些情况下、这种情况仍可接受。SnapMirror主动同步仍可提供RPO = 0的数据保护、并且在站点发生故障时、运行正常的副本将处于活动状态、并可使用与上述统一访问相同的过程恢复操作。

使用虚拟化主机可以更轻松地配置简单的自动化故障转移过程。例如、如果SQL Server数据文件与启动VMDK一起同步复制到二级存储、则在发生灾难时、可以在备用站点上激活整个环境。管理员可以在正常运行的站点上手动激活主机、也可以通过VMware HA等服务自动执行此过程。

## SQL Server故障转移集群实例

SQL Server故障转移实例也可以托管在作为子操作系统运行在物理服务器或虚拟服务器上的Windows故障转移集群上。这种多主机架构可提供SQL Server实例和存储故障恢复能力。在需要在保持增强性能的同时实现强大故障转移流程的高需求环境中、此类部署非常有用。在故障转移集群设置中、当主机或主存储受到影响时、SQL服务将故障转移到二级主机、同时、二级存储将可用于提供IO。无需自动化脚本或管理员干预。

### 故障情形

要规划完整的SnapMirror主动同步应用程序架构、需要了解SM-AS如何在各种计划内和计划外故障转移场景中做出响应。

在以下示例中、假设站点A已配置为首选站点。

### 复制连接丢失

如果SM-AS复制中断、则无法完成写入IO、因为集群无法将更改复制到相反站点。

#### 站点A (首选站点)

首选站点上的复制链路故障会导致写入IO处理暂停大约15秒、因为ONTAP会在确定复制链路确实无法访问之前重试复制的写入操作。15秒后、站点A系统将恢复读取和写入IO处理。SAN路径不会更改、LUN将保持联机状态。

#### 站点 B

由于站点B不是SnapMirror主动同步首选站点、因此其LUN路径将在大约15秒后变得不可用。

### 存储系统故障

存储系统故障的结果与丢失复制链路的结果几乎相同。正常运行的站点应出现大约15秒的IO暂停。15秒过后、IO将照常在该站点上恢复。

### 调解器丢失

调解器服务不直接控制存储操作。它可用作集群之间的备用控制路径。它主要用于自动执行故障转移、而不存在脑裂情况的风险。在正常操作下、每个集群都会将更改复制到其配对集群、因此、每个集群都可以验证配对集群是否联机并提供数据。如果复制链路失败、复制将停止。

安全自动故障转移需要调解器的原因是、否则存储集群将无法确定双向通信丢失是网络中断还是实际存储故障所致。

调解器为每个集群提供一个备用路径、以验证其配对集群的运行状况。具体情形如下：

- 如果集群可以直接与其配对集群联系、则复制服务将正常运行。无需执行任何操作。
- 如果首选站点无法直接或通过调解器与其配对站点联系、则会假定配对站点实际不可用或已隔离、并且其LUN路径已脱机。然后、首选站点将继续释放RPO = 0状态、并继续处理读写IO。
- 如果非首选站点无法直接与其配对站点联系、但可以通过调解器与其联系、则它会使其路径脱机、并等待复制连接返回。
- 如果非首选站点无法直接联系其配对站点或无法通过操作调解器联系其配对站点、则会假定配对站点实际不可用或已隔离、并且其LUN路径已脱机。然后、非首选站点将继续释放RPO = 0状态、并继续处理读写IO。它将承担复制源的角色、并成为新的首选站点。

如果调解器完全不可用：

- 复制服务因任何原因发生故障(包括非首选站点或存储系统发生故障)、都会导致首选站点释放RPO = 0状态并恢复读写IO处理。非首选站点将使其路径脱机。
- 首选站点发生故障将导致中断、因为非首选站点无法验证对等站点是否真正脱机、因此非首选站点无法安全地恢复服务。

### 正在还原服务

解决故障(例如、还原站点间连接或启动故障系统)后、SnapMirror活动同步端点将自动检测是否存在故障复制关系、并将其恢复为RPO = 0状态。重新建立同步复制后、故障路径将再次联机。

在许多情况下、集群模式应用程序会自动检测故障路径的返回情况、这些应用程序也会恢复联机。在其他情况下、可能需要进行主机级SAN扫描、或者可能需要手动将应用程序恢复联机。它取决于应用程序及其配置方式、通常、此类任务可以轻松实现自动化。ONTAP本身具有自我修复能力、不需要任何用户干预即可恢复RPO = 0存储操作。

### 手动故障转移

更改首选站点只需简单的操作即可。在集群之间切换复制行为的权限时、IO将暂停一两秒钟、但IO不会受到影响。

## ASA R2系统上的存储配置

### 概述

NetApp ASA R2是一款简化且功能强大的解决方案、适用于运行任务关键型工作负载的纯SAN客户。运行ONTAP存储解决方案的ASA R2平台与Microsoft SQL Server相结合、可以实现企业级数据库存储设计、从而满足当今最苛刻的应用程序要求。

以下ASA平台归类为支持所有SAN协议(iSCSI、FC、NVMe/FC、NVMe/TCP)的ASA R2系统。iSCSI、FC、NVMe/FC和NVMe/TCP协议支持对称双主动多路径架构、以便主机和存储之间的所有路径都处于主动/优化状态：

- ASAA1K
- ASAA90
- ASAA70
- ASAA50

- ASA A30
- ASA A20

有关详细信息、请参见 ["NetApp ASA"](#)

要优化基于ONTAP的SQL Server解决方案、需要了解SQL Server I/O模式和特征。为SQL Server数据库精心设计的存储布局必须支持SQL Server的性能要求、同时还必须在整体上最大限度地提高基础架构的可管理性。良好的存储布局还可以成功完成初始部署、并随着业务的增长逐步平稳地扩展环境。

## 数据存储设计

Microsoft建议将数据和日志文件放在不同的驱动器上。对于同时更新和请求数据的应用程序、日志文件会占用大量写入资源、而数据文件(取决于您的应用程序)会占用大量读/写资源。对于数据检索、不需要日志文件。因此、可以通过放置在其自己驱动器上的数据文件来满足数据请求。

创建新数据库时、Microsoft建议为数据和日志指定单独的驱动器。要在创建数据库后移动文件、必须使数据库脱机。有关Microsoft的更多建议、请参见 ["将数据和日志文件放在不同的驱动器上"](#)。

## 存储单元注意事项

ASA中的存储单元是指SCSI/FC主机的LUN或NVMe主机的命名空间。根据支持的协议、系统将提示您创建LUN和/或NVMe命名空间。在为数据库部署创建存储单元之前、请务必了解SQL Server I/O模式和特征如何根据工作负载以及备份和恢复要求而有所不同。请参见以下存储单元的NetApp建议：

- 避免在同一主机上运行的多个SQL Server之间共享同一存储单元、以避免复杂的管理。如果在同一主机上运行多个SQL Server实例、则除非某个节点上的存储单元接近限制、否则请避免共享、而是为每个主机的每个实例设置单独的存储单元、以简化数据管理。
- 使用NTFS挂载点而不是驱动器盘符、以超过Windows中26个驱动器盘符的限制。
- 禁用快照计划和保留策略。请改用SnapCenter来协调SQL Server数据存储单元的Snapshot副本。
- 将SQL Server系统数据库放在专用存储单元上。
- tempdb是一个系统数据库、SQL Server会使用它作为临时工作空间、尤其是用于I/O密集型DBCC CHECKDB操作。因此、请将此数据库放在专用存储单元上。在存储单元计数存在挑战的大型环境中、您可以在仔细规划后将tempdb与系统数据库整合到同一存储单元中。tempdb的数据保护不是高优先级、因为每次重新启动SQL Server时都会重新创建此数据库。
- 将用户数据文件(.mdf)放在单独的存储单元上、因为它们是随机读/写工作负载。通常、创建事务日志备份的频率比创建数据库备份的频率更高。因此、请将事务日志文件(.ldf)放在与数据文件不同的存储单元或VMDK上、以便可以为每个存储单元或VMDK创建独立的备份计划。这种分离还会将日志文件的顺序写入I/O与数据文件的随机读/写I/O隔离开来、并显著提高SQL Server性能。
- 确保用户数据库文件和用于存储日志备份的日志目录位于单独的存储单元上、以防止保留策略在存储策略中的SnapMirror功能中使用快照时覆盖快照。
- 请勿在同一存储单元上混合使用数据库文件和非数据库文件、例如与全文搜索相关的文件。
- 将数据库二级文件(作为文件组的一部分)放在单独的存储单元上可提高SQL Server数据库的性能。只有当数据库的文件不与任何其他文件共享其存储单元 .mdf`时、此隔离才有效 ` .mdf。
- 在Windows服务器中使用磁盘管理器格式化磁盘时、请确保分区的分配单元大小设置为64K。
- 请勿将用户数据库或系统数据库放置在托管挂载点的存储单元上。
- 请参见 ["Microsoft Windows和本机MPIO、并遵循适用于现代SAN的ONTAP最佳实践"](#) 在MPIO属性中将Windows上的多路径支持应用于iSCSI设备。



- 如果您使用的是无中断故障转移集群实例、则必须将用户数据库置于在Windows服务器故障转移集群节点之间共享的存储单元上、并且物理磁盘集群资源会分配给与SQL Server实例关联的集群组。

## 数据库文件和文件组

在初始部署阶段、在ONTAP上正确放置SQL Server数据库文件至关重要。这样可以确保最佳性能、空间管理、备份和还原时间、并可根据您的业务需求进行配置。

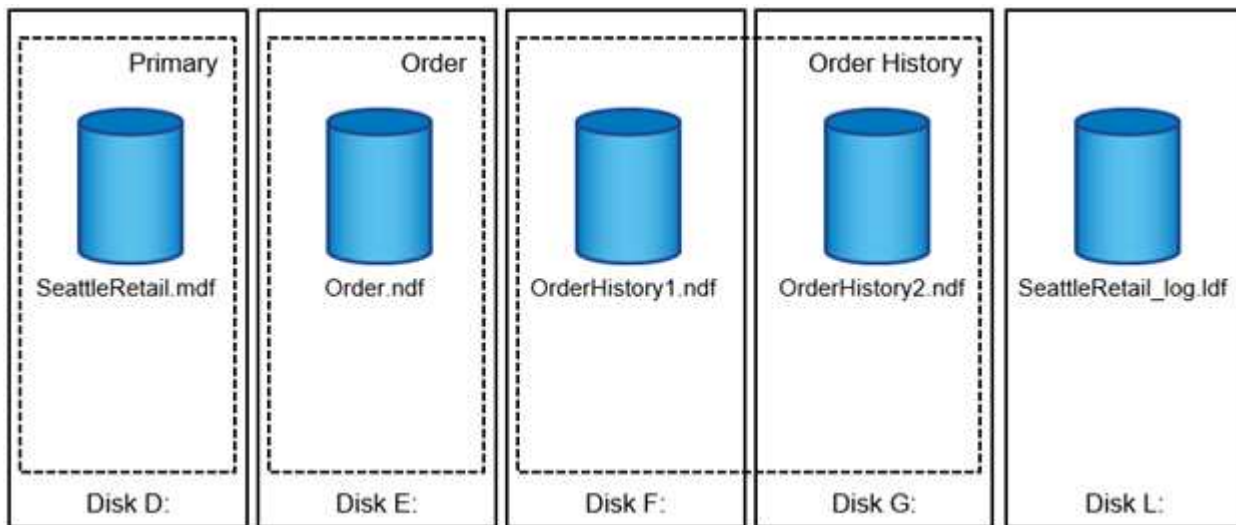
从理论上讲、SQL Server (64位)支持每个实例32、767个数据库以及524、272 TB的数据库大小、尽管典型安装通常包含多个数据库。但是、SQL Server可以处理的数据库数量取决于负载和硬件。SQL Server实例托管数十个、数百个甚至数千个小型数据库的情况并不少见。

### 数据库文件和文件组

每个数据库由一个或多个数据文件和一个或多个事务日志文件组成。事务日志存储有关数据库事务以及每个会话所做的所有数据修改的信息。每次修改数据时、SQL Server都会在事务日志中存储足够的信息、以便撤消(回滚)或重做(重放)操作。SQL Server事务日志是SQL Server在数据完整性和稳定性方面声誉的重要组成部分。事务日志对于SQL Server的运行异常、一致性、隔离和持久性(ACID)功能至关重要。一旦对数据页面进行任何更改、SQL Server就会写入事务日志。每个数据操作语言(DML)语句(例如SELECT、INSERT、UPDATE-DELETE)都是一个完整的事务、事务日志可确保执行整个基于集合的操作、从而确保事务的正常运行。

每个数据库都有一个主数据文件、默认情况下、该文件的扩展名为.mdf。此外、每个数据库都可以具有二级数据库文件。默认情况下、这些文件的扩展名为.NDF。

所有数据库文件都分组到文件组中。文件组是逻辑单元、可简化数据库管理。它们允许在逻辑对象放置和物理数据库文件之间进行隔离。创建数据库对象表时、您可以指定应将其放置在哪个文件组中、而不必担心基础数据文件配置。



通过将多个数据文件放在文件组中、您可以将负载分布在不同的存储设备上、这有助于提高系统的I/O性能。相反、事务日志不会从多个文件中受益、因为SQL Server会按顺序写入事务日志。

通过将文件组中的逻辑对象放置与物理数据库文件分隔开、您可以微调数据库文件布局、从而最大程度地利用存储子系统。支持指定工作负载的数据文件数量可以根据需要进行更改、以支持I/O要求和预期容量、而不会影响应用程序。数据库布局中的这些变化对于应用程序开发人员来说是透明的、他们将数据库对象放在文件组中、而不是数据库文件中。





\* NetApp建议\*避免将主文件组用于除系统对象之外的任何其他对象。为用户对象创建单独的文件组或一组文件组可简化数据库管理和灾难恢复、尤其是在大型数据库的情况下。

#### 数据库实例文件初始化

您可以在创建数据库或向现有数据库添加新文件时指定初始文件大小和自动增长参数。SQL Server在选择应将数据写入哪个数据文件时使用比例填充算法。它会将大量数据按比例写入文件中的可用空间。文件中的可用空间越多、它处理的写入就越多。



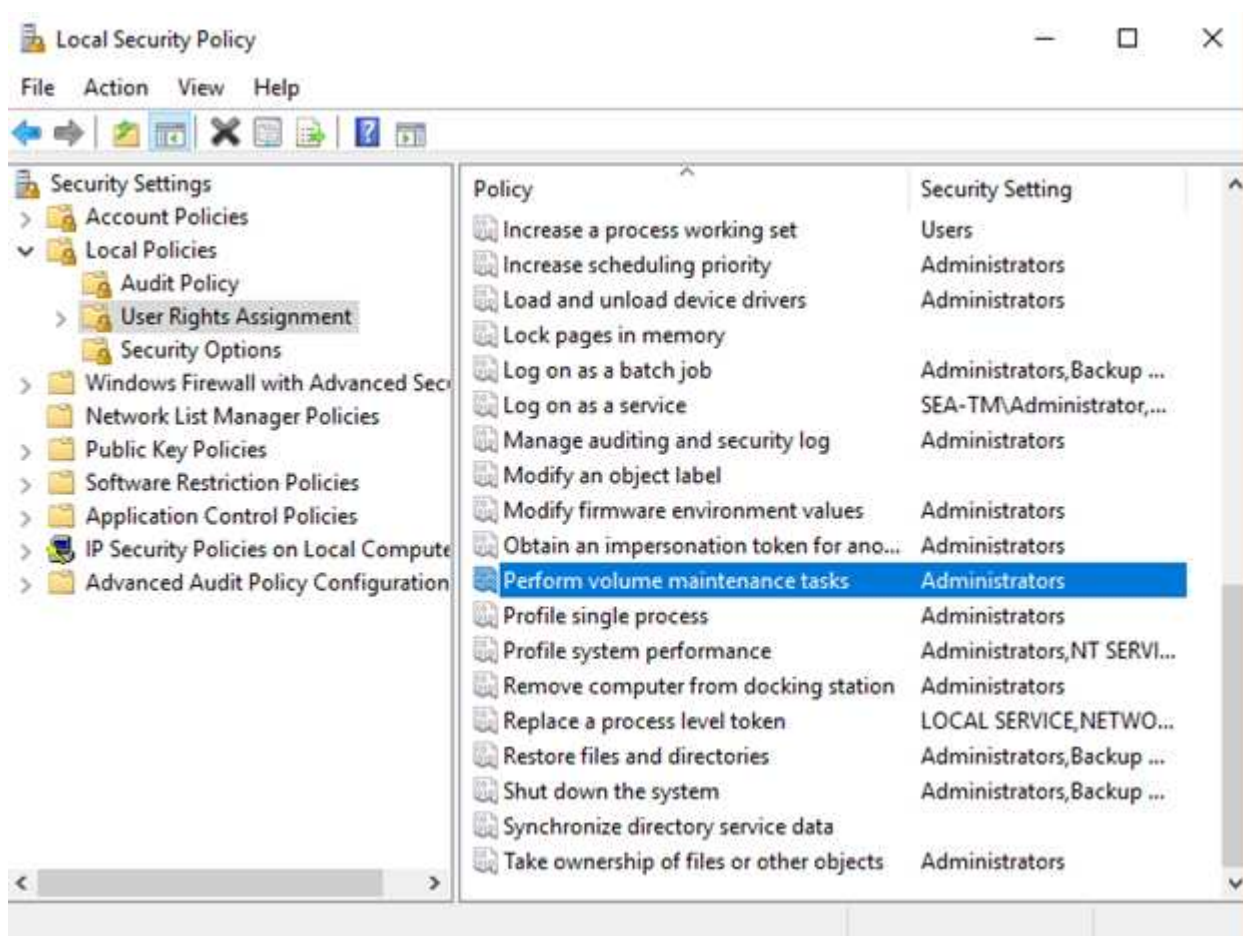
\* NetApp建议\*单个文件组中的所有文件都具有相同的初始大小和自动增长参数，并且增长大小以MB而不是百分比为单位定义。这有助于比例填充算法均匀地平衡数据文件之间的写入活动。

每当SQL Server增加文件大小时、它都会用零填充新分配的空间。该进程会阻止所有需要写入相应文件的会话、或者在事务日志增长时生成事务日志记录。

SQL Server始终将事务日志零、并且此行为无法更改。但是、您可以通过启用或禁用即时文件初始化来控制数据文件是否置零。启用即时文件初始化有助于加快数据文件增长速度、并减少创建或还原数据库所需的时间。

即时文件初始化会带来较小的安全风险。启用此选项后、数据文件的未分配部分可能包含先前删除的操作系统文件中的信息。数据库管理员可以检查此类数据。

您可以通过向SQL Server启动帐户添加SA\_Manage\_volume\_name权限(也称为"执行卷维护任务")来启用即时文件初始化。您可以在本地安全策略管理应用程序(secpol.msc)下执行此操作、如下图所示。打开"Perform volume Maintenance Task (执行卷维护任务)"权限的属性、并将SQL Server启动帐户添加到其中的用户列表中。



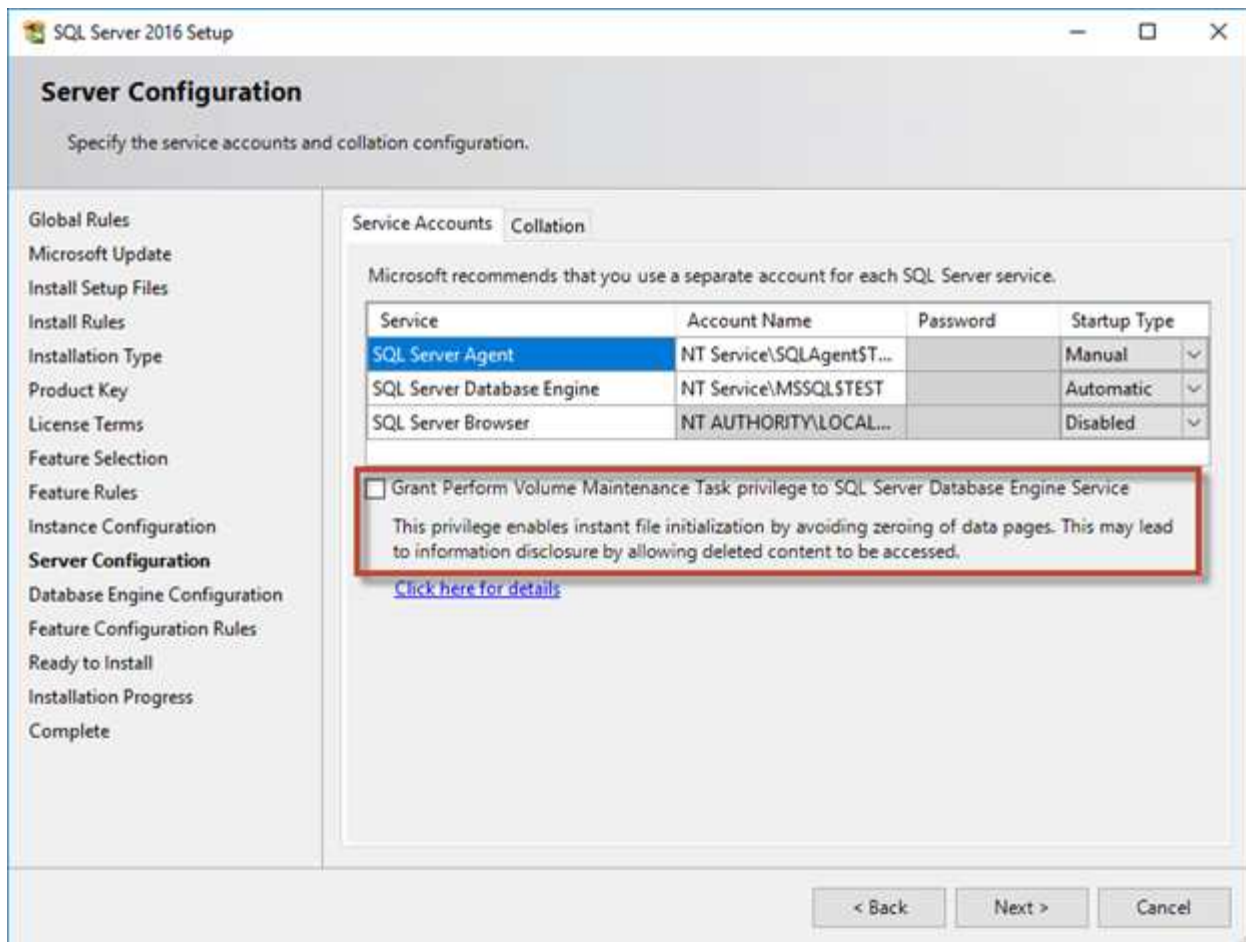
要检查权限是否已启用、您可以使用以下示例中的代码。此代码设置了两个跟踪标志、用于强制SQL Server将追加信息写入错误日志、创建小型数据库以及读取日志内容。

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

如果未启用即时文件初始化、则SQL Server错误日志会显示、除了将ldf日志文件置零之外、SQL Server还会将MDF数据文件置零、如以下示例所示。启用即时文件初始化后、它仅显示日志文件置零。

|     | 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      | FixupLog Tail(progress) zeroing C:\Program Files\Microsoft SQL Server\MSSQL\LOG\ |
| 372 | 2017-02-09 08:10:08.160 | spid53      | Zeroing C:\Program Files\Microsoft SQL Server\MSSQL\LOG\                         |
| 373 | 2017-02-09 08:10:08.170 | spid53      | Zeroing completed on C:\Program Files\Microsoft SQL Server\MSSQL\LOG\            |
| 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.   |

在SQL Server 2016中、执行卷维护任务得到了简化、稍后在安装过程中作为一个选项提供。此图显示了为SQL Server数据库引擎服务授予执行卷维护任务权限的选项。



控制数据库文件大小的另一个重要数据库选项是自动缩放。启用此选项后、SQL Server会定期缩减数据库文件、减小其大小并向操作系统释放空间。此操作会占用大量资源、并且很少有用、因为新数据进入系统一段时间后、数据库文件会再次增长。不应在数据库上启用自动缩放。

## 日志目录

日志目录在SQL Server中指定、用于在主机级别存储事务日志备份数据。如果使用SnapCenter备份日志文件、则SnapCenter使用的每个SQL Server主机都必须配置一个主机日志目录、才能执行日志备份。

将日志目录放在专用存储单元上。主机日志目录中的数据量取决于备份的大小以及备份的保留天数。SnapCenter只允许每个SQL Server主机有一个主机日志目录。您可以在SnapCenter → Host → Configure Plug-in中配置主机日志目录。



- NetApp建议\*对主机日志目录执行以下操作：
- 确保主机日志目录不会被任何可能损坏备份快照数据的其他类型的数据共享。
- 在SnapCenter将事务日志复制到的专用存储单元上创建主机日志目录。
- 如果您使用的是无中断故障转移集群实例、则用于主机日志目录的存储单元必须是与要在SnapCenter中备份的SQL Server实例位于同一集群组中的集群磁盘资源。

## 数据保护

数据库备份策略应基于已确定的业务要求、而不是理论能力。通过将ONTAP的Snapshot技

术与Microsoft SQL Server API相结合、无论用户数据库大小如何、您都可以快速创建应用程序一致的备份。为了满足更高级或横向扩展的数据管理要求、NetApp提供了SnapCenter。

## SnapCenter

SnapCenter是一款适用于企业级应用程序的NetApp数据保护软件。使用适用于SQL Server的SnapCenter插件以及适用于Microsoft Windows的SnapCenter插件管理的操作系统、可以快速轻松地保护SQL Server数据库。

SQL Server实例可以是独立设置、故障转移集群实例、也可以始终位于可用性组上。这样、可以通过单一管理平台从主副本或二级副本保护、克隆和还原数据库。SnapCenter可以管理内部、云端和混合配置中的SQL Server数据库。此外、还可以在原始主机或备用主机上几分钟内创建数据库副本、以供开发或报告之用。

SQL Server还要求操作系统和存储之间进行协调、以确保创建快照时Snapshot中存在正确的数据。在大多数情况下、执行此操作的唯一安全方法是使用SnapCenter或T-SQL。如果未进行这种额外协调、则创建的快照可能无法可靠地恢复。

有关适用于SnapCenter的SQL Server插件的详细信息、请参见 ["TR-4714：《使用NetApp SnapCenter的SQL Server最佳实践指南》"](#)。

## 使用T-SQL快照保护数据库

在SQL Server 2022中、Microsoft引入了T-SQL快照、可通过它编写脚本并自动执行备份操作。您可以为快照准备数据库、而不是执行全尺寸副本。数据库备份就绪后、您可以利用ONTAP REST API创建快照。

以下是备份工作流示例：

1. 使用alter命令冻结数据库。这样可以使数据库在底层存储上为一致的快照做好准备。冻结后、您可以解冻数据库并使用backup命令记录快照。
2. 使用新的backup group和backup server命令同时对存储单元上的多个数据库执行快照。
3. 如果数据库工作负载跨多个存储单元、请创建一致性组以简化管理任务。一致性组是指作为单个单元进行管理的一组存储单元。
4. 执行完整备份或copy\_only完整备份。这些备份也会记录在msdb中。
5. 在快照完整备份之后、使用通过正常流式传输方法创建的日志备份执行时间点恢复。如果需要、还支持流式差异备份。

要了解更多信息，请参见 ["了解T-SQL快照的Microsoft文档"](#)。



\* NetApp建议\*使用SnapCenter创建Snapshot副本。上述T-SQL方法同样有效、但SnapCenter可实现备份、还原和克隆过程的完全自动化。同时、它还会执行发现以确保创建的快照正确无误。

## 灾难恢复

### 灾难恢复

企业数据库和应用程序基础架构通常需要进行复制、以防止发生自然灾害或意外业务中断、同时最大限度地减少停机时间。

SQL Server无中断可用性组复制功能是一个绝佳的选择、NetApp提供了将数据保护与无中断集成的选项。但



是、在某些情况下、您可能需要考虑使用以下选项的ONTAP复制技术。

## SnapMirror

SnapMirror技术为通过LAN和广域网复制数据提供了快速灵活的企业解决方案。SnapMirror技术仅在创建初始镜像后将更改后的数据块传输到目标、从而显著降低网络带宽要求。它可以配置为同步或异步模式。NetApp ASA中的SnapMirror同步复制使用SnapMirror活动同步进行配置。

### SnapMirror活动同步

对于许多客户来说、业务连续性不仅需要拥有远程数据副本、还需要能够快速利用这些数据、而在NetApp ONTAP中、可以使用SnapMirror主动同步来实现这一点

使用SnapMirror主动同步时、您实际上拥有两个不同的ONTAP系统、它们会维护LUN数据的独立副本、但会相互协作、为该LUN提供一个实例。从主机角度来看、它是一个LUN实体。基于iSCSI/FC的LUN支持SnapMirror主动同步。

SnapMirror主动同步可以提供RPO = 0复制、并且可以轻松地在两个独立的集群之间实施。当两个数据副本处于同步状态后、两个集群只需镜像写入即可。在一个集群上进行写入时、该写入会复制到另一个集群。只有在两个站点上的写入均已完成时、才会向主机确认写入。除了此协议拆分为行为之外、这两个集群在其他方面都是正常的ONTAP集群。

SM-AS的一个关键用例是粒度复制。有时、您不希望将所有数据作为一个单元进行复制、或者您需要能够有选择地对某些工作负载进行故障转移。

SM-AS的另一个主要用例是主动-主动操作、您希望在位于两个不同位置的两个不同集群上提供完全可用的数据副本、这些集群具有相同的性能特征、如果需要、也不需要站点间延伸SAN。如果应用程序受支持、则您的应用程序可能已在这两个站点上运行、这样可以减少故障转移操作期间的整体恢复时间。

## SnapMirror

下面是有关适用于SQL Server的SnapMirror的建议：

- 同步复制与SnapMirror主动同步结合使用、后者对快速数据恢复的需求较高、而异步解决方案则可实现RPO的灵活性。
- 如果您使用SnapCenter备份数据库并将快照复制到远程集群、请勿出于一致性目的从控制器计划SnapMirror更新。而是应启用SnapCenter中的SnapMirror更新、以便在完整备份或日志备份完成后更新SnapMirror。
- 在集群中的不同节点之间平衡包含SQL Server数据的存储单元、以允许所有集群节点共享SnapMirror复制活动。此分布可优化节点资源的使用。

有关SnapMirror的详细信息、请参见 ["TR-4015：《适用于ONTAP 9的SnapMirror配置和最佳实践指南》"](#)。

### SnapMirror活动同步

#### 概述

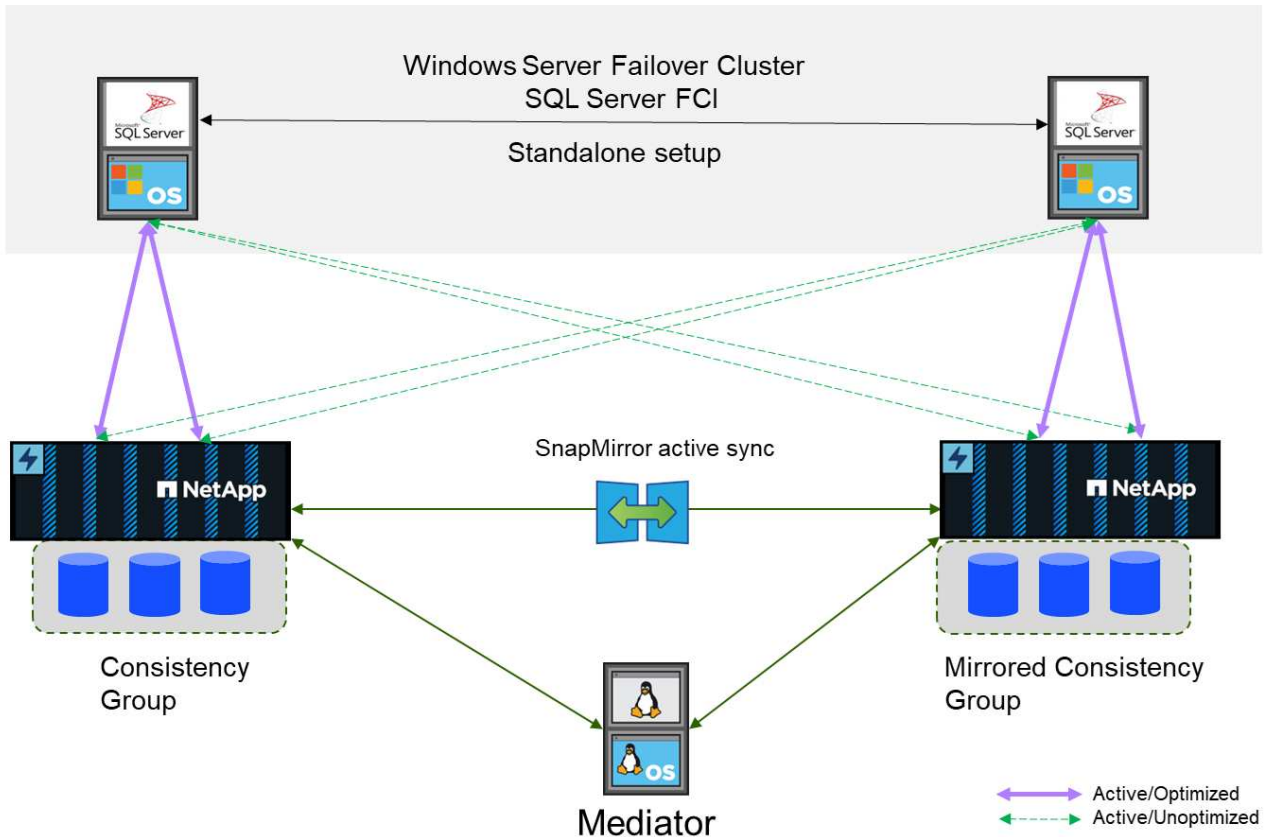
通过SnapMirror主动同步、各个SQL Server数据库和应用程序可以在存储和网络中断期间继续运行、并可实现透明的存储故障转移、而无需任何手动干预。

SnapMirror主动同步支持对称主动/主动架构、可提供同步双向复制、以实现业务连续性和灾难恢复。它可以跨多个故障域同时对数据进行读写访问、从而帮助您保护关键SAN工作负载的数据访问、从而确保无中断运行、并

在发生灾难或系统故障时最大限度地减少停机时间。

SQL Server主机使用光纤通道(Fibre Channel、FC)或iSCSI LUN访问存储。在托管已复制数据副本的每个集群之间进行复制。由于此功能是存储级别复制、因此在独立主机或故障转移集群实例上运行的SQL Server实例可以对任一集群执行读/写操作。有关规划和配置步骤,请参阅["有关SnapMirror active sync的ONTAP文档"](#)。

### SnapMirror主动与对称主动/主动的架构



### 同步复制

在正常操作下、每个副本始终是一个RPO = 0的同步副本、但有一个例外。如果无法复制数据、则ONTAP将不再需要复制数据并恢复在一个站点上提供IO、而另一个站点上的LUN将脱机。

### 存储硬件

与其他存储灾难恢复解决方案不同、SnapMirror主动同步可提供非对称平台灵活性。每个站点的硬件不必相同。通过此功能、您可以调整用于支持SnapMirror活动同步的硬件的大小。如果需要支持完整的生产工作负载、远程存储系统可以与主站点完全相同;但是、如果灾难导致I/O减少、则与远程站点上较小的系统相比、可能会更经济高效。

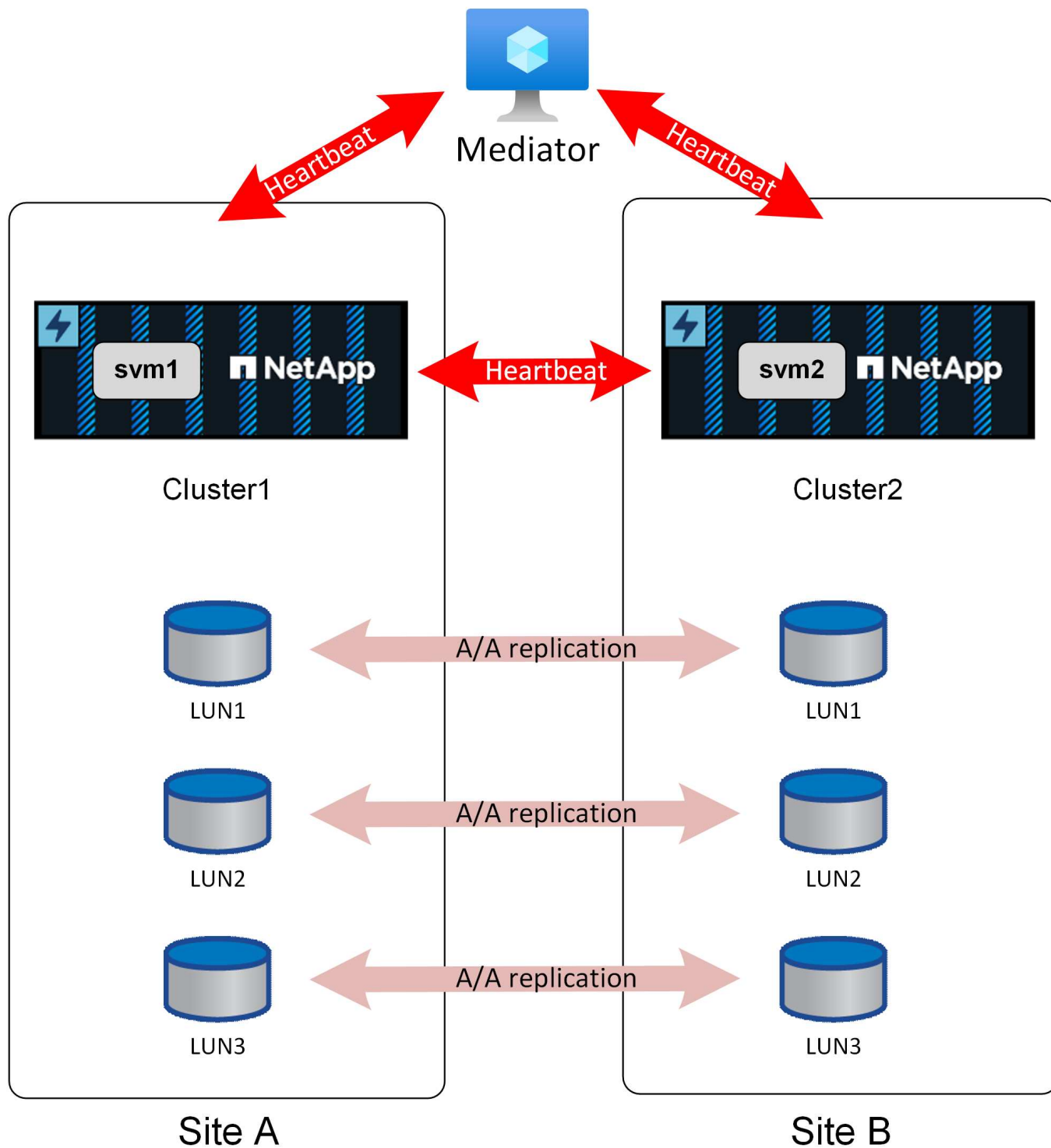
#### • ONTAP调解器\*\*

ONTAP调解器是从NetApp支持下载的软件应用程序、通常部署在小型虚拟机上。ONTAP调解器不是Tiebreaker。它是参与SnapMirror活动同步复制的两个集群的备用通信通道。ONTAP根据通过直接连接和调解器从合作伙伴处收到的响应来推动自动化操作。



要安全地自动执行故障转移、需要使用调解器。理想情况下、它会放置在独立的第三个站点上、但如果与参与复制的集群之一主机代管、它仍可满足大多数需求。

调解器并不是真正的断路器、但它提供的功能实际上就是这样。它不会执行任何操作、而是为集群到集群的通信提供备用通信通道。



自动化故障转移的第一大挑战是脑裂问题、如果两个站点彼此断开连接、就会出现该问题。应该发生什么？您不

希望让两个不同的站点将自己指定为数据的无故障副本、但单个站点如何区分实际丢失相对站点与无法与相反站点通信之间的区别？

这是调解者进入画面的地方。如果放置在第三个站点上、并且每个站点都与该站点建立了单独的网络连接、则每个站点都有一条额外的路径来验证另一个站点的运行状况。再次查看上图、并考虑以下情形。

- 如果调解器发生故障或无法从一个或两个站点访问、会发生什么情况？
  - 两个集群仍可通过复制服务所使用的同一链路彼此通信。
  - 数据仍会提供RPO = 0保护
- 如果站点A发生故障、会发生什么情况？
  - 站点B将看到两个通信通道关闭。
  - 站点B将接管数据服务、但没有RPO = 0镜像
- 如果站点B发生故障、会发生什么情况？
  - 站点A将看到两个通信通道关闭。
  - 站点A将接管数据服务、但没有RPO = 0镜像

还需要考虑另一种情形：丢失数据复制链路。如果站点之间的复制链路丢失、显然无法执行RPO = 0镜像。那么应该发生什么呢？

这由首选站点状态控制。在SM-AS关系中、其中一个站点是另一个站点的二级站点。这对正常操作没有影响、并且所有数据访问都是对称的、但是如果复制中断、则必须断开连接才能恢复操作。结果是、首选站点将继续操作而不进行镜像、而二级站点将暂停IO处理、直到复制通信恢复为止。

首选站点

SnapMirror主动同步行为是对称的、但有一个重要例外-首选站点配置。

SnapMirror主动同步会将一个站点视为"源"、而将另一个站点视为"目标"。这意味着单向复制关系、但这不适用于IO行为。复制是双向的、对称的、镜像两端的IO响应时间相同。

该 `source` 名称用于控制首选站点。如果复制链路丢失、则源副本上的LUN路径将继续提供数据、而目标副本上的LUN路径将变得不可用、直到重新建立复制并使SnapMirror重新进入同步状态为止。然后、这些路径将恢复提供数据。

可通过SystemManager查看源/目标配置：

Relationships

Local destinationsLocal sources

SearchDownloadShow/hideFilter

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

或在命令行界面上：

```
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
```

关键在于源是位于第一个Storage Virtual Machine上的SVM。如上所述、术语"源"和"目标"并不表示复制的数据流。两个站点都可以处理写入并将其复制到相反站点。实际上、两个集群都是源和目标。将一个集群指定为源集群的效果只是控制在复制链路丢失时哪个集群作为读写存储系统继续存在。

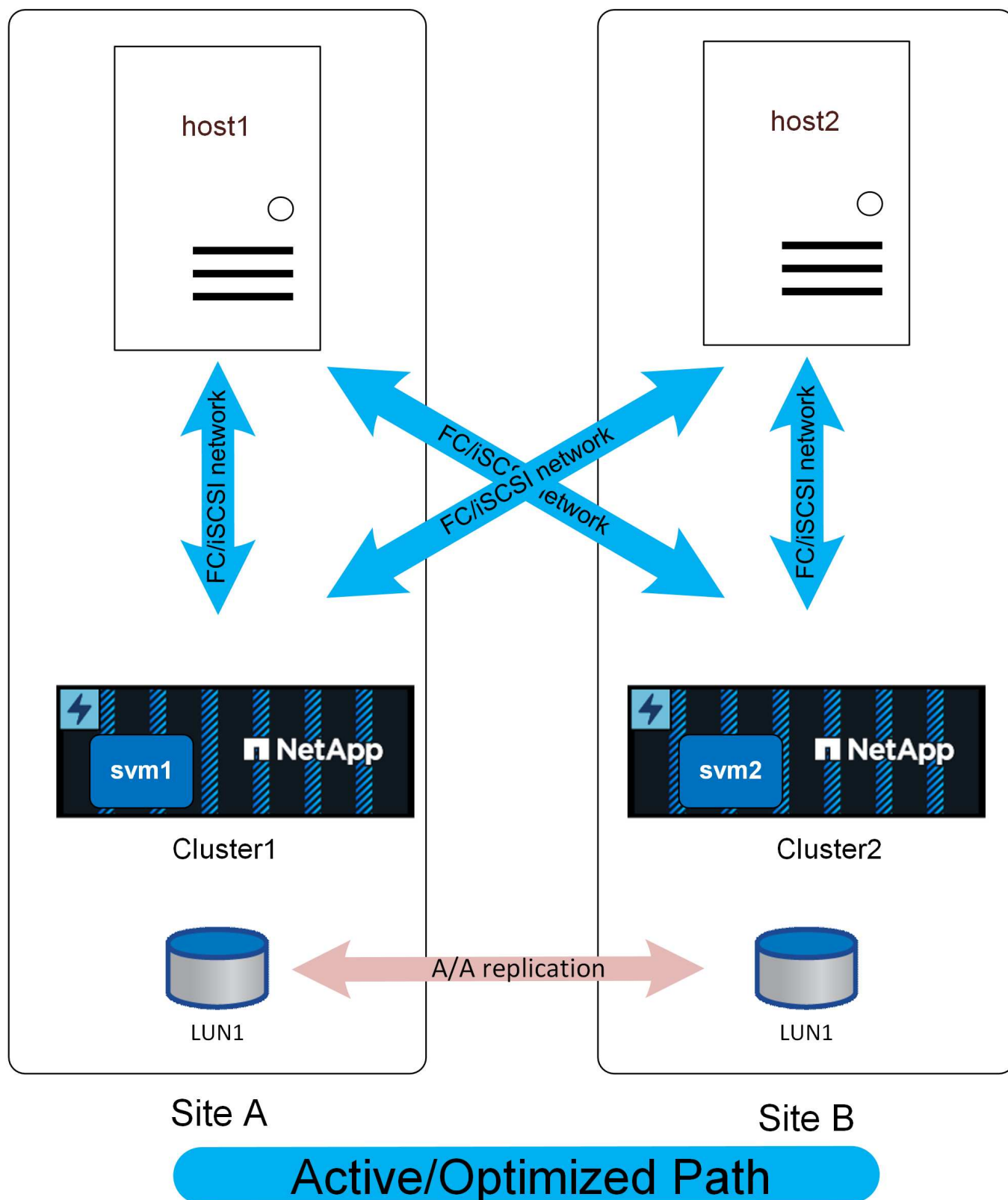
网络拓扑

统一访问

统一访问网络意味着主机能够访问两个站点(或同一站点中的故障域)上的路径。

SM-AS的一项重要功能是、可以对存储系统进行配置、使其知道主机所在的位置。将LUN映射到给定主机时、您可以指示它们是否接近给定存储系统。

NetApp ASA系统可在集群上的所有路径之间提供主动-主动多路径功能。这也适用于SM-AS配置。



使用统一访问时、IO将跨越WAN。这是一个全网状网络集群、在所有使用情形中、这可能是可取的、也可能不理想。

如果两个站点之间相距100米且具有光纤连接、则WAN上不会出现可检测到的额外延迟、但如果两个站点相距较远、则两个站点上的读取性能都会受到影响。使用非一致访问网络的ASA可以获得ASA的成本和功能优势、而不

会造成跨站点延迟访问损失、也不会使用主机邻近功能允许两个站点进行站点本地读/写访问。

在低延迟配置中使用SM-A的ASA具有两个有趣的优势。首先、从本质上说、它可以将任何一台主机的性能提高一倍、因为使用两倍路径的控制器可以为IO提供服务。其次、在单站点环境中、它可以提供极高的可用性、因为整个存储系统可能会丢失、而不会中断主机访问。

## 邻近设置

接近是指每个集群的配置、表示特定主机WWN或iSCSI启动程序ID属于本地主机。这是配置LUN访问的第二个可选步骤。

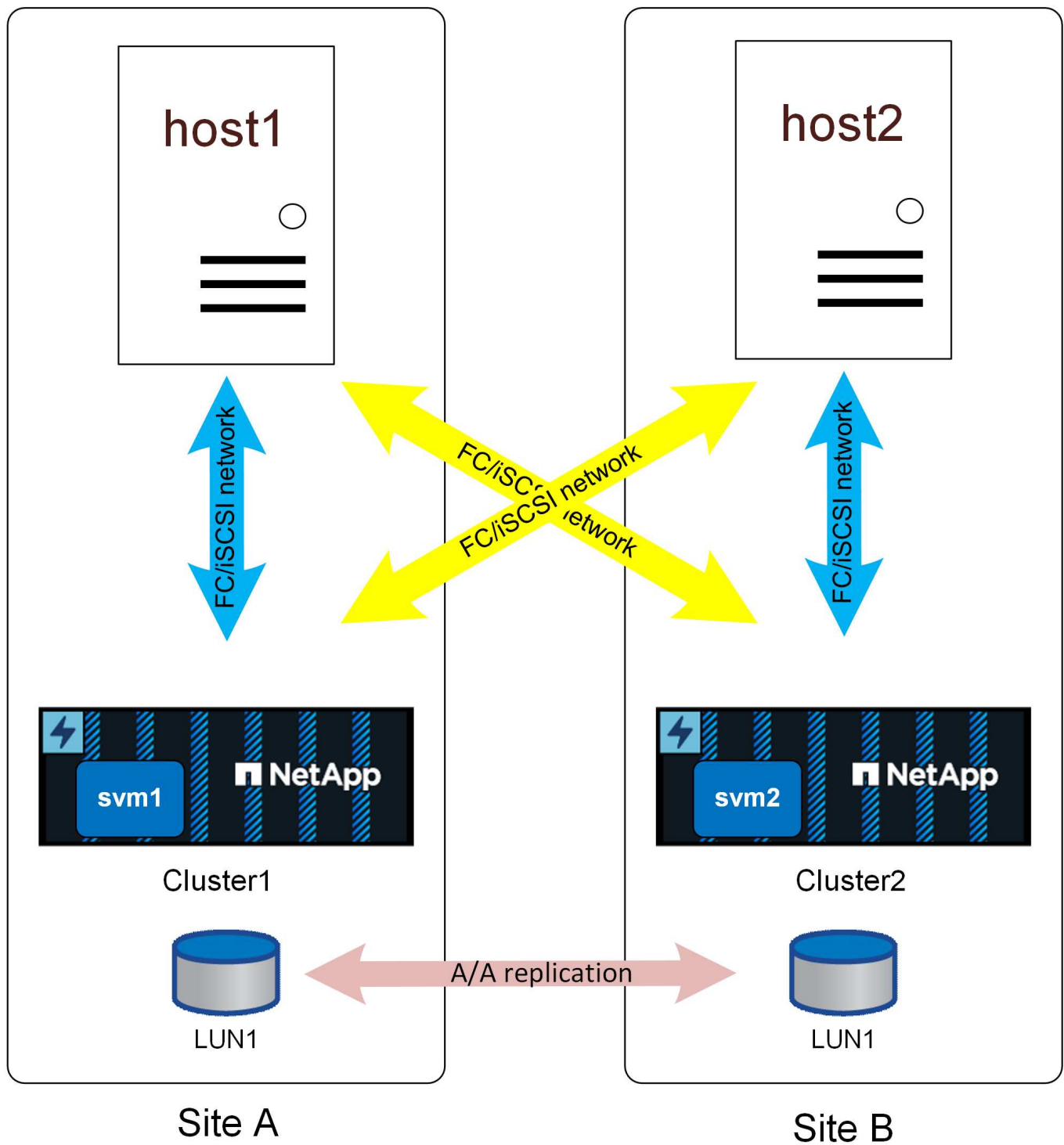
第一步是常规的igroup配置。每个LUN都必须映射到一个igroup、该igroup包含需要访问该LUN的主机的wwn/iSCSI ID。此选项用于控制哪个主机对LUN具有\_access\_访问权限。

第二个可选步骤是配置主机邻近性。这不控制访问、而是控制\_priority\_。

例如、可以将站点A的主机配置为访问受SnapMirror活动同步保护的LUN、并且由于SAN跨站点扩展、因此可以使用站点A上的存储或站点B上的存储为该LUN提供路径

如果没有邻近设置、则该主机将平等使用这两个存储系统、因为这两个存储系统都会公布主动/优化路径。如果站点之间的SAN延迟和/或带宽有限、则可能无法实现这一点、您可能希望确保在正常操作期间、每个主机优先使用指向本地存储系统的路径。这可通过将主机的wwn/iSCSI ID作为近端主机添加到本地集群来配置。可通过命令行界面或SystemManager完成此操作。

配置主机邻近性后、路径将如下所示。



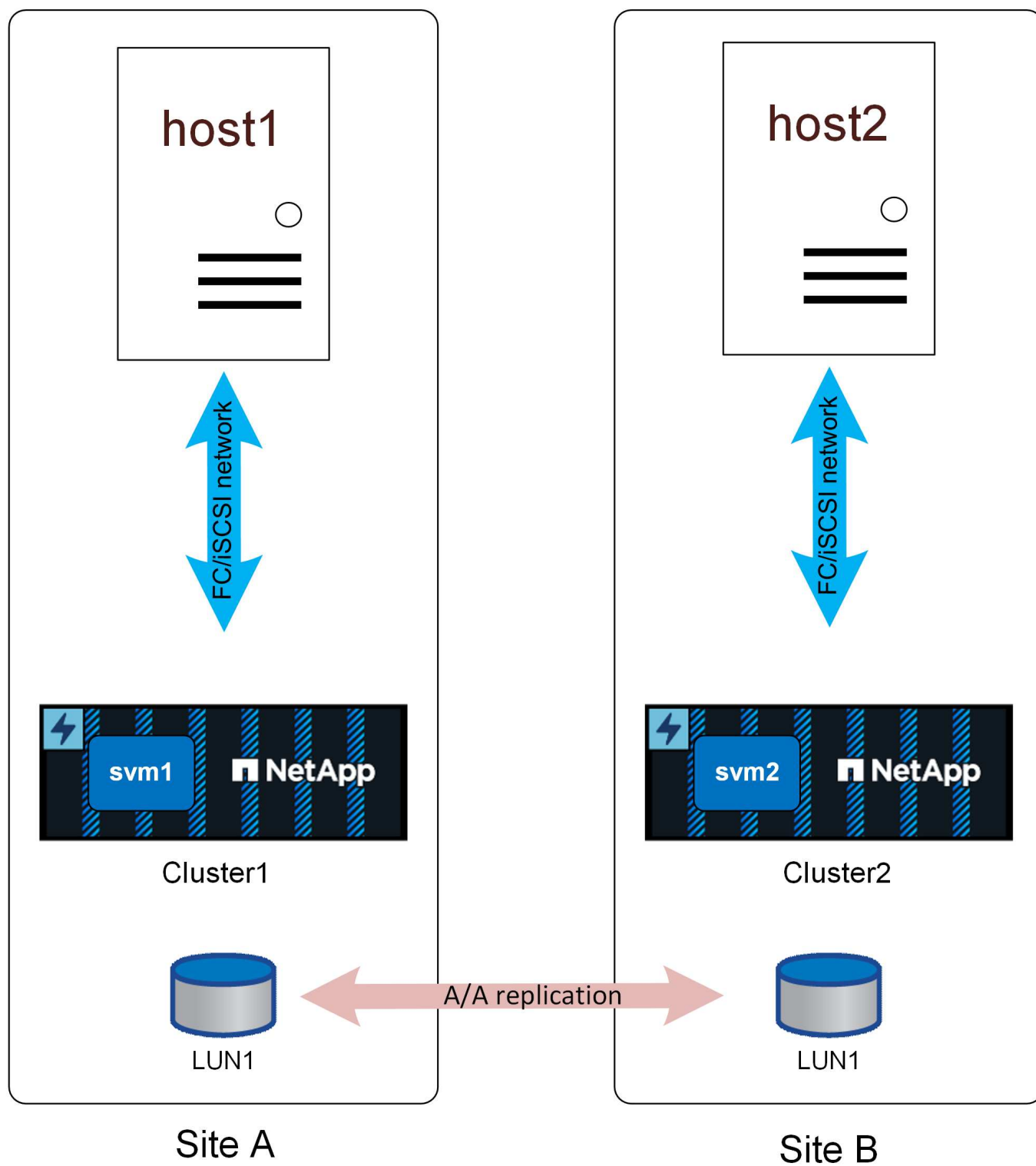
Active/Optimized Path

Active Path



## 非一致访问

非一致访问网络意味着每个主机只能访问本地存储系统上的端口。SAN不会跨站点(或同一站点内的故障域)进行扩展。



## Active/Optimized Path

这种方法的主要优势是SAN的精简性、您无需在网络上延伸SAN。某些客户的站点间连接延迟不足、或者缺少通过站点间网络传输FC SAN流量的基础架构。

非一致访问的缺点是、某些故障情形(包括丢失复制链路)将导致某些主机无法访问存储。如果本地存储连接丢

失、则作为单个实例运行的应用程序(例如、本质上仅在任何给定挂载的单个主机上运行的非集群数据库)将失败。数据仍会受到保护、但数据库服务器将无法再访问。它需要在远程站点上重新启动、最好是通过自动化过程重新启动。例如、VMware HA可以在一台服务器上检测到全路径关闭的情况、并在具有可用路径的另一台服务器上重新启动VM。

相比之下、Oracle RAC等集群应用程序可以提供在两个不同站点上同时提供的服务。丢失站点并不意味着整个应用程序服务都会丢失。实例仍可用、并且在正常运行的站点上运行。

在许多情况下、通过站点间链路访问存储的应用程序所产生的额外延迟开销是不可接受的。这意味着统一网络可用性的提高微乎其微、因为如果站点上丢失存储、则无论如何都需要关闭故障站点上的服务。

为了简单起见、这些图中未显示通过本地集群的冗余路径。ONTAP存储系统本身就是HA、因此控制器故障不应导致站点故障。它只会导致受影响站点上使用的本地路径发生更改。

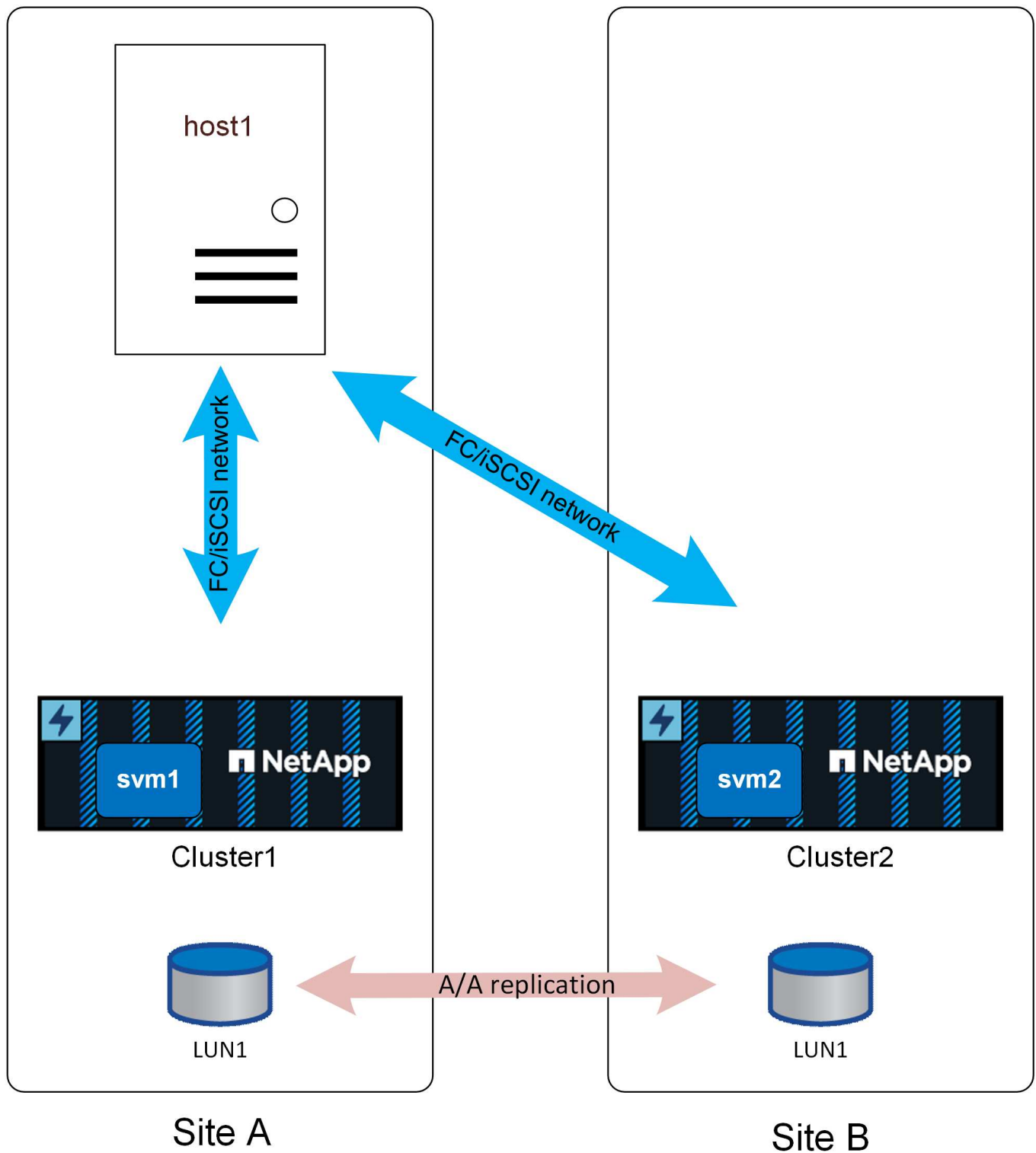
## 概述

可以将SQL Server配置为以多种方式与SnapMirror活动同步配合使用。正确答案取决于可用的网络连接、RPO要求和可用性要求。

### SQL Server的独立实例

文件布局和服务器的最佳实践与文档中建议的相同["基于ONTAP的SQL Server"](#)。

使用独立设置时、SQL Server只能在一个站点上运行。可能["统一"](#)会使用访问权限。



使用统一访问时、任一站点的存储故障都不会中断数据库操作。当然、如果包含数据库服务器的站点发生完全故障、则会导致中断。

某些客户可以为在远程站点上运行的操作系统配置预先配置的SQL Server设置、并使用与生产实例相同的构建版本进行更新。故障转移需要激活备用站点上的独立SQL Server实例、发现LUN并启动数据库。由于不需要从存储端执行任何操作、因此可以使用Windows PowerShell cmdlet自动完成整个过程。

"非一致性"也可以使用访问、但如果数据库服务器所在的存储系统因数据库没有可用的存储路径而出现故障、则

会导致数据库中断。在某些情况下、这种情况仍可接受。SnapMirror主动同步仍可提供RPO = 0的数据保护、并且在站点发生故障时、运行正常的副本将处于活动状态、并可使用与上述统一访问相同的过程恢复操作。

使用虚拟化主机可以更轻松地配置简单的自动化故障转移过程。例如、如果SQL Server数据文件与启动VMDK一起同步复制到二级存储、则在发生灾难时、可以在备用站点上激活整个环境。管理员可以在正常运行的站点上手动激活主机、也可以通过VMware HA等服务自动执行此过程。

## SQL Server故障转移集群实例

SQL Server故障转移实例也可以托管在作为子操作系统运行在物理服务器或虚拟服务器上的Windows故障转移集群上。这种多主机架构可提供SQL Server实例和存储故障恢复能力。在需要在保持增强性能的同时实现强大故障转移流程的高需求环境中、此类部署非常有用。在故障转移集群设置中、当主机或主存储受到影响时、SQL服务将故障转移到二级主机、同时、二级存储将可用于提供IO。无需自动化脚本或管理员干预。

### 故障情形

要规划完整的SnapMirror主动同步应用程序架构、需要了解SM-AS如何在各种计划内和计划外故障转移场景中做出响应。

在以下示例中、假设站点A已配置为首选站点。

### 复制连接丢失

如果SM-AS复制中断、则无法完成写入IO、因为集群无法将更改复制到相反站点。

### 站点A (首选站点)

首选站点上的复制链路故障会导致写入IO处理暂停大约15秒、因为ONTAP会在确定复制链路确实无法访问之前重试复制的写入操作。15秒后、站点A系统将恢复读取和写入IO处理。SAN路径不会更改、LUN将保持联机状态。

### 站点 B

由于站点B不是SnapMirror主动同步首选站点、因此其LUN路径将在大约15秒后变得不可用。

### 存储系统故障

存储系统故障的结果与丢失复制链路的结果几乎相同。正常运行的站点应出现大约15秒的IO暂停。15秒过后、IO将照常在该站点上恢复。

### 调解器丢失

调解器服务不直接控制存储操作。它可用作集群之间的备用控制路径。它主要用于自动执行故障转移、而不存在脑裂情况的风险。在正常操作下、每个集群都会将更改复制到其配对集群、因此、每个集群都可以验证配对集群是否联机并提供数据。如果复制链路失败、复制将停止。

安全自动故障转移需要调解器的原因是、否则存储集群将无法确定双向通信丢失是网络中断还是实际存储故障所致。

调解器为每个集群提供一个备用路径、以验证其配对集群的运行状况。具体情形如下：

- 如果集群可以直接与其配对集群联系、则复制服务将正常运行。无需执行任何操作。

- 如果首选站点无法直接或通过调解器与其配对站点联系、则会假定配对站点实际不可用或已隔离、并且其LUN路径已脱机。然后、首选站点将继续释放RPO = 0状态、并继续处理读写IO。
- 如果非首选站点无法直接与其配对站点联系、但可以通过调解器与其联系、则它会使其路径脱机、并等待复制连接返回。
- 如果非首选站点无法直接联系其配对站点或无法通过操作调解器联系其配对站点、则会假定配对站点实际不可用或已隔离、并且其LUN路径已脱机。然后、非首选站点将继续释放RPO = 0状态、并继续处理读写IO。它将承担复制源的角色、并成为新的首选站点。

如果调解器完全不可用：

- 复制服务因任何原因发生故障(包括非首选站点或存储系统发生故障)、都会导致首选站点释放RPO = 0状态并恢复读写IO处理。非首选站点将使其路径脱机。
- 首选站点发生故障将导致中断、因为非首选站点无法验证对等站点是否真正脱机、因此非首选站点无法安全地恢复服务。

正在还原服务

解决故障(例如、还原站点间连接或启动故障系统)后、SnapMirror活动同步端点将自动检测是否存在故障复制关系、并将其恢复为RPO = 0状态。重新建立同步复制后、故障路径将再次联机。

在许多情况下、集群模式应用程序会自动检测故障路径的返回情况、这些应用程序也会恢复联机。在其他情况下、可能需要进行主机级SAN扫描、或者可能需要手动将应用程序恢复联机。它取决于应用程序及其配置方式、通常、此类任务可以轻松实现自动化。ONTAP本身具有自我修复能力、不需要任何用户干预即可恢复RPO = 0存储操作。

手动故障转移

更改首选站点只需简单的操作即可。在集群之间切换复制行为的权限时、IO将暂停一两秒钟、但IO不会受到影响。

## 数据安全

确保SQL Server数据库环境的安全是一项多方面的工作、不仅限于管理数据库本身。ONTAP提供了多种独特的功能、旨在保护数据库基础架构的存储方面。

### Snapshot 副本

存储快照是目标数据的时间点副本。ONTAP的实施支持设置各种策略、并为每个卷存储多达1024个快照。ONTAP中的快照可节省空间。只有在原始数据集发生更改时、才会占用空间。它们也是只读的。快照可以删除、但不能更改。

在某些情况下、可以直接在ONTAP上计划快照。在其他情况下、可能需要SnapCenter等软件来编排应用程序或操作系统操作、然后再创建快照。无论哪种方法最适合您的工作负载、主动快照策略都可以通过频繁、轻松访问从启动LUN到任务关键型数据库的所有备份来确保数据安全。



ONTAP灵活卷或更简单地说、卷与LUN并不同义。卷是文件或LUN等数据的管理容器。例如、数据库可能位于一个8 LUN条带集上、所有LUN都包含在一个卷中。

有关快照的详细信息、请参见["ONTAP文档。"](#)



## 防篡改快照

从ONTAP 9.12.1开始、快照不仅是只读的、还可以防止意外或故意删除。此功能称为防篡改快照。可以通过Snapshot策略设置和实施保留期限。生成的快照在达到到期日期之前无法删除。不会覆盖管理或支持中心。

这样可以确保即使侵入者、恶意内部人员甚至勒索软件攻击导致访问ONTAP系统本身、也不会影响备份。如果与频繁的快照计划结合使用、则可以提供极其强大的数据保护、并且RPO非常低。



不能使用光纤池对防篡改快照进行分层。

有关防篡改快照的详细信息、请参见["ONTAP文档。"](#)



在新的ASA平台中、使用\_vault\_policy将快照复制到远程集群、然后锁定目标以使快照不受篡改。

## SnapMirror 复制

也可以将快照复制到远程系统。这包括防篡改快照、即在远程系统上应用和实施保留期限。结果是、数据保护优势与本地快照相同、但数据位于另一个存储阵列上。这样可以确保销毁原始阵列不会影响备份。

第二个系统还为管理安全性打开了新的选项。例如、某些NetApp客户会隔离主存储系统和二级存储系统的身份验证凭据。没有一个管理用户可以同时访问这两个系统、这意味着恶意管理员无法删除所有数据副本。

有关SnapMirror的详细信息、请参见["ONTAP文档。"](#)

## Storage Virtual Machine

新配置的ONTAP存储系统与新配置的VMware ESX服务器类似、因为在创建虚拟机之前、这两个系统都不能支持任何用户。通过ONTAP、您可以创建一个Storage Virtual Machine (SVM)、此SVM将成为存储管理的最基本单元。每个SVM都有自己的存储资源、协议配置、IP地址和FCP WWN。这是ONTAP多租户的基础。

例如、您可以为关键生产工作负载配置一个SVM、为开发活动配置另一个SVM。然后、您可以限制某些管理员对生产SVM的访问、同时授予开发人员对开发SVM中存储资源的更广泛的控制权限。您可能还需要为财务和人力资源团队提供第三个SVM、以存储非常关键的只读数据。

有关SVM的详细信息、请参见["ONTAP文档。"](#)

## 管理RBAC

ONTAP为管理登录提供了功能强大的基于角色的访问控制(Role-Based Access Control、RBAC)。某些管理员可能需要完全集群访问权限、而其他管理员则可能只需要访问某些SVM。高级帮助台人员可能需要增加卷大小的能力。这样、您就可以为管理用户授予执行其工作职责所需的访问权限、而无需执行任何其他操作。此外、您还可以使用不同供应商提供的PKI来保护这些登录、限制对ssh密钥的访问、并强制执行失败登录尝试锁定。

有关管理访问控制的详细信息、请参见["ONTAP文档。"](#)

## 多因素身份验证(MFA)

ONTAP和某些其他NetApp产品现在支持使用多种方法进行多因素身份验证(MFA)。其结果是、用户名/密码被泄露本身并不是没有第二个因素(例如FOB或智能手机应用程序)的数据的安全线程。

有关多因素身份验证(MFA)的详细信息、请参见["ONTAP文档。"](#)

## API RBAC

自动化需要API调用、但并非所有工具都需要完全管理访问权限。为了帮助保护自动化系统的安全、还在API级别提供了RBAC。您可以将自动化用户帐户限制为所需的API调用。例如、监控软件不需要更改访问权限、只需要读取访问权限即可。配置存储的工作流不需要删除存储。

有关API RBAC的详细信息、请参见["ONTAP文档。"](#)

## 多管理员验证(MAV)

通过要求两个不同的管理员(每个管理员都有自己的凭据)批准某些活动、可以进一步实现多"因素"身份验证。其中包括更改登录权限、运行诊断命令和删除数据。

有关多管理员验证(MAV)的详细信息、请参见["ONTAP文档。"](#)

## 版权信息

版权所有 © 2026 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

## 商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。