



Technical Report

Best Practice Guide for Microsoft SQL Server with NetApp EF-Series

Mitch Blackburn, Pat Sinthusan, NetApp
January 2014 | TR-4259

Abstract

This best practice guide is intended for storage administrators and database administrators to help them successfully deploy Microsoft® SQL Server® 2012 on NetApp® EF-Series storage.

TABLE OF CONTENTS

1	Executive Summary.....	4
1.1	Intended Audience	4
1.2	Caveats.....	4
2	Introduction to EF-Series.....	4
2.1	EF-Series Hardware Overview.....	4
2.2	SANtricity	6
2.3	EF-Series Volume Group.....	7
2.4	EF-Series Volume	8
3	Provisioning the EF-Series.....	8
3.1	Provision EF-Series Storage Using SANtricity 11.10 GUI.....	9
3.2	Provision EF-Series Storage Using SANtricity 11.10 CLI	15
3.3	Windows Volume Mount Points	19
4	Provisioning SQL Server 2012	19
4.1	SQL Server Database Files	19
4.2	Database Files for tempdb.....	20
5	High Availability	21
5.1	EF-Series and SANtricity 11.10	21
5.2	SQL Server High-Availability Options	23
6	Sizing	25
6.1	EF-Series I/O Overview	25
6.2	SQL Server I/O Overview.....	26
6.3	Estimating I/O	27
7	Monitoring	30
7.1	EF-Series Performance Monitoring Using NetApp Storage Explorer Plug-In for Microsoft SQL Server Management Studio.....	30
7.2	EF-Series Performance Monitoring Using SANtricity 11.10	36

Summary	38
References	40
Version History	40

LIST OF TABLES

Table 1) EF-Series performance comparison.....	5
Table 2) Transaction latency versus used storage.	24
Table 3) Comparison of usable capacity for differing RAID levels.....	26

LIST OF FIGURES

Figure 1) EF-Series EF550 with 16GB FC host interface option.	6
Figure 2) SANtricity disk structure.	8
Figure 3) EF550 average response time on 75% 8KB random read for differing RAID levels.	9
Figure 4) OLTP database comparison: latency.	19
Figure 5) EF-Series HA using Snapshot technology and mirroring.	23
Figure 6) Sample architecture design using AlwaysOn Availability Groups to synchronize database transactions between EF-Series and FAS storage.	25
Figure 7) E-Series performance sizing tool.	28
Figure 8) Input fields for EF-Series performance sizing.	29
Figure 9) Output from EF-Series performance sizing tool.	30

1 Executive Summary

In most online transaction processing (OLTP) systems, the processor, memory, and I/O subsystem in a server are well balanced and are not considered to be performance bottlenecks. The major source of performance issues in OLTP environments is typically related to the storage I/O activity. The speed of traditional hard disk drive (HDD)–based storage systems does not match the processing capabilities of the servers.

As a result, a situation often occurs where a powerful processor sits idle, waiting for the storage I/O requests to complete, negatively affecting user and business productivity. The negative effect on productivity extends the time to return on investment (ROI) and increases overall total cost of ownership (TCO). Therefore, storage IOPS performance and latency become strategic considerations for business. It is critical to make sure that the response time goals are met and performance optimization is realized for other system resources (processor and memory).

The NetApp EF-Series Flash Array provides a robust platform for delivering exceptional performance of 330,000 IOPS for the EF540 and 450,000 IOPS for the EF550 Flash Array, both with submillisecond response times to mission-critical applications. The EF-Series Flash Array leverages the latest technology in solid-state disk (SSD) along with a strong heritage of the ability to handle diverse workloads to provide superior business value through the acceleration of latency-sensitive and high-I/O environments.

The EF-Series, available with up to 96TB of raw SSD capacity, provides capacity and bullet-proof reliability to meet the requirements of the most demanding organizations. This technical report provides an overview of best practices for Microsoft SQL Server 2012 with NetApp EF-Series.

1.1 Intended Audience

This technical report is intended for NetApp employees, partners, and NetApp field personnel who are responsible for deploying such a solution in a customer environment. It is assumed that the reader is familiar with the various components of the solution.

1.2 Caveats

- This document assumes that the database is either being relocated to an EF-Series storage system or being created on an EF-Series storage system to achieve high performance.
- This document assumes that the user wants to improve performance of an OLTP application.

2 Introduction to EF-Series

2.1 EF-Series Hardware Overview

The EF-Series Flash Array continues NetApp's long-standing heritage of delivering powerful solutions to meet unique business needs. With more than 400,000 sustained IOPS and submillisecond response times, the EF-Series enables business-critical applications to deliver faster results and improve customer experience.

This combination of high IOPS and ultra-low latency makes an EF-Series Flash Array a great choice for database-driven applications requiring a dedicated extreme performance solution.

Table 1 provides a comparison of the EF540 and EF550 Flash Arrays.

Table 1) EF-Series performance comparison.

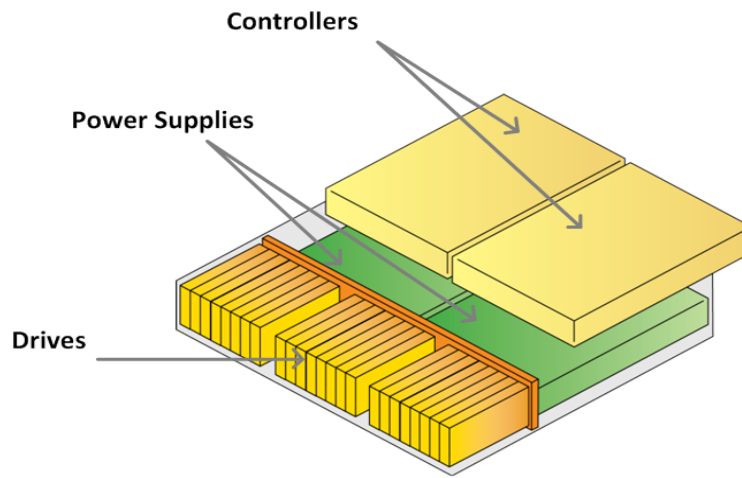
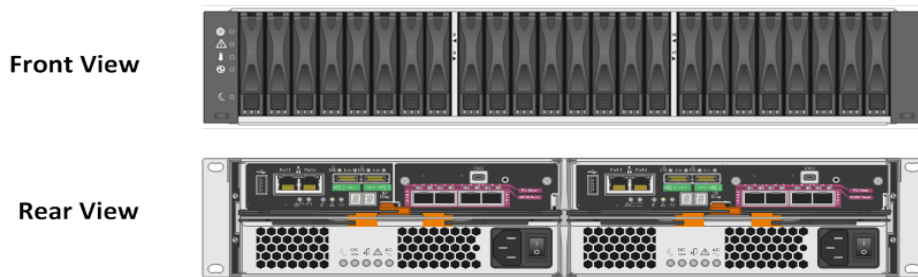
Performance Comparison	EF540	EF550
Burst I/O rate: cache reads (512KB)	800,000 IOPS	900,000 IOPS
Sustained I/O rate: random disk reads (4KB)	300,000 to 330,000 IOPS	400,000 to 450,000 IOPS
Sustained I/O rate: random disk writes (4KB)	65,000 IOPS	75,000 IOPS
Sustained throughput: sequential disk reads (512KB)	6GB	12GB
Sustained throughput: sequential disk write (cache mirror disabled 512KB)	4GB	10GB
Sustained throughput: sequential disk write (cache mirror enabled 512KB)	3GB	6GB

The EF-Series Flash Array is a fully redundant 2U/24 system that supports up to four additional 2U trays of SSD drives for up to 96TB of raw capacity.

The EF-Series has dual active controllers, four 6Gb SAS disk expansion ports, and 24GB of battery-backed controller cache and runs on the performance-optimized and enterprise-proven SANtricity[®] software.

As illustrated in Figure 1, the EF-Series supports dual controller canisters, power supplies, and fan units for hardware redundancy. The shelves are sized to support up to 24 2.5" disk drives in 2U of rack space.

Figure 1) EF-Series EF550 with 16GB FC host interface option.



Along with performance, the key to maximizing value is to maximize efficiency. Historically, companies have sacrificed efficiency to achieve extreme performance levels by overprovisioning their storage. But that is changing. The all-flash EF-Series helps customers balance performance and efficiency by eliminating overprovisioning, thereby dramatically reducing costs.

With the performance of more than 1,000 traditional drives, a single EF-Series can meet extreme requirements with 95% less rack space, power, and cooling. This is a significant benefit to customers who are used to deploying partially filled disks for improving application performance.

In addition to cost efficiency, the EF-Series Flash Array provides application efficiency. By completing a higher volume of application operations, customers can become efficient and drive better results.

The EF-Series has a fully redundant I/O path that provides automated failover, which is surprisingly not given in many of the flash products out there today, but it is an absolute requirement for the enterprise accounts looking to implement this type of technology.

All management tasks are performed while the EF-Series remains online with complete read/write data access. This allows storage administrators to make configuration changes and conduct maintenance without disrupting application I/O.

The EF-Series Flash Array also offers advanced data protection common to enterprise storage to protect against data loss and downtime events, both locally with Snapshot™ technology and remotely with synchronous and asynchronous replication.

2.2 SANtricity

SANtricity is the GUI management interface for the EF-Series that is based on the Java® framework. SANtricity can be installed on Windows® or Linux® operating systems. The management application must

be installed on a management node that does not participate in production data delivery. The software is available in 32-bit and 64-bit versions, and the install process detects if installation of the package was performed on the incorrect OS version.

SANtricity client software can be installed on Windows or Linux operating systems for an out-of-band management of the storage array. In this configuration, the host agent functionality for in-band management does not function, and the number of client connections is limited to eight. To manage the storage arrays by using in-band connections, the management client must be run on a server OS and have Fibre Channel (FC) connectivity to all arrays. The number of client connections being limited to eight for out-of-band management does not apply to the in-band management.

To create volume groups on flash arrays, the first step while configuring SANtricity is to assign a redundant array of inexpensive disks (RAID) level. This assignment is then applied to the disks selected to form the volume group.

Flash arrays support RAID levels 0, 1, 10, 3, 5, and 6 or Dynamic Disk Pools (DDP).

To simplify the process of provisioning storage, NetApp recommends using the SANtricity automatic configuration feature. The configuration wizard analyzes the available disk capacity on the array and selects disks that maximize array performance and fault tolerance while meeting capacity requirements, hot spares, and any other criteria specified in the wizard.

2.3 EF-Series Volume Group

The volume group is the top-level unit of storage in the EF-Series. When a storage array is deployed, the first step in presenting the available disk capacity to various hosts is to create:

- The number of disks required to meet performance requirements
- The desired level of RAID protection to meet specific business requirements
- Volume groups with sufficient capacity

Capacity planning is dependent on detailed customer input and discovery; however, protection and performance planning is a standardized implementation practice.

For example, the EF-Series Flash Arrays support multiple RAID levels, and each RAID level provides standardized functionality with the associated best practices. One of these best practices includes disk selection criteria to achieve disk-level, drawer-level, and shelf-level protection from common disk failure scenarios.

When selecting disks to create a volume group, administrators follow a standard pattern that uses both controller channels and spreads the disks across shelves and drawers in the configuration. This method does not offer a level of data protection for RAID level 0 volume groups; however, it does establish the disk selection pattern of allocating disks for RAID levels that offers protection against single- and double-disk fault scenarios.

Size the volume groups to meet business requirements; however, large volume groups that use RAID levels 1, 3, 5, 6, or 10 require a significant number of hours to complete the reconstruction process from a failed disk. The larger the volume group, the longer the reconstruction time.

NetApp SANtricity E-Series DDPs simplify RAID management, improve data protection, and maintain predictable performance under all conditions by using next-generation technology to minimize the performance impact of drive failures and return the system to optimal conditions up to eight times more quickly than traditional RAID technology. With shorter rebuild times and patented prioritization reconstruction technology, DDP significantly reduces exposure to numerous cascading disk failures and offers a level of data protection that cannot be achieved with the traditional RAID. Flash Arrays support T10PI data assurance technology. During the volume group creation process, the arrays also use immediate availability format (IAF) to initialize disks. When the T10PI technology is integrated with the IAF disk initialization method, initialization cycles might last longer than 24 hours.

To create a volume group, storage administrators should consider the following:

- The reconstruction time, especially for business-critical and high-availability applications.

The availability of hot-spare disks that meets the following requirements:

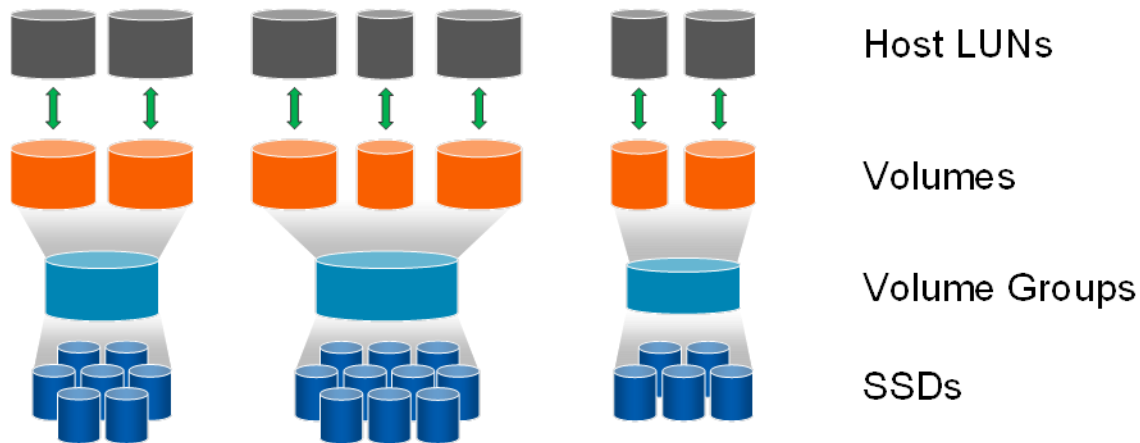
- The disk type must match the disks being protected.
- Full disk encryption (FDE)–enabled disks can protect non–FDE enabled disks; however, the reverse is not true.
- Disk capacity must exceed the used capacity of the protected disks.
- Hot spares are global; protection is extended to all assigned disks in the array regardless of volume group assignments.
- There must be a sufficient quantity of spare disks to protect multiple volume groups based on the business-critical nature of the groups.

For additional information about setting up volume groups, refer to the SANtricity online help documentation.

2.4 EF-Series Volume

As shown in Figure 2, a volume is the logical storage entity created for a host to access disks on the storage array. A volume is created from the capacity available on a volume group. Although a volume might include more than one drive, a volume appears as one logical entity to the host. The volume is presented to the host as disk capacity in the form of a LUN.

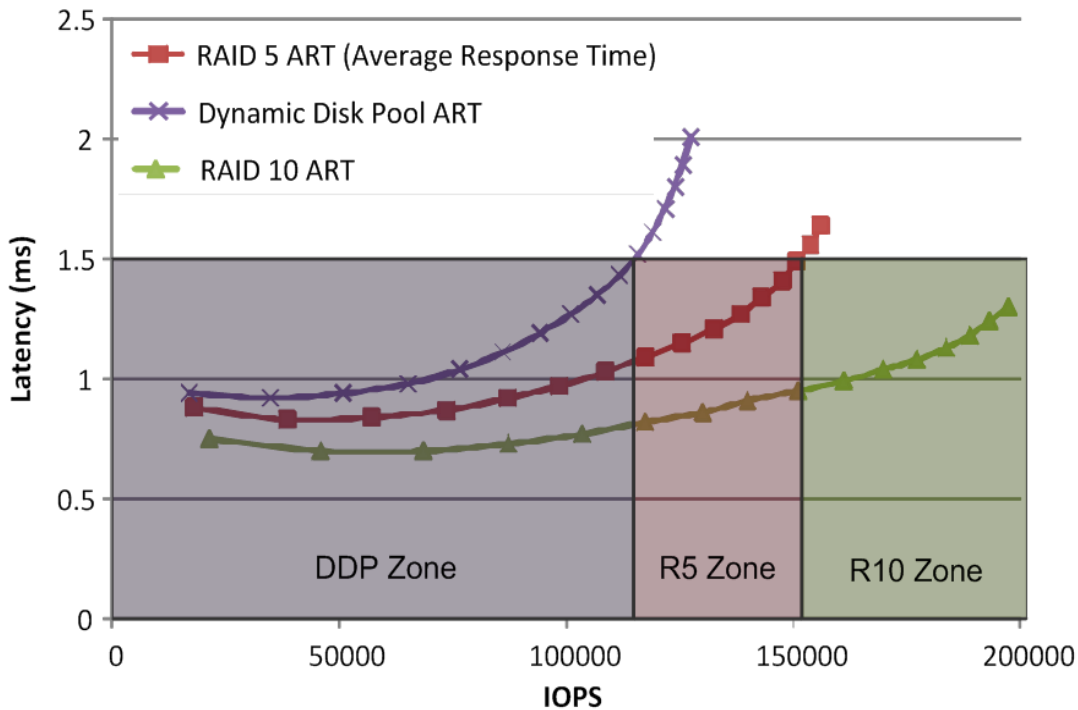
Figure 2) SANtricity disk structure.



3 Provisioning the EF-Series

Looking at EF550 performance for a workload that approximates a SQL Server OLTP database, the RAID level chosen for the volume group has a significant effect on performance. Figure 3 shows the effect on performance for RAID 5, DDP, and RAID 10. As expected, RAID 10 outperforms RAID 5, which in turn outperforms DDP for this workload. In this example, if a customer needed 1.5ms response time and had a requirement for 120,000 IOPS, we would suggest using RAID 5 or RAID 10. However, if the performance requirement was 80,000 IOPS, DDPs would be the best configuration.

Figure 3) EF550 average response time on 75% 8KB random read for differing RAID levels.



To achieve maximum performance, NetApp recommends using RAID 10 for all volume groups. If space is limited, application data files and tempdb data files should be on RAID 10 volume groups and database log files stored on a RAID 5 volume group.

3.1 Provision EF-Series Storage Using SANtricity 11.10 GUI

To create a volume group using SANtricity from unconfigured capacity in the storage system, complete the following steps:

1. In the Array Management window (AMW), click the Hardware tab and verify that the required number of hot spare drives has been allocated.

Note: For additional information on configuring hot spare drives, refer to the SANtricity in application help section Using Hot Spare Drives.

SANtricity®

Storage Array Storage Copy Services Host Mappings Hardware Monitor Upgrade Help

EF550 Optimal

Summary Performance Storage & Copy Services Host Mappings Hardware **Setup**

Drive type:
SAS,DA Drive, 512 bytes, SSD, 744.712 GB

Tray 99 (front) [View Tray Components](#)

Tray 99 (back)

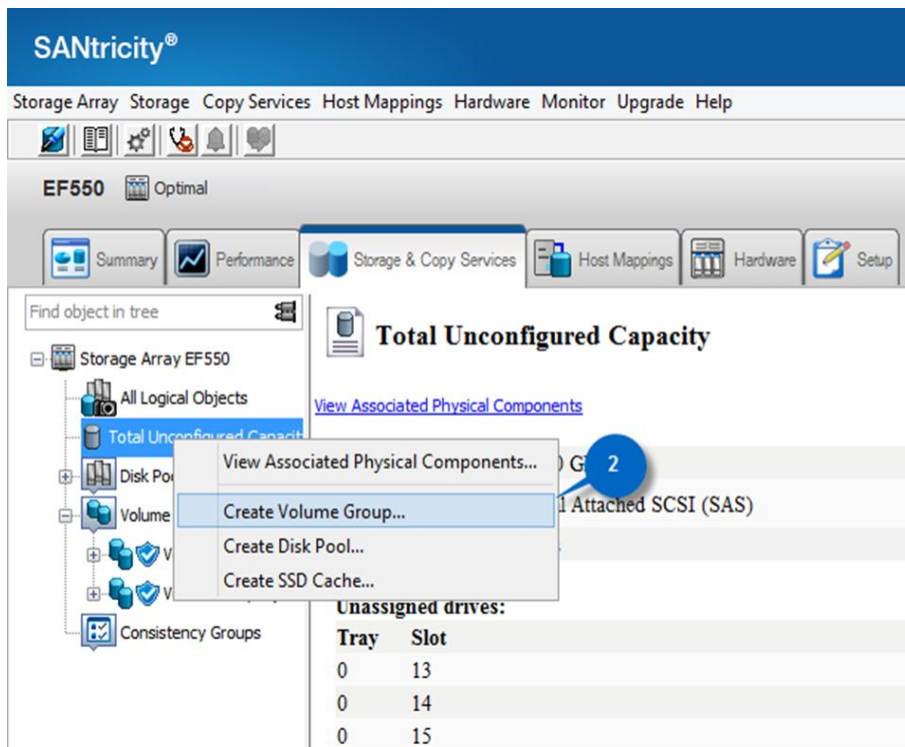
Tray 0 (front) [View Tray Components](#)

Controller in Tray 99, Slot A

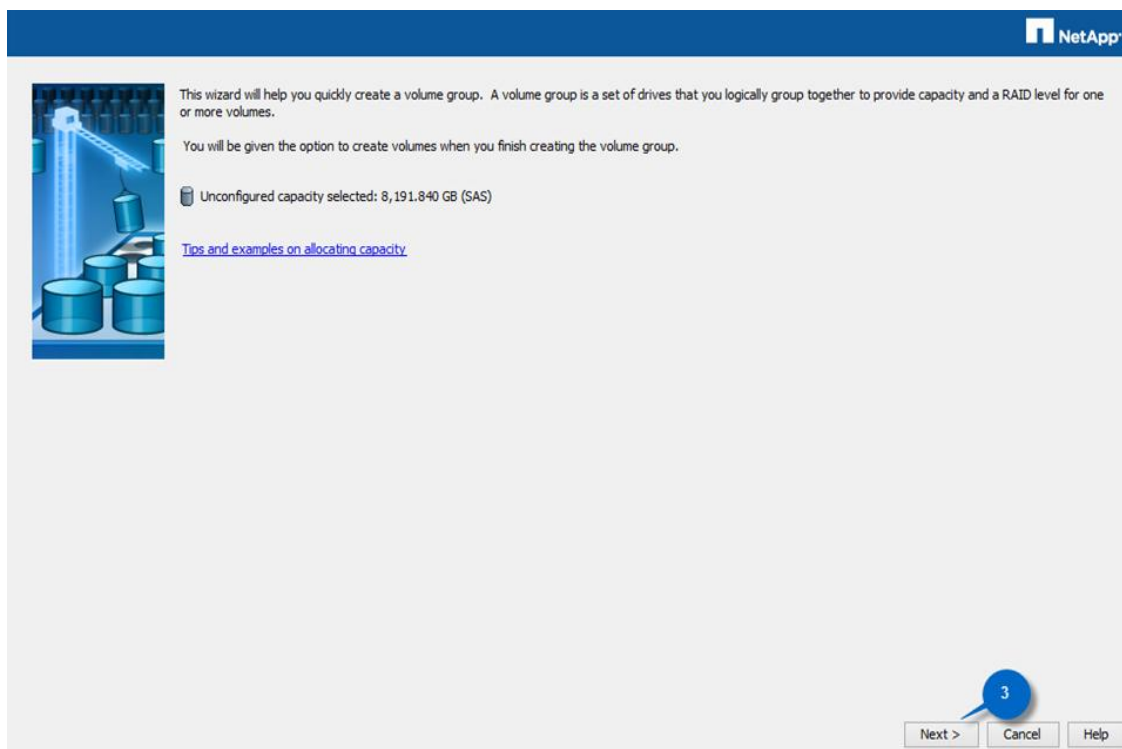
[View Complete Controller Properties](#)

Base Controller Properties	
Status:	Online
Current configuration	
Firmware version:	08.10.01.00
Appware version:	08.10.01.00
Bootware version:	08.10.01.00
NVSRAM version:	N5501-810834-DB1
Pending configuration	
Firmware version:	None
Appware version:	None
Bootware version:	None
NVSRAM version:	None
Transferred on:	None
Model name:	5504
Board ID:	5501
Submodel ID:	201
Product ID:	INF-01-00
Revision:	0810
Replacement part number:	A100000
Part number:	111-01313
Serial number:	SV25027916
Vendor:	NETAPP
Date of manufacture:	April 8, 2013
Trunking supported:	No
Data Cache	
Total present:	10240 MB
Total used:	8192 MB
Data Cache Module	
Status:	Optimal
Location:	Controller A, Slot 1
Capacity:	4,096 MB
Replacement Part number:	Not Available

- In the AMW, select the Storage & Copy Services tab, right-click Total Unconfigured Capacity, and select Create Volume Group.



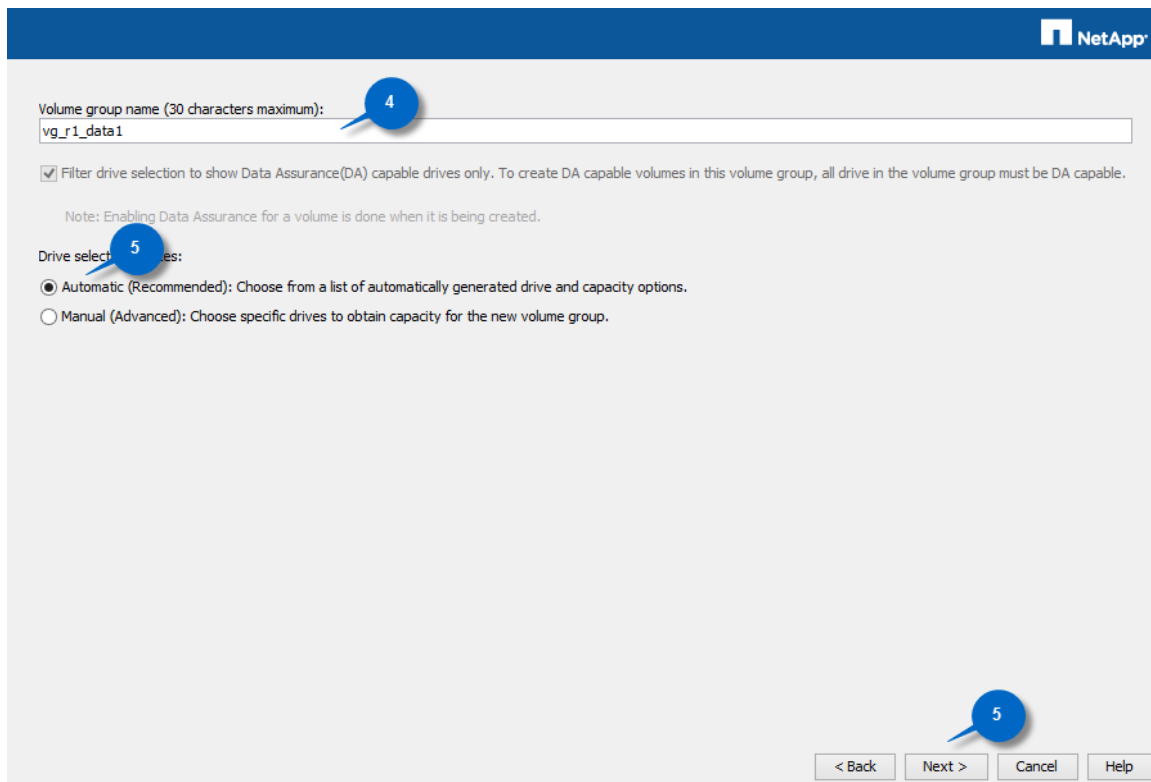
- On the Welcome page, click Next.



- Enter a volume group name that will aid in the management of the environment overtime.

Note: Volume group names must not exceed 30 characters and cannot contain spaces. The name string can contain letters, numbers, underscores (_), dashes (-), and pound signs (#).


5. To create a volume group automatically, select Automatic (Recommended) and click Next.



The screenshot shows the NetApp volume group creation wizard. At the top right is the NetApp logo. The main form area has a blue header bar. Below the header, there is a text input field for 'Volume group name (30 characters maximum):' with the value 'vg_r1_data1'. A blue callout bubble with the number '4' points to this field. Below the input field is a checkbox labeled 'Filter drive selection to show Data Assurance(DA) capable drives only. To create DA capable volumes in this volume group, all drive in the volume group must be DA capable.' which is checked. Below this is a note: 'Note: Enabling Data Assurance for a volume is done when it is being created.' Below the note is a section titled 'Drive selection:' with a blue callout bubble with the number '5' pointing to it. Under 'Drive selection:', there are two radio button options: 'Automatic (Recommended): Choose from a list of automatically generated drive and capacity options.' which is selected, and 'Manual (Advanced): Choose specific drives to obtain capacity for the new volume group.' At the bottom right of the form are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'. A blue callout bubble with the number '5' points to the 'Next >' button.

6. Select the desired RAID level from the drop-down list. For database and tempdb files, NetApp recommends using RAID 10.
Note: RAID 1 or “disk mirroring” offers high performance and the best data availability. Select four or more drives to achieve mirroring and striping (RAID 10 or RAID 1+0). Usable capacity is half of the drives in the volume group.
7. Select the desired volume group configuration from the list of available configurations and click Finish.

- g. Click Finish to create the new volume.



[Tips on storage provisioning](#)

Volume Parameters

Volume group name: vg_r1_data1
Volume group RAID level: 10
Free capacity: 1,489,421 GB

New volume capacity: Units: 9 a
1.200 TB 9 b

Volume name: 9 b
v_sqldata1

Map to host: 9 c
Map Later

Quality of Service (QoS) Attributes

☒ Enable data assurance (DA) protection on the new volume 9 d
☐ Use SSD cache 9 e

Volume I/O characteristics type: Custom 9 e

☐ Enable dynamic cache read prefetch 9 f

Segment size: 128 KB 9 f

9 g

Finish Cancel Help

10. From the Storage & Copy Services tab, confirm that the new volume group is displayed in the storage system tree and that the new volume is branching from the new volume group.

The screenshot displays the SANtricity 11.10 web interface. The top navigation bar includes links for Storage Array, Storage, Copy Services, Host Mappings, Hardware, Monitor, Upgrade, and Help. Below this, the 'Storage Array EF550' is selected, showing an 'Optimal' status. The left sidebar contains a tree view with categories like All Logical Objects, Total Unconfigured Capacity, Disk Pools, Volume Groups, and Consistency Groups. The 'Volume Groups' section is expanded, showing 'vg_data01 (RAID 1)' and 'v_data01 (1,433.600 GB)'. The main panel displays the configuration for 'Volume "v_data01"'. A table lists the volume's status, capacity, and identifiers.

Volume "v_data01"	
Volume status:	Optimal
Thin provisioned:	No
Capacity:	1,433.600 GB
Volume world-wide identifier:	60:08:0e:50:00:29:3a:68:00:00:25:f0:52:a9:eb:b7
Subsystem ID (SSID):	10
LUN:	Not Mapped
Accessible By:	NA

3.2 Provision EF-Series Storage Using SANtricity 11.10 CLI

The creation of the volume group and volume shown in section 3.1, "Provision EF-Series Storage Using SANtricity 11.10 GUI" can also be scripted and run from SANtricity.

To create a script first of the storage objects using SANtricity, complete the following steps:

1. In the AMW, from the Storage Array menu, select Configuration > Save.

SANtricity®

Storage Array | Storage | Copy Services | Host Mappings | Hardware | Monitor | Upgrade | Help

Premium Features...
 Security
 Change
Configuration
 Rename...
 Preferences...
 Exit

Automatic
 Hot Spare Coverage...
Save...
 Clear

Volume "v_data01"

[View Associated Physical Components](#)


Volume status:	Optimal
Thin provisioned:	No
Capacity:	1,433.600 GB
Volume world-wide identifier:	60:08:0e:50:00:29:3a
Subsystem ID (SSID):	10
RAID level:	10
LUN:	Not Mapped
Accessible By:	NA

Volume Groups

- vg_data01 (RAID 10) (1,489.423 GB)
 - v_data01 (1,433.600 GB)**
 - Free Capacity (55.823 GB)
- VolumeGroup1 (RAID 5) (3,723.563 GB)
- VolumeGroup2 (RAID 5) (3,723.563 GB)

Consistency Groups

2. Select the Volume configuration to save and click Yes.



This option will save the configuration of this storage array to a script file so that you can copy the configuration to another storage array.
Note: This script file should not be used for disaster recovery purposes. Refer to the Online Help for more information.

IMPORTANT:
Do not use this option if there is an operation currently being performed that is changing the configuration. For more information about viewing the status of operations in progress, refer to the Online Help.

Save configuration
Select which elements of your configuration to save.

NOTE: You cannot save the volume-to-LUN mappings without also saving the volume configuration and topology elements.

☐ Storage array settings
☒ Volume configuration
☐ Volume-to-LUN mappings
☐ Topology

Use the Load Configuration option to load the saved configuration script file.

Are you sure you want to continue?

Yes No Help

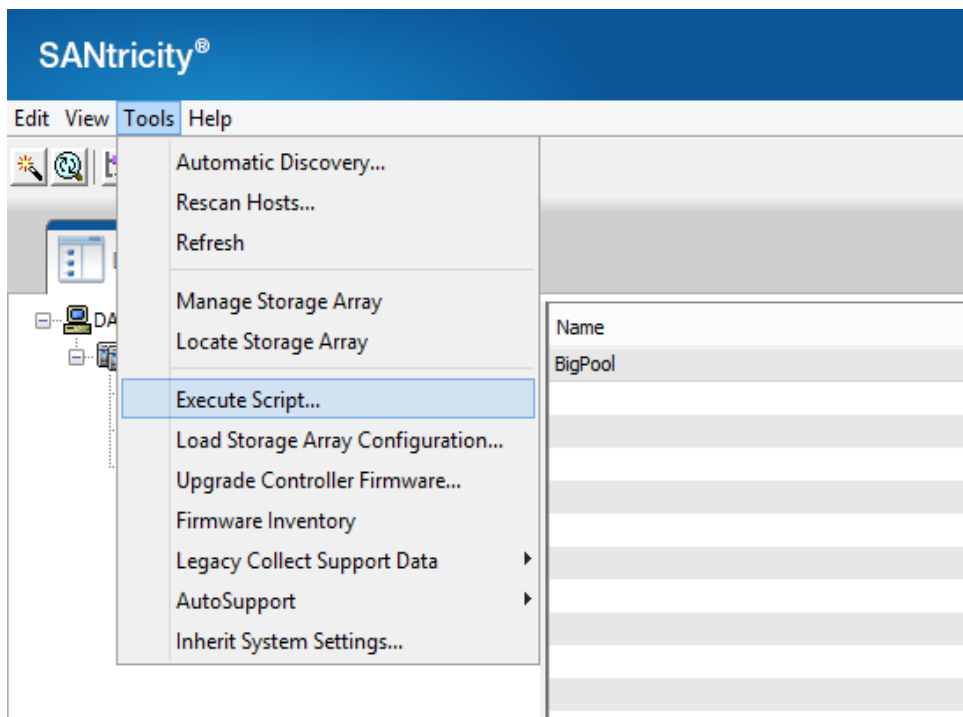
3. Follow the Windows prompt to save the configuration file to a location of your choice.
4. After the storage configuration is saved, edit it to recreate the same configuration on multiple storage arrays as desired.
Note: Verify that the volume group drives are the same as in the original array or modify as needed.
5. The following script was generated using the functionality in SANtricity to save a storage configuration in a text file.

```
// Logical configuration information from Storage Array Site_C_01.
// Saved on December 9, 2013
// Firmware package version for Storage Array Site_C_01 = 98.10.02.01

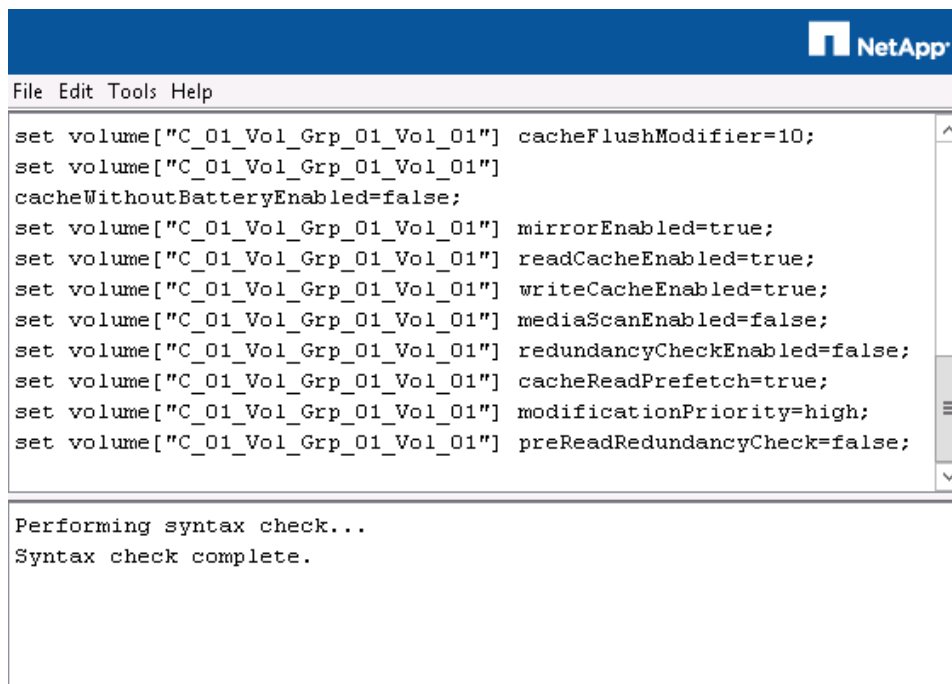
show "Creating Volume Group C_01_Vol_Grp_01, RAID 5.";
//This command creates volume group <C_01_Vol_Grp_01>.
create volumeGroup drives=(99,2,5 99,3,6 99,4,5 99,5,6 99,2,6 99,3,7 99,4,6 99,5,7 99,2,7)
RAIDLevel=5 userLabel="C_01_Vol_Grp_01" securityType=capable
dataAssurance=none;

show "Creating volume C_01_Vol_Grp_01_Vol_01 on volume group C_01_Vol_Grp_01.";
//This command creates volume <C_01_Vol_Grp_01_Vol_01> on volume group <C_01_Vol_Grp_01>.
create volume volumeGroup="C_01_Vol_Grp_01" userLabel="C_01_Vol_Grp_01_Vol_01" owner=A
capacity=107374182400 Bytes segmentSize=512 dssPreAllocate=true dataAssurance=enabled
mapping=none;
show "Setting additional attributes for volume C_01_Vol_Grp_01_Vol_01.";
// Configuration settings that can not be set during Volume creation.
set volume["C_01_Vol_Grp_01_Vol_01"] cacheFlushModifier=10;
set volume["C_01_Vol_Grp_01_Vol_01"] cacheWithoutBatteryEnabled=false;
set volume["C_01_Vol_Grp_01_Vol_01"] mirrorEnabled=true;
set volume["C_01_Vol_Grp_01_Vol_01"] readCacheEnabled=true;
set volume["C_01_Vol_Grp_01_Vol_01"] writeCacheEnabled=true;
set volume["C_01_Vol_Grp_01_Vol_01"] mediaScanEnabled=false;
set volume["C_01_Vol_Grp_01_Vol_01"] redundancyCheckEnabled=false;
set volume["C_01_Vol_Grp_01_Vol_01"] cacheReadPrefetch=true;
set volume["C_01_Vol_Grp_01_Vol_01"] modificationPriority=high;
set volume["C_01_Vol_Grp_01_Vol_01"] preReadRedundancyCheck=false;
```

6. In the Enterprise Management window (EMW), click Tools > Execute Script.



7. Paste the script into the Script Editor and verify the syntax before executing.



8. As shown in section 3.1, "Provision EF-Series Storage Using SANtricity 11.10 GUI from the Storage & Copy Services tab, confirm that the new volume group is displayed in the storage system tree and that the new volume is branching from the new volume group.

3.3 Windows Volume Mount Points

NetApp storage solutions and Microsoft SQL Server 2005 onward support mount points. Mount points are directories on a volume that can be used to mount a different volume. Mounted volumes can be accessed by referencing the path of the mount point. Mount points eliminate the Windows 26-drive-letter limit and offer greater application transparency when moving data between LUNs, moving LUNs between hosts, and unmounting and mounting LUNs on the same host. This is because the underlying volumes can be moved around without changing the mount point path name.

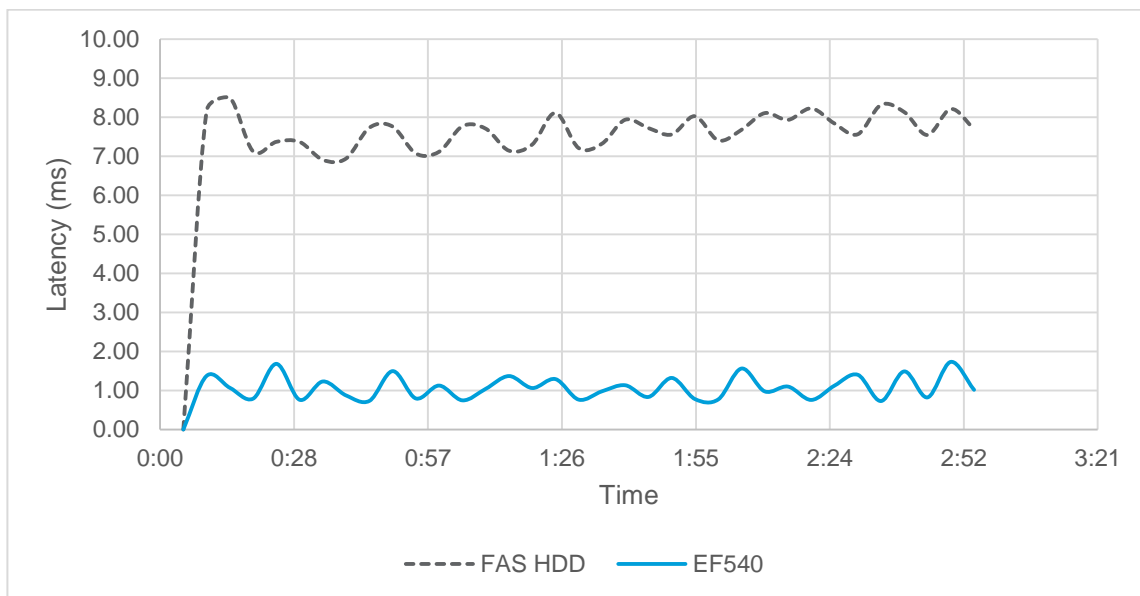
NetApp recommends:

- Using NTFS mount points instead of drive letters to surpass the 26-drive-letter limitation in Windows.
- When using volume mount points, the name given to the volume label and mount point must be the same.

4 Provisioning SQL Server 2012

In comparison tests of OLTP databases, where the database files reside on either NetApp FAS3170 SAS HDD storage or NetApp EF-Series storage system, results show an impressive performance improvement of read latency when the database files reside on an EF array, as shown in Figure 4. The EF-Series Flash Array allows the DBA to increase the performance of existing applications while lowering the cost of the IOPS ratio without the need for rearchitecting the application.

Figure 4) OLTP database comparison: latency.



4.1 SQL Server Database Files

Provisioning database files on EF-Series Flash Array can be done in two ways:

1. During the database creation, in order to create a database that has database files residing on EF-Series, you can use the following T-SQL:

```
-- Assuming C:\MSSQL\Data and C:\MSSQL\Log is the mount points of EF-Series LUNs  
  
USE master;  
GO  
CREATE DATABASE Sales  
ON
```

```

( NAME = Sales_dat,
  FILENAME = 'C:\MSSQL\Data\saledat.mdf',
  SIZE = 10,
  MAXSIZE = 50,
  FILEGROWTH = 5 )
LOG ON
( NAME = Sales_log,
  FILENAME = 'C:\MSSQL\Log\salelog.ldf',
  SIZE = 5MB,
  MAXSIZE = 25MB,
  FILEGROWTH = 5MB ) ;
GO

```

2. After the database creation, in order to move the database files from non-EF-Series LUNs to EF-Series LUNs, you must stop SQL Server service and detach the database files. After the databases have been detached, you can copy the files to the path or mount points that reside in EF-Series LUNs. After the files have been copied, you can attach the database from the new location.

The common best practice is to separate data, transaction logs, and tempdb files in separate logical LUNs. The origin of this recommendation lies with the separation of the types of workload between different physical storages. This is still a valid recommendation for environments where you can guarantee that separation. However, it is common to see customers deploying SQL Server in a shared storage environment, where physical separation is much harder to achieve and usually is not even necessary for performance reasons.

It is still a good idea to maintain separation to help with manageability so that potential problems are easier to isolate. For example, separating tempdb onto its own logical disk means that you can presize it to fill the disk without worrying about space requirements for other files. The more separation you implement, the easier it is to correlate logical disk performance to specific database files.

4.2 Database Files for tempdb

The tempdb system database is a global resource that is available to all users connected to the instance of SQL Server and is used to hold the following:

- Temporary user objects that are explicitly created, such as global or local temporary tables, temporary stored procedures, table variables, or cursors.
- Internal objects that are created by the SQL Server Database Engine, such as work tables to store intermediate results for spools or sorting.
- Row versions that are generated by data modification transactions in a database that uses read-committed row versioning isolation or Snapshot isolation transactions.
- Row versions that are generated by data modification transactions for features, such as online index operations, multiple active result sets (MARS), and AFTER triggers.

Operations within tempdb are minimally logged, enabling transactions to be rolled back. tempdb is recreated every time SQL Server is started so that the system always starts with a clean copy of the tempdb database. Temporary tables and stored procedures are dropped automatically on disconnect, and no connections are active when the system is shut down. Therefore, there is never anything in tempdb to be saved from one session of SQL Server to another. Backup and restore operations are not allowed on tempdb.

Because every SQL Server instance has a shared database named tempdb that is for use by temporary objects and there is only one per instance, it often proves to be a bottleneck for those systems that make heavy usage of it. Typically, this happens because of PAGELATCH, in-memory latch contention on the allocation bitmap pages inside of the data files.

It is possible to reduce the contention on the in-memory pages by adding additional data files to tempdb, with the same initial size and autogrowth configuration. This works because SQL Server uses a round robin, proportional fill algorithm to stripe the writes across the data files. When multiple data files exist for

a database, all of the writes to the files are striped to those files, with the writes to any particular file based on the proportion of free space that the file has to the total free space across all of the files. This means that writes are proportionally distributed to the files according to their free space and to make sure that they fill at the same time, irrespective of their size.

Microsoft recommends up to a one-to-one mapping between the number of files and logical CPUs. During testing of massive workloads, they have seen performance benefits, even with hundreds of data files.

A more pragmatic approach, however, is to have a one-to-one mapping between files and logical CPUs up to eight, and then add files if you continue to see allocation contention or if you are looking to push the I/O subsystem harder.

Because the NetApp EF-Series Flash Array provides a robust platform for delivering exceptional performance, you can create multiple tempdb files and place them on the EF-Series Flash Array. You can accomplish this by using the following Transact SQL.

```
select *
from sys.database_files

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 to C:\MSSQL\Tempdb path
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');

-- Assign proper size for tempdev01
ALTER DATABASE [tempdb] MODIFY FILE ( NAME = N'tempdev01', SIZE = 2GB, FILEGROWTH = 100 MB );
ALTER DATABASE [tempdb] MODIFY FILE ( NAME = N'templog', SIZE = 4GB, FILEGROWTH = 100 MB );

-- Add more tempdb files
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev02', FILENAME = N'C:\MSSQL\Tempdb\tempdev02.ndf' , SIZE = 2GB , FILEGROWTH = 100 MB);
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev03', FILENAME = N'C:\MSSQL\Tempdb\tempdev03.ndf' , SIZE = 2GB , FILEGROWTH = 100 MB);
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev04', FILENAME = N'C:\MSSQL\Tempdb\tempdev04.ndf' , SIZE = 2GB , FILEGROWTH = 100 MB);
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev05', FILENAME = N'C:\MSSQL\Tempdb\tempdev05.ndf' , SIZE = 2GB , FILEGROWTH = 100 MB);
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev06', FILENAME = N'C:\MSSQL\Tempdb\tempdev06.ndf' , SIZE = 2GB , FILEGROWTH = 100 MB);
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev07', FILENAME = N'C:\MSSQL\Tempdb\tempdev07.ndf' , SIZE = 2GB , FILEGROWTH = 100 MB);
ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev08', FILENAME = N'C:\MSSQL\Tempdb\tempdev08.ndf' , SIZE = 2GB , FILEGROWTH = 100 MB);
```

5 High Availability

5.1 EF-Series and SANtricity 11.10

The NetApp EF-Series storage system has been architected for highest reliability and availability with features such as:

- Dual active controller with automated I/O path failover
- RAID levels 0, 1, 5, 6, and 10 or Dynamic Disk Pools (DDP)
- Redundant, hot-swappable controllers, disk drives, power supplies, and fans
- Automatic drive failover detection and rebuild using global hot spares
- Mirrored data cache with battery backup and destage to memory

- Nondisruptive controller firmware upgrades
- Proactive drive health monitoring
- Background media scan with autoparity check and correction

All components are fully redundant and may be swapped without powering off the system or even halting operation. This includes controllers, disk drives, power supplies, and fans. The EF-Series power supplies offer an 80 plus efficiency rating. The EF-Series Flash Array features several functions designed to protect data in every circumstance. Multiple RAID levels are available for use with varying levels of redundancy. Failover from one path to another in the case of a lost connection is also automatically included with the system. Within the shelf, each drive has a connection to each controller so that even internal connection issues can be quickly overcome. Volumes on the system are available for host I/O from the moment they are created and can even have significant properties altered without stopping I/O.

Other features of the EF-Series Flash Array that protect data include mirroring and backing up controller cache. If power is lost to the system during operation, onboard batteries are used to de stage the data from cache memory to internal controller flash so that it will be available when power is restored. The RAID algorithms allow the system to recreate any lost data in the rare case of drive failure. Users also have the option of confirming data with RAID parity at all times and even continuing a rebuild when hitting an unreadable sector.

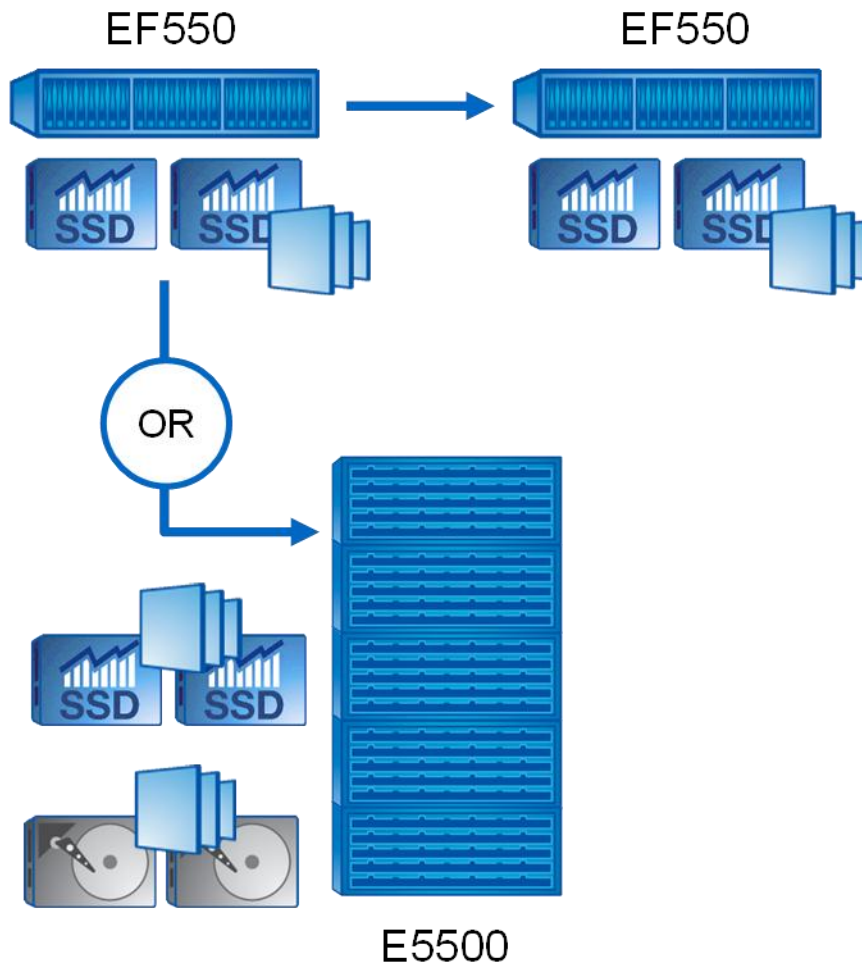
Behind the scenes, the system performs other tasks that protect data at all times. The optional media scan feature looks for inconsistencies even on sectors not currently being accessed by any host. The EF-Series proactively tracks SSD wear life and flags drives that are approaching the end of their expected life. All types of diagnostic data are constantly collected for later use by support if necessary.

Not only does the EF-Series offer the detailed reliability and availability features already described, but using the SANtricity software it is also possible to maximize availability:

- High-speed, high-efficiency Snapshot technology
 - Protects data in seconds
 - Reduces flash consumption by storing only changed blocks
- Robust disaster recovery protection
 - Synchronous mirroring for no-data-loss protection of content
 - Asynchronous mirroring for long-distance protection and compliance
- Flexible protection to maximize ROI
 - Recovery target can be flash, NL-SAS, or mixed based on cost/performance needs
 - Delivers speed without breaking budgets

Figure 5 illustrates a graphical representation of the HA possibilities using Snapshot technology and mirroring. For more information, refer to the NetApp Support site [Documentation](#) library and the SANtricity online help.

Figure 5) EF-Series HA using Snapshot technology and mirroring.



5.2 SQL Server High-Availability Options

The high availability of SQL Server solutions masks the effects of a hardware or software failure and maintains the availability of applications so that the perceived downtime for users is minimized. SQL Server provides several solutions such as:

- **Log shipping.** Log shipping operates at the database level. It can maintain one or more warm standby databases (referred to as secondary databases) for a single production database that is referred to as the primary database. For more information about log shipping, see [About Log Shipping \(SQL Server\)](#).
- **Database mirroring.** Database mirroring is a solution to increase database availability by supporting almost instantaneous failover. Database mirroring can be used to maintain a single standby database, or mirror database, for a corresponding production database that is referred to as the principal database. For more information, see [Database Mirroring \(SQL Server\)](#).
- **AlwaysOn Failover Cluster Instances.** AlwaysOn Failover Cluster Instances leverages Windows Server® Failover Clustering (WSFC) functionality to provide local high availability through redundancy at the server instance level: a failover cluster instance (FCI). An FCI is a single instance of SQL Server that is installed across WSFC nodes and possibly across multiple subnets. On the network, an FCI appears to be an instance of SQL Server running on a single computer, but the FCI provides

failover from one WSFC node to another if the current node becomes unavailable. For more information, see [AlwaysOn Failover Cluster Instances \(SQL Server\)](#).

- **AlwaysOn Availability Groups.** AlwaysOn Availability Groups is an enterprise-level high-availability and disaster recovery solution introduced in SQL Server 2012 to enable you to maximize availability for one or more user databases. AlwaysOn Availability Groups requires that the SQL Server instances reside on WSFC nodes. For more information, see AlwaysOn Availability Groups (SQL Server). Availability Groups support two availability modes, asynchronous-commit mode and synchronous-commit mode, as follows:
 - Asynchronous-commit mode is a disaster recovery solution that works well when the availability replicas are distributed over considerable distances. For more information, see [Asynchronous-Commit Availability Mode](#).
 - Synchronous-commit mode emphasizes high availability over performance, at the cost of increased transaction latency. Under synchronous-commit mode, transactions wait to send the transaction confirmation to the client until the secondary replica has hardened the log to disk. For more information, see [Synchronous-Commit Availability Mode](#).

Automatic failover provides high availability by making sure that the database is quickly made available again after the loss of the primary replica. To configure an availability group for automatic failover, you need to set both the current primary replica and one secondary replica to synchronous-commit mode with automatic failover.

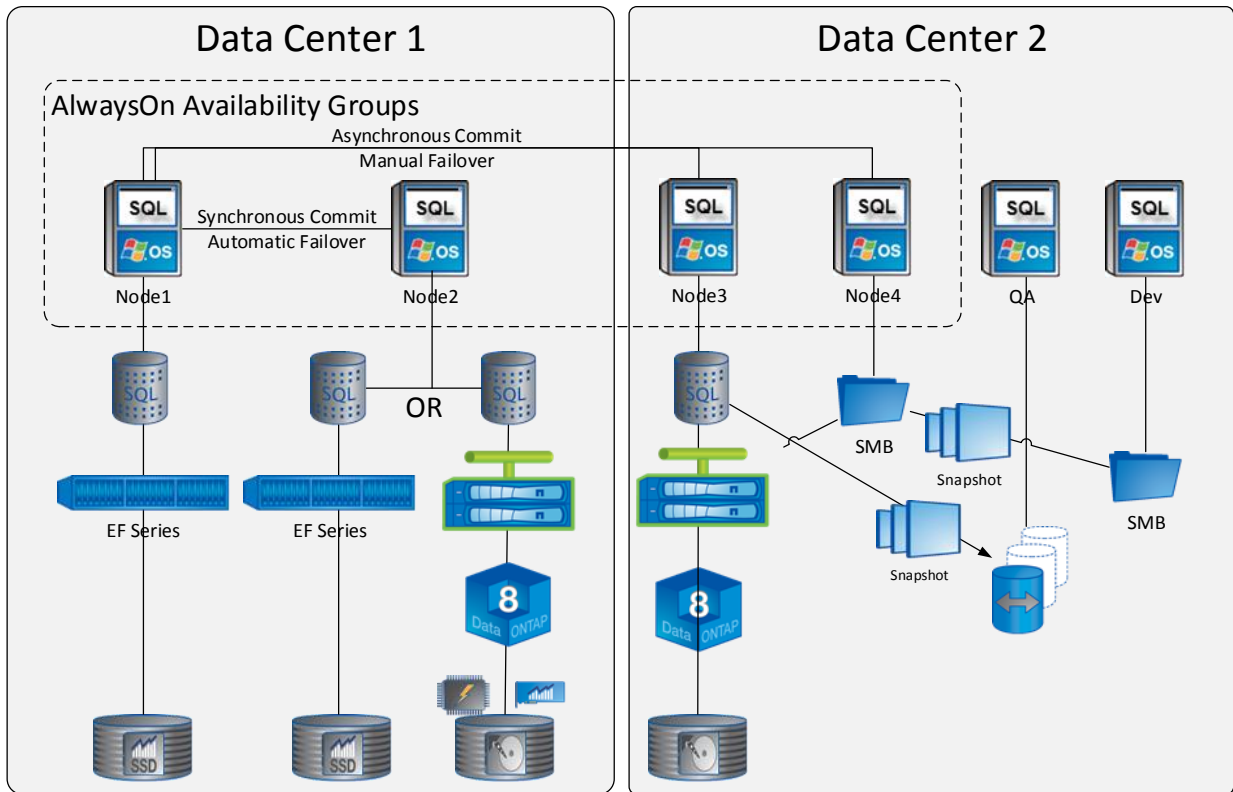
FAS storage systems allow you to create Snapshot copies and clone databases easily using SnapDrive® for Windows and SnapManager® for SQL Server. Figure 6 illustrates the simple topology that utilizes both EF-Series and FAS storage.

Table 2 lists the transaction latency based on the used storage.

Table 2) Transaction latency versus used storage.

Transaction Latency	Storage Architecture
Low	EF-Series Flash Array to EF-Series Flash Array or EF-Series Flash Array to FAS with Flash Cache™ or Flash Pool™
Medium	EF array to FAS SAS drives
High	EF array to FAS SAS or SATA drives

Figure 6) Sample architecture design using AlwaysOn Availability Groups to synchronize database transactions between EF-Series and FAS storage.



To set up hybrid storage, the following requirements are needed:

- SnapDrive for Windows 7 or higher must be installed in all Windows hosts.
- Place other replica on FAS storage and install SMSQL 7 for backup and clone and utilize SMSQL clone lifecycle.
- Use clustered Data ONTAP® 8.2 if you want to utilize SMSQL with SMB 3.0 for ease of use by developers and test engineers.

6 Sizing

SQL Server performance has generally been centered on I/O. Traditionally, this performance was improved by either increasing the number of spindles or making the spindles go more quickly. With the advent of the EF-Series Flash Array, performance improvement can be achieved by using the SSD drives.

6.1 EF-Series I/O Overview

There are several factors that can affect the overall performance of an EF-Series storage system, including physical components, such as networking infrastructure, and the configuration of the underlying storage itself. Generically, storage system performance tuning can be defined as following a 40/30/30 rule, where 40% of the tuning and configuration is at the storage system level, 30% is at the file system level, and the final 30% is at the application level. The following sections largely describe the 40% related

to storage system specifics. At a high level, some of the considerations at the file system and application level include:

- **I/O size.** E-Series storage systems are largely responsive systems that require a host to request an I/O operation in order to complete that operation. The I/O size of the individual requests from the host can have a significant effect on either the number of I/O operations per second (IOPS) or throughput (generally described in terms of megabytes per second [MB/s] or gigabytes per second [GB/s]). Larger I/Os typically lead to lower numbers of IOPS and larger MB/s, and the opposite is true as well. This relationship is defined with the equation $\text{throughput} = \text{IOPS} \times \text{I/O size}$.
- **Read versus write requests.** In addition to the I/O size, the percentage of read versus write I/O requests being processed at the storage system level also has a potential effect on the storage system and should be considered when designing a solution.
- **Sequentiality or randomness of the data stream.** The sequentiality (or lack thereof) of the host requests to the underlying disk media logical block addresses (LBAs) has a significant effect on performance at the storage system level, in terms of the ability of physical media to respond effectively to the request with minimal latency as well as the effectiveness of the storage system's caching algorithms. A possible exception to increased latency of random requests might be for solid state devices, which do not have mechanically invoked latency.
- **Number of concurrent I/O operations.** The amount of outstanding I/O operations applied to a given volume can vary based on several factors, including whether the file system uses raw, buffered, or direct I/O. Generally, most volumes in an E-Series storage system are striped across several disk drives. Providing a minimal amount of outstanding I/O to each individual disk can cause underutilization of the resources in the storage system, resulting in less than desired performance characteristics.

For those SQL Server customers new to NetApp EF-Series, it might be helpful to review the differences between RAID 10, RAID 5, and DDP technology. For more information, refer to the E-Series Performance Sizing Guide found on the NetApp Support site [Documentation](#) library. Table 3 lists comparison of usable capacity for differing RAID levels. For completeness, all RAID levels supported by NetApp EF-Series are shown.

Table 3) Comparison of usable capacity for differing RAID levels.

Desired Feature	RAID 0	RAID 10	RAID 5	RAID 6	DDP
Usable capacity	100%	50%	$(N-1)/N$ where N is the selected drive count in the volume group	$(N-2)/N$ where N is the selected drive count in the volume group	80% minus selected preservation capacity

6.2 SQL Server I/O Overview

SQL Server is sensitive to I/O latency issues due to the concurrent transactional nature of the SQL Server engine. SQL Server is built on a complicated system of row, page, extent, and table locks that provides transactional consistency throughout the SQL Server system. A poor I/O structure (for example, when I/O takes too long to respond) causes resources to be held longer than necessary, resulting in blocking within the system. When this occurs, it is typically not obvious that the I/O subsystem is the root cause.

- **SQL Server reads.** When reading data from SQL Server, the client first goes to the buffer cache. If the data is not in the buffer cache, SQL Server goes to the I/O subsystem to retrieve the data. The statement does not complete until 100% of the data is read; the user connection or process remains in an I/O wait state until completion.

- **SQL Server writes.** The user writes to the transaction log and the buffer cache. If the data to be modified is not already in the buffer cache, then it must be read into the buffer cache from the I/O subsystem. The buffer manager enables the transaction log to be written to first, before changes are written to the database. This is known as write-ahead logging (WAL). When the user makes the change and the COMMIT is executed, a log write is displayed informing about the change that took place, allowing the COMMIT to complete. After the COMMIT is complete, the user process can continue on to the next stage or command without having to wait for the changes to be written to the disk. ROLLBACK TRANSACTION follows the same process as the COMMIT, but in reverse. The buffer manager moves the data from the cache to the disk. It keeps track of log sequence numbers (LSNs) for each log record.
- **Transaction log.** The SQL Server transaction log is a write-intensive operation that is sequential in nature. The transaction log is used to provide recoverability of data in the case of database or instance failure.

The OLTP database system within the SQL Server environment is most dependent on getting the greatest number of transactions through the system in the least amount of time. Examples of different types of OLTP systems include web order systems and manufacturing tracking systems. OLTP systems can have large volumes of transactions per second (TPS), and for the OLTP system it is all about throughput. For these transactions to take place, SQL Server relies on an efficient I/O subsystem. Based on a Microsoft SQL Server best practices article (<http://msdn.microsoft.com/en-us/library/cc966412.aspx>), an OLTP transaction profile has the following attributes:

- OLTP processing is generally random in nature for both reads and writes issued against data files.
- I/O activity will be approximately 67% read and 33% write.
- Read activity (in most cases) is consistent and uses point queries; it does not consist of large time-consuming queries.
- Write activity to the data files occurs during checkpoint operations (frequency is determined by recovery interval settings).
- Log writes are sequential in nature with a varying size that depends on the nature of the workload (sector aligned up to 60KB).
- Log reads are sequential in nature (sector aligned up to 120KB).

6.3 Estimating I/O

Estimating the number of I/O operations required for a system is crucial when sizing a database. This exercise helps the administrator understand how to keep the database instance performing within acceptable limits. You must estimate I/O when you are unable to get the actual physical I/O numbers for the system. This is typically the case in new systems that are in the process of being constructed. Following are the formulas for estimating I/O.

New OLTP Database System

To estimate I/O for a new database system without access to the system, do as follows:

1. Estimate the number of transactions for a given period size.
2. Multiply the number of transactions by the 0.85 saturation rate and then divide that by the number of seconds in a day. The seconds in a day are determined by the hours of operation for the database. If the database operates in a 24-hour environment, the number is 86,400.
The formula for estimating the number of I/O operations is:
Total I/O = (estimated number of transactions × 0.85)/seconds in a day.
For example, if there are 40,000 transactions on a system that operates 24 hours per day, the formula is as follows:
 $(40,000 \times 0.85) / 86,400 = 0.3935 \text{ IOPS}$

3. After determining the I/O required, determine the read and write I/O by multiplying the number of I/O operations by the percentage of reads or writes. The I/O activity will be approximately 67% read and 33% write for an OLTP system.

The formula for I/O in megabytes is (number of transactions × 0.85)/seconds in a day × type % = I/O megabytes.

For example, to determine the reads and writes for an OLTP system, the formula is as follows:

$((40,000 \times 0.85)/86,400) \times 0.67 = 0.3148\text{MB reads}$

$((40,000 \times 0.85)/86,400) \times 0.33 = 0.0787\text{MB writes}$

Existing OLTP Database System

When sizing for an existing database environment, understanding the type of workload and interpreting the statistical data are helpful. It is important to gather statistics during periods of peak stress on the system. PerfMon allows you to see the high-water marks for the time frame in which you monitor the system.

After either IOPS or throughput (MB/sec) of the system is captured, it can be input into the [EF Performance Sizing System Modeler](#). Figure 7 and Figure 8 show the input fields for the sizing tool. Figure 9 lists the output.

Figure 7) E-Series performance sizing tool.

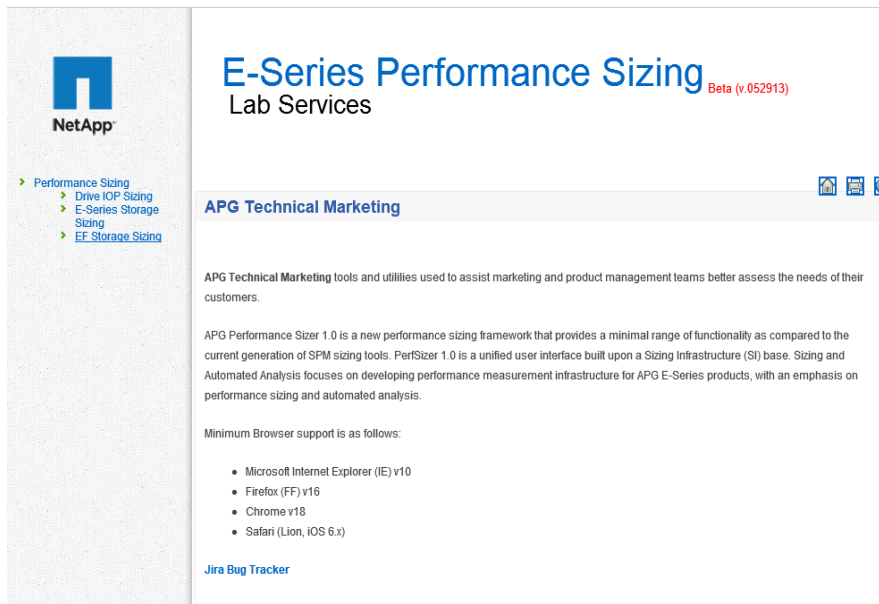


Figure 8) Input fields for EF-Series performance sizing.

The screenshot displays the 'EF Performance Sizing System Modeler' interface. On the left is a vertical sidebar with the NetApp logo and a 'Performance Sizing' link. The main content area is divided into four sections: 'EF Hardware', 'RAID Configuration', 'Workload Configuration', and 'Target Performance Criteria'. Each section contains various input fields for configuring the storage system.

EF Performance Sizing System Modeler

EF Hardware

- Controller Models: EF540 24GB
- Interface Types (HIC): 8Gb FC
- Disk Shelves: 2U24
- Disk Types: 800 GB SSD

RAID Configuration

- RAID Level: RAID 10

Workload Configuration

- IO Request Size (KB): 128
- IO Pattern: Random
- Write (%): 30
- Read (%): 70

Target Performance Criteria

- ☒ IOPs
- ☐ Throughput (MB/sec)
- Target Performance Value: 200000

Figure 9) Output from EF-Series performance sizing tool.

System Commodities

Base Storage (RBOD):

Commodity	Description	Part Number	Quantity
System	EF540; 24GB	EF540-24X800-R6	3
HIC	4 Port; 8Gb; FC	X-48855-00-R6	6
--	--	--	--
--	--	--	--

Expansion Storage (EBOD):

Commodity	Description	Part Number	Quantity
--	--	--	--
--	--	--	--

System Pricing Summary

Commodity Pricing (\$):

Commodity:	Description:	List Price:	Total Price:
System	EF540; 24GB(GB)	\$345600.00	\$1036800
Interface	4 Port; 8Gb; FC	\$2020.00	\$12120
		=====	
Total:			\$1048920

Performance Pricing (\$):

Price(\$)	per IOP:	\$3/IOP
Price(\$)	per GB:	\$19/GB
Price(\$)	per (MB/s):	\$22/(MB/s)
Price(\$)	per Watt:	\$546/Watt

Performance Summary

Total Throughput (MB/sec):	26215
Total IOPs:	200000

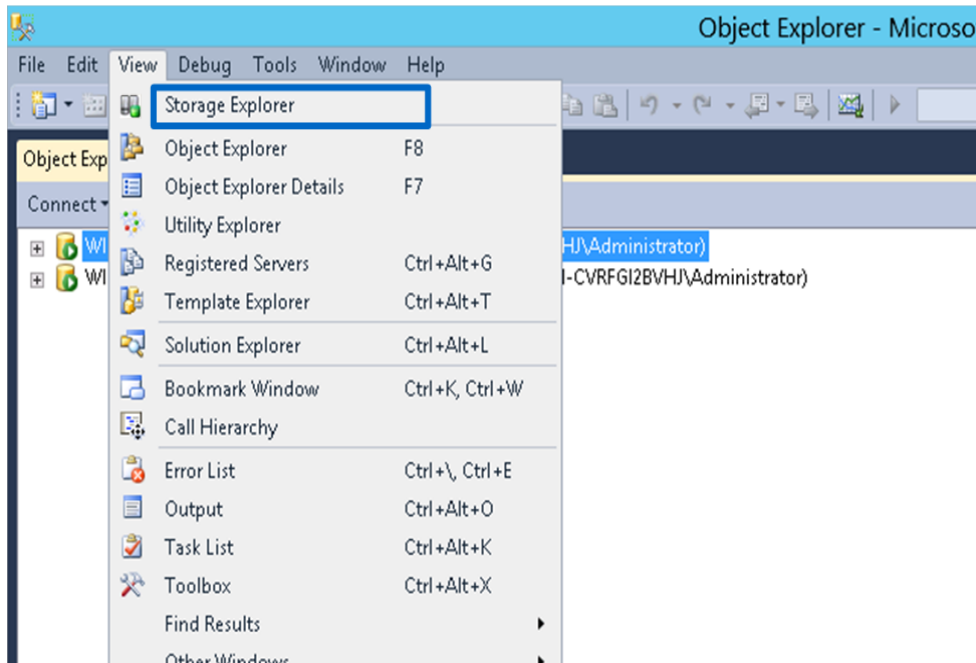
System Summary

Model:	EF540
Total System Capacity (GB):	57600(GB)
Total Disk Drives:	72
Total Systems:	3
Total Cache Size/System:	24GB
Total Rack Units:	6U
Total Racks(x40U):	1x40U

Power Summary

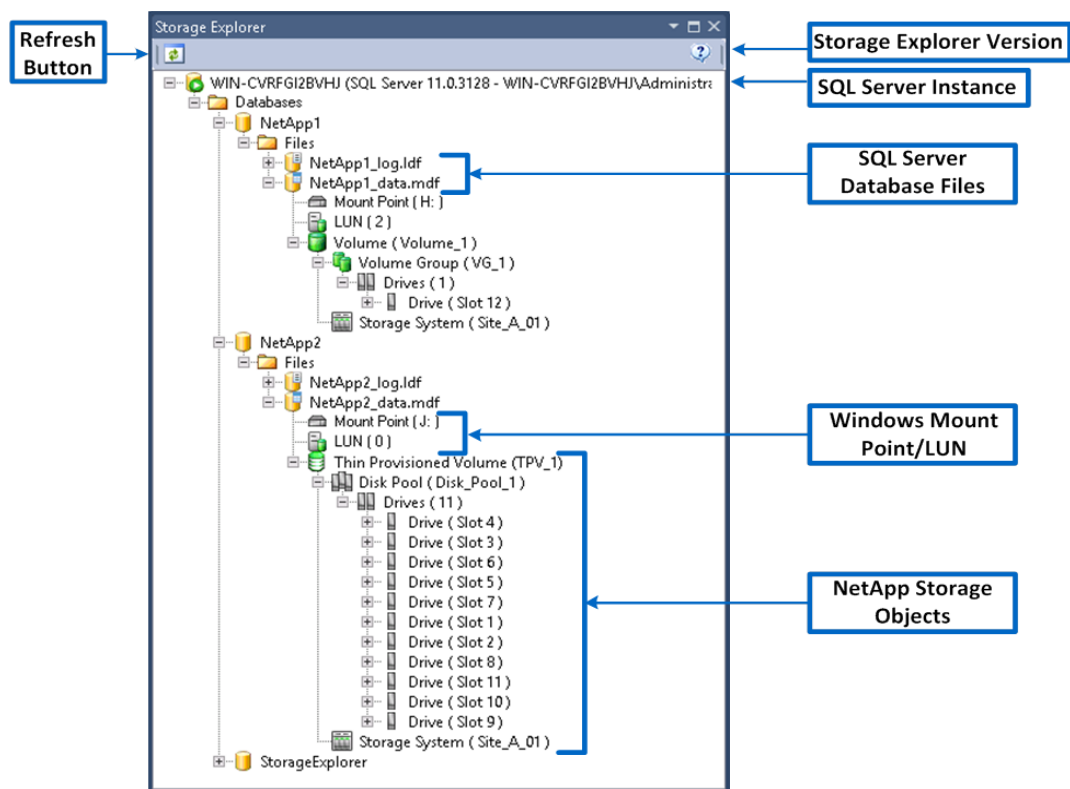
Amps:	VA:	Watts:	BTU:
8.82	1941	1921.59	6556.74
IOPs per Watt:			105/Watt
MB/s per Watt:			14/Watt

1. The NetApp E-Series Plug-In for SSMS is started from within SSMS. After an instance that uses NetApp E-Series storage has been selected in the Object Explorer, on the View menu select Storage Explorer.



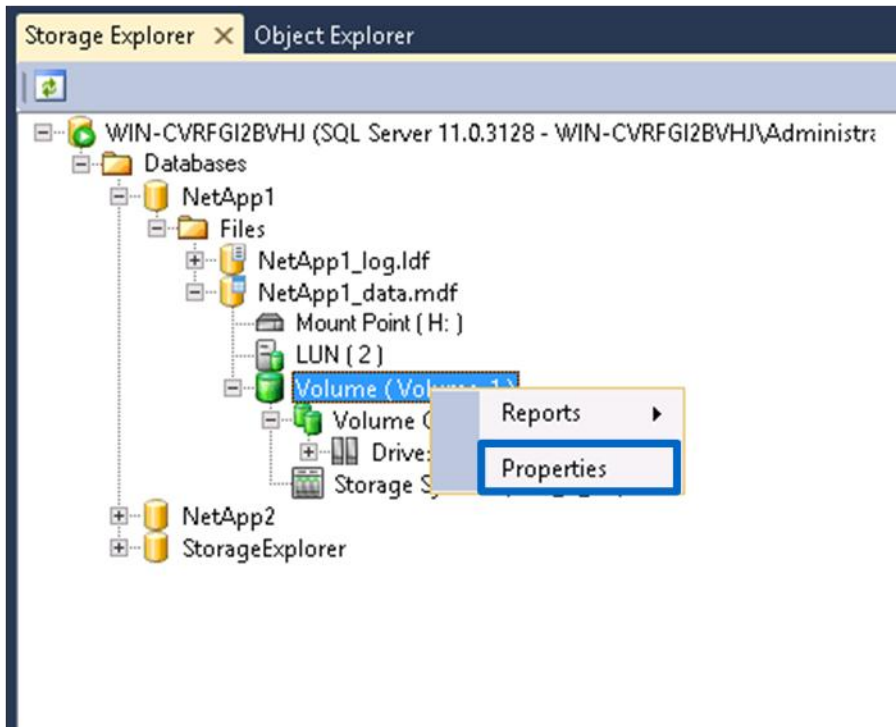
2. After Storage Explorer starts, expand the Databases object to see all databases that have their database files on NetApp E-Series storage. Continue scrolling down to see all objects.
3. To change what instance is being viewed in Storage Explorer, select the instance to be viewed in SSMS Object Explorer and select View > Storage Explorer.

The following screenshot illustrates the expanded Storage Explorer tree.

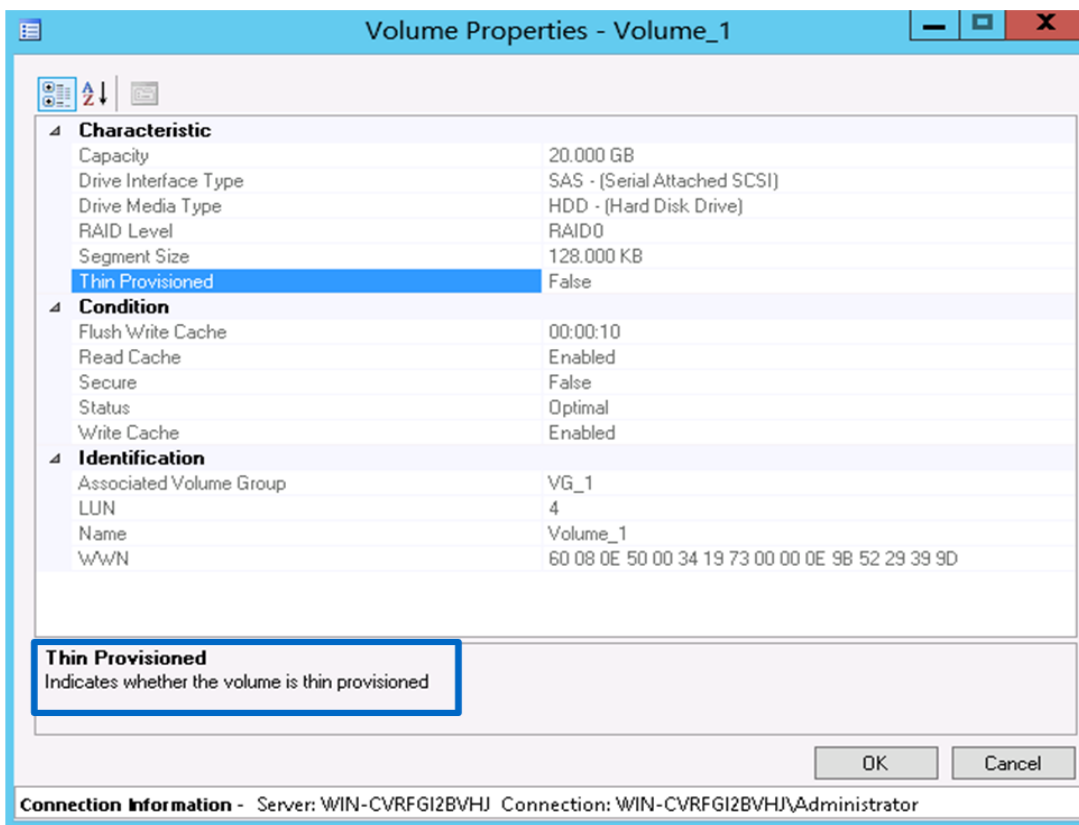


Viewing Properties

To view the properties of a NetApp storage object, right-click the object. The objects that display properties are the **volume**, thin-provisioned volume (TPV), volume group, DDP, drive, tray, and storage system.



The properties for the volume are illustrated in the previous image. By selecting a property in the list, a brief description is provided under the Thin Provisioned pane.

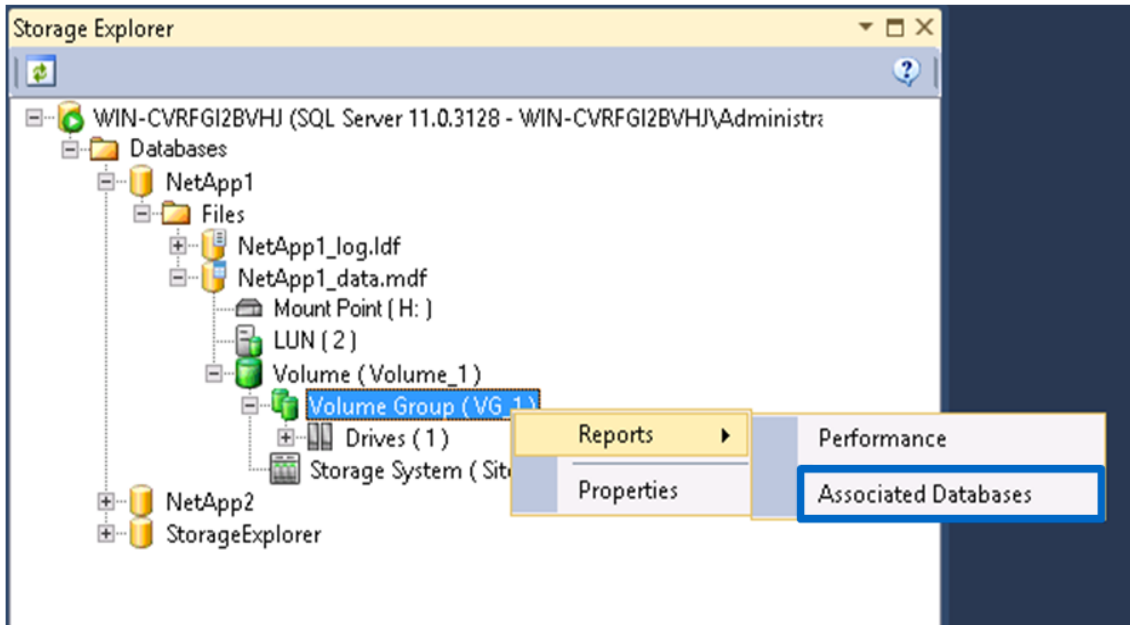


Viewing Reports

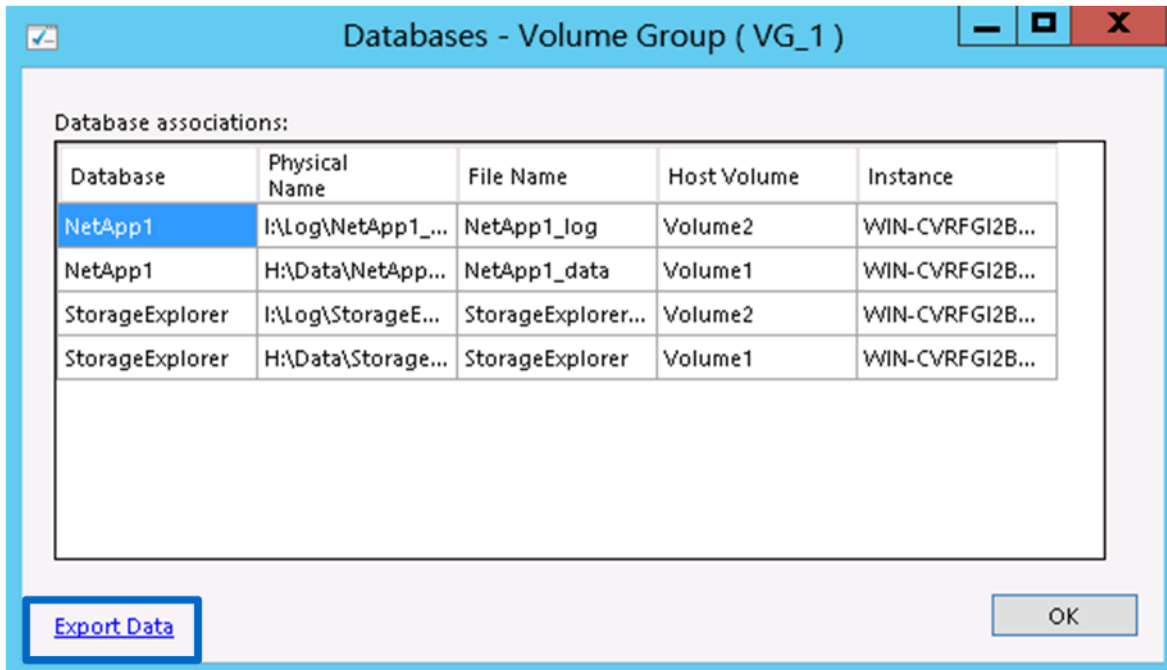
To access the two types of reports that are provided by the Storage Explorer application, right-click the object and choose the type of report.

Associated Databases Report

The Associated Databases report is available for volumes, volume groups, TPV, and DDP.

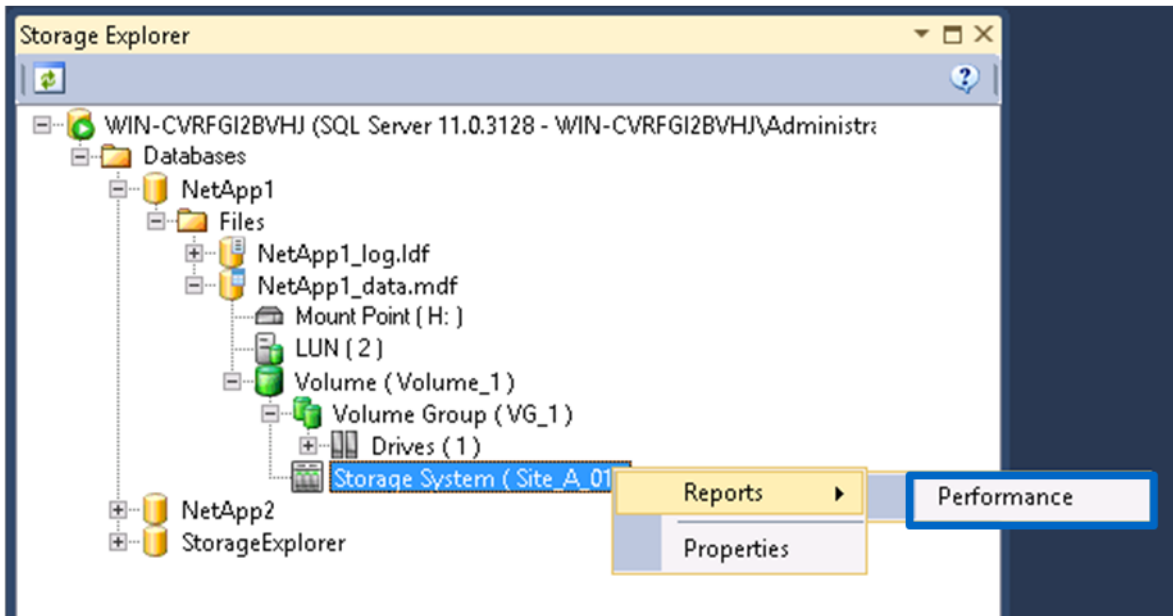


The Associated Databases report provides a listing of all database files stored on the object chosen.

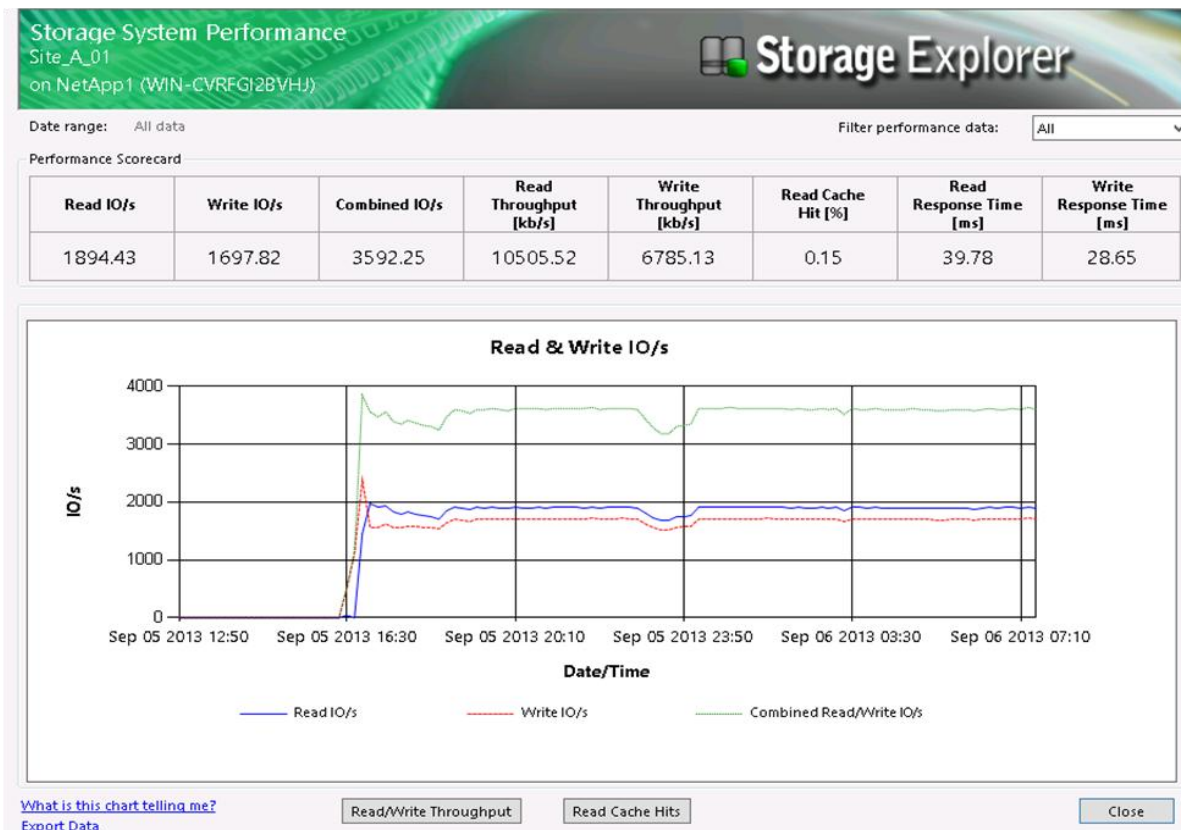


Performance Report

The performance reports are available for volumes, volume groups, TPV, DDP, and storage systems.



The performance reports represent all activity occurring on the specific object. For example, on the Storage System Performance report, the I/O activity for all databases on the storage system is shown, not just one database or datafile.



This allows the DBA to begin at a volume level, verify what database files are on the volume using the Associated Databases report, and then look at the performance reports at this level before moving on to the volume group level. The same type of activity is then performed at the volume group level and finally for the whole storage array.

You can download the user's guide with complete information on installation and operation of the NetApp E-Series Plug-In for SSMS from [NetApp Support](#) with the plug-in software.

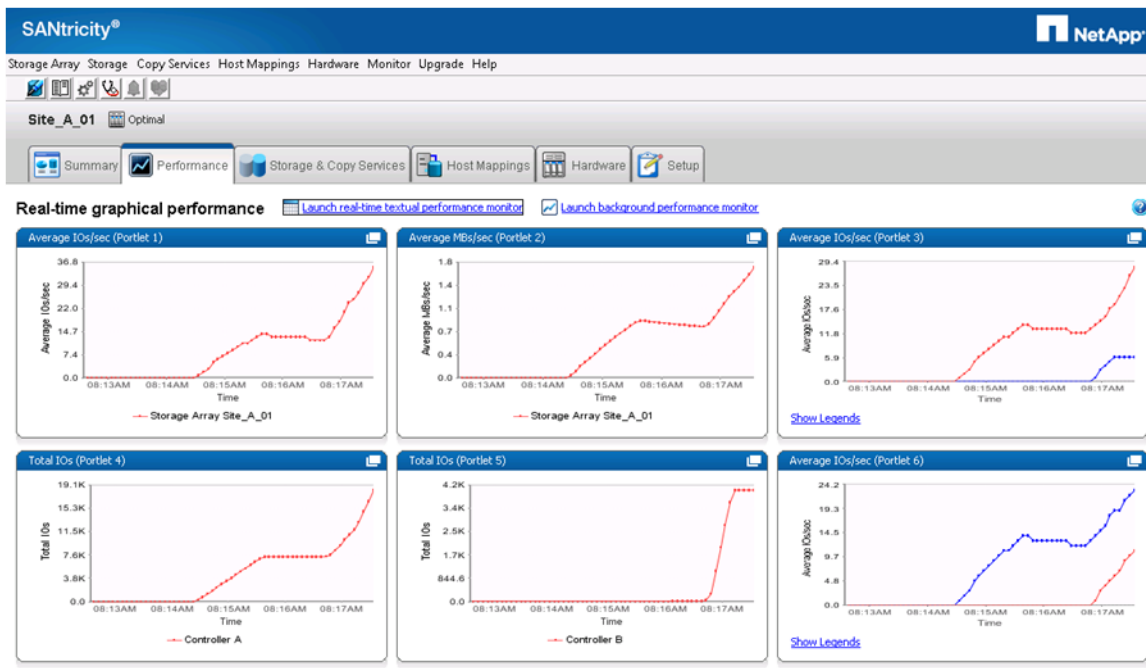
7.2 EF-Series Performance Monitoring Using SANtricity 11.10

While a storage system is in operation, it is often useful to be able to see how the storage system is performing. Using SANtricity, it is possible to view EF-Series performance data in both textual and graphical dashboard formats. Additional details of the monitor feature are located in the Concepts for SANtricity Storage Manager version 11.10 found on the NetApp Support site [Documentation](#) library and the SANtricity online help.

The performance monitor provides visibility into performance activity across your monitored storage devices. You can use the performance monitor dashboard to perform these tasks:

- View different performance metrics on six graphs in real time for up to five monitored devices per graph.
- Performance metrics include:
 - I/O latency for drives and volumes
 - Current or maximum I/O per second
 - Throughput for the entire storage array and volumes
 - Cache hit percentage
 - Total I/Os
- Links that provide a convenient way to start textual performance monitoring or background performance monitoring.

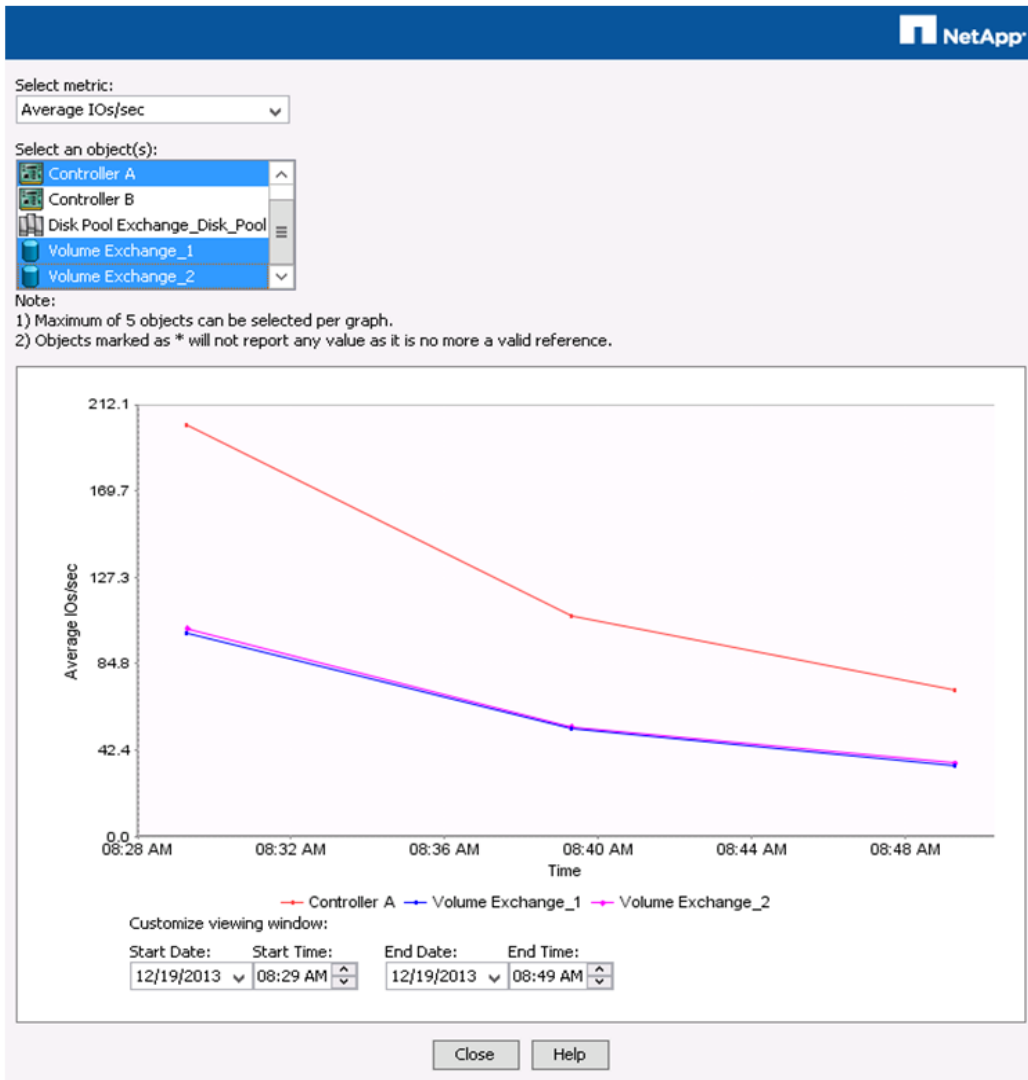
The following screenshot illustrates the performance data that can be collected and viewed by using SANtricity Performance Monitor Dashboard.



Real-time performance data can also be monitored in tabular format (actual values of the collected metrics) and saved to a file for later analysis. The textual view is illustrated.

[illegible]

SANtricity also provides for background performance monitoring. Various reporting attributes, such as time increments and filtering criteria, can be specified to examine performance trends and to pinpoint the cause of availability and performance issues. The following screenshot illustrates the background performance monitoring.



Summary

In general, clients might experience the following challenges in OLTP environments:

- Failure to meet user expectations and service levels because of slow application response time.
- Decreased user and business productivity.
- Application and data availability concerns (slow batch processing, long backup windows, and hardware failure rates).
- Increased storage performance and capacity requirements.
- Scalability constraints because of data center space, power, and cooling limits.
- Increasing TCO
 - Rising data center power and cooling costs.
 - Rising server, network, and storage infrastructure management and support costs.

The NetApp EF-Series Flash Array resolves the challenges in the following ways:

- Dramatically boosting the performance of existing applications and lowering cost per IOPS ratio without the need to rearchitect the application.
- Increasing user productivity with better response times, improving business efficiency.
- Increasing data availability by using advanced system-level high availability and reliability technologies, reducing the number of solution components and shortening batch processing and backup times.
- Increasing storage performance and capacity while decreasing power, cooling, and space requirements.
- Reducing TCO
 - Reducing energy costs because of lower power and cooling requirements.
 - Reducing the number of systems, devices, and components that are required to build the solution by increasing usage of available resources.
 - Reducing management and support costs because of fewer components to deploy and support.
- Faster ROI because of better resource usage.

The NetApp EF-Series Flash Array provides exceptional performance under varying workloads and in a variety of configurations. The system is designed to be flexible, robust, and easy to manage. In addition, it provides superior value to an organization that requires the utmost performance for its mission-critical applications.

References

The following references were used in this TR:

- NetApp Support site documentation library
<http://support.netapp.com/portal/documentation>
- NetApp Support
<http://support.netapp.com/NOW/cgi-bin/software>

Version History

Version	Date	Document Version History
Version 1.0	January 2014	Initial release.

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

NetApp provides no representations or warranties regarding the accuracy, reliability, or serviceability of any information or recommendations provided in this publication, or with respect to any results that may be obtained by the use of the information or observance of any recommendations provided herein. The information in this document is distributed AS IS, and the use of this information or the implementation of any recommendations or techniques herein is a customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. This document and the information contained herein may be used solely in connection with the NetApp products discussed in this document.

Go further, faster®