Technical Report

NetApp Storage Performance Primer

Clustered Data ONTAP 8.3

Bob Allegretti, Roy Scaife, NetApp
April 2015 | TR-4211

Abstract

This paper describes the basic performance concepts as they relate to NetApp® storage systems and the clustered Data ONTAP® operating system. It also describes how operations are processed by the system, how different features in clustered Data ONTAP can affect performance, and how to observe the performance of a cluster.

Data Classification

NetApp Confidential – Limited Use

Version History

Version	Date	Document Version History
Version 1.1	April 2015	Clustered Data ONTAP 8.3 revisions. Authors: Bob Allegretti and Roy Scaife.
Version 1.0	July 2013	Initial version for clustered Data ONTAP 8.2. Authors: Paul Updike, Roy Scaife, and Chris Wilson.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Overview

The demand on IT departments for storage has been steadily increasing, but budgets have not expanded accordingly. Many departments are trying to squeeze more out of their storage infrastructures in both capacity and performance. This document provides performance groundwork as well as the architecture of a NetApp storage system and how the architecture works with clustered Data ONTAP to provide efficiently performing data storage.

Performance is notoriously known for its inherent complexity. NetApp provides simple and capable tools for performance management. This document introduces the reader to an off-box performance management tool specifically designed for clustered Data ONTAP systems. OnCommand® Performance Manager (OPM) is a part of the OnCommand Unified Manager product portfolio and is provided at no additional cost.

2 Introduction to Clustered Data ONTAP Performance

This document is not intended to be a guide on tuning or a deep troubleshooting guide, but rather a general overview of the architecture and operation, the performance management and monitoring principles following the normal performance management paradigm, and the capabilities of NetApp Data ONTAP and FAS systems. Before reading this guide, you should understand the basic concepts of NetApp clustered Data ONTAP. For an introduction to clustered Data ONTAP, see TR-3982: NetApp Clustered Data ONTAP 8.3 and 8.2.x: An Introduction.

2.1 Performance Fundamentals

Many variables affect the performance of a storage system. Out of all the metrics that can be measured, two specifically give the most insight into the performance of the storage system. The first, throughput, describes how much work the system is doing. This is presented in units of work per fixed unit of time (for example MB/s or IOPS). The second, latency, describes the average time it takes to complete a unit of work, referred to as a user operation (for example, read or write operation in ms/op units).

The Data ONTAP operating system and the underlying cluster hardware work efficiently to make sure data is secure and always available. The operations a system performs are a direct function of the client operations requested by applications within an enterprise. The operations requested by an application are referred to as an "application workload," often shortened to simply "workload." Workload characteristics that can affect and describe performance include:

- **Throughput.** The number of operations or amount of data payload over a given period of time.
- **Concurrency.** The number of operations in flight at any point in time.
- **Operation size.** The size of the operations requested of the storage system. The data portion of the operation, for example, a read operation, is often referred to as block size or payload.
- **Operation type.** The type of operation requested of the storage system (for example, read, write).
- **Randomness.** The distribution of data access across a dataset in an unpredictable pattern.
- **Sequentiality.** The distribution of data access across a dataset in a repeatable pattern. Many patterns can be detected: forward, backward, skip counts, and others.
- **Working set size.** The amount of data considered to be active and frequently used to complete work.
- **Dataset size.** The amount of data that exists in a system that is both active and at rest.

Modifying any of these workload characteristics ultimately ends up affecting the performance of the system and can be observed in either latency or throughput. Over time, offered load almost always increases at times without plan or warning. Therefore, to meet performance requirements, the storage administrator must observe the performance of the system and adapt as necessary by making changes to the storage system configuration.

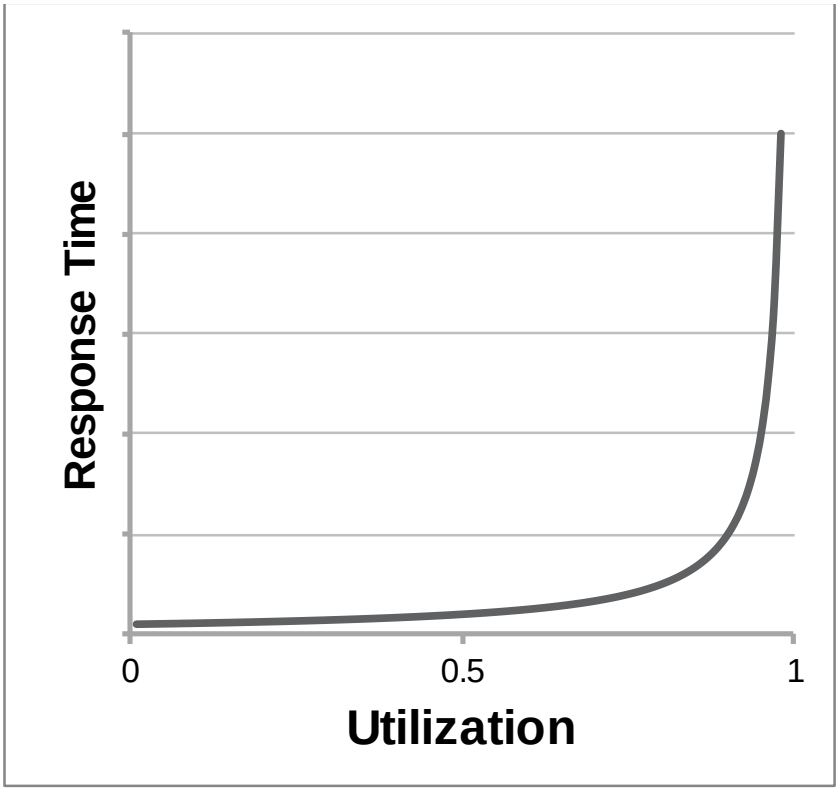
2.2 Normal Performance Relationships

For the purposes of day-to-day management, there are a few guiding principles behind performance. These can be stated as the relationships between the fundamental characteristics of a workload and the resulting performance:

- Throughput is a function of latency.
- Latency is a function of throughput.
- Throughput is a function of concurrency.
- Throughput is a function of operation size.
- Throughput is a function of randomness of operations.
- The host application controls the amount and type of operations.

As a result, the preceding relationships can be summed up by an exponential growth curve as depicted in Figure 1, where response time (or latency) increases nonlinearly as utilization (or throughput) increases.

Figure 1) Response time exponential growth curve as utilization reaches peak.



Throughput and Latency

Workloads can be defined as either closed-loop or open-loop systems. In closed-loop systems, a feedback loop exists. Operation requests from applications are dependent upon the completion of previous operations and, when bounded by the number of concurrent operation requests, limit the offered load. In this scenario the number of concurrent requests is fixed, and the rate that operations that can be completed depends on how long it took (latency) for previous operations to be completed. Simply put, in closed-loop systems, throughput is a function of latency; if latency increases, throughput decreases.

In open-loop systems, operations are performed without relying on feedback from previous operations. This can be a single enterprise-class application generating multiple asynchronous request or hundreds of independently running servers issuing a single threaded request. This means that the response time

from those operations doesn't affect when other operations will be requested. The requests will occur when necessary from the application. As offered load to the system increases, the utilization of the resources increases. As the resource utilization increases, so does operation latency. Because of this utilization increase, we can say that latency is a function of throughput in open-looped systems, although indirectly.

Concurrency

Storage systems are designed to handle many operations at the same time. In fact, peak efficiency of the system can never be reached until it is processing a large enough number of I/Os such that there is always an operation waiting to be processed behind another process. Concurrency, the number of outstanding operations in flight at the same time, allows the storage system to handle the workload in a more efficient manner. The effect can be dramatic in terms of throughput results.

Concurrency is often a difficult concept to grasp because it is very abstract. One way to picture it is to imagine a single application sending a thousand requests in one second and to also imagine a thousand applications sending one request in one second. The concurrency effects are identical to those of the system handling those requests.

Little's Law: A Relationship of Throughput, Latency, and Concurrency

Little's Law describes the observed relationship between throughput (arrival rate), latency (residence time), and concurrency (residents):

$L = A \times W$

This equation says that the concurrency of the system (L) is equal to the throughput (A) multiplied by latency (W). This would mean that for higher throughput, either concurrency would have to increase and/or latency to decrease. This explains why low-concurrency workloads, even with low latencies, can have lower than expected throughput.

Operation Size

A similar effect on concurrency is observed with the size of operations on a system. More work, when measured in megabytes per second, can be done with larger operations than can be done with smaller operations. Each operation has overhead associated with it at each point along the way in transfer and processing. By increasing the operation size, the ratio of overhead to data is decreased, which allows more throughput in the same time. Similarly, when work depends on latency in low-concurrency workloads, a larger operation size increases the efficiency of each individual operation.

Small operations might have a slightly better latency than large operations, so the operations per second could be potentially higher, but the throughput in megabytes will hold a general trend of being lower with smaller operations.

Data Access (Random or Sequential)

Protocol operations sent to a storage system are assigned to a logical location within a data file or LUN. This logical address is subsequently translated into an actual physical location on the permanent storage media. The order of operations and the location of the data being accessed over time determine how random a workload is. If the logical addresses are in order (next to one another), they are considered sequential.

For read operations, performance improves on a NetApp storage system for sequential data. This is because fewer drive seeks and operations are required from one disk I/O operation to the next.

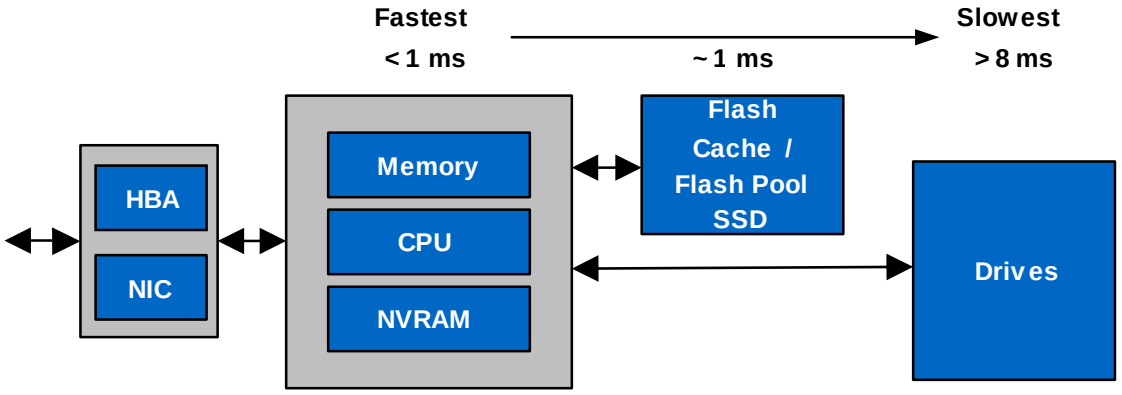
Data ONTAP is write-optimized. Due to the way writes are written to storage, almost all writes behave as if they are sequential writes. Thus, we see less improvement in random versus sequential writes.

2.3 Cluster Node System Architecture Overview

Storage systems are designed to store and retrieve large amounts of data permanently, inexpensively, and quickly. Unfortunately, to store lots of data you need to use a slow medium: the mechanical disk drive. To access data quickly, you need a fast medium such as silicon-based random access memory (RAM), which is neither persistent nor inexpensive. It is also important to remember that different workloads affect different parts of the system in different ways. This creates a problem as to how to optimize access to data to provide the best performance. NetApp does this by innovating in the way data is stored and accessed through the use of unique combinations of spinning disk, flash, and RAM.

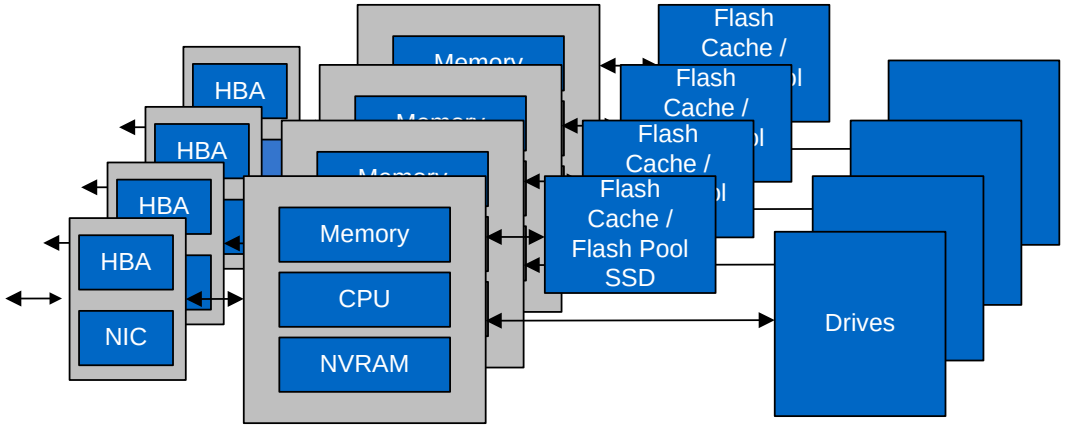
A NetApp storage system may be logically divided into three main areas when discussing performance. Those are connectivity, the system itself, and the storage subsystem. When we speak of connectivity, we refer to the network and HBA interfaces that connect the storage system to the clients and hosts. The system itself is the combination of CPU, memory, and NVRAM. Finally, the storage subsystem consists of the disks, and also Flash Cache™ and Flash Pool™ intelligent caching. The following picture logically represents a NetApp system.

Figure 2) High-level cluster node system architecture.



A system running clustered Data ONTAP consists of individual nodes joined together by the cluster interconnect. Every node in the cluster is capable of storing data on disks attached to it, essentially adding "copies" of the preceding architecture to the overall cluster. Clustered Data ONTAP has the capability to nondisruptively add additional nodes to the system to scale out both the performance and capacity of the system.

Figure 3) Scaling of the architecture.



2.3.1 Connectivity: NICs and HBAs

NICs and HBAs provide the connectivity to client, management, and cluster interconnect networks. Adding more or increasing the speed of NICs or HBAs can scale client network bandwidth.

2.3.2 Controller Subsystem: Memory, CPU, and NVRAM

Common to most systems, NetApp systems contain CPUs and some amount of memory, depending on the controller model. As with any computer, the CPUs serve as the processing power to complete operations for the system. Besides serving operating system functions for Data ONTAP, the memory in a NetApp controller also acts as a cache. Incoming writes are coalesced in main memory prior to being written to disk. Memory is also used as a read cache to provide extremely fast access time to recently read data.

NetApp systems also contain NVRAM. NVRAM is battery-backed memory that is used to protect in-bound writes as they arrive. This allows write operations to be committed safely without having to wait for a disk operation to complete and reduces latency significantly. High-availability (HA) pairs are created by mirroring NVRAM across two controllers.

Increasing the capacity of these components requires upgrading to a higher controller model. Clustered Data ONTAP allows nodes to be evacuated and upgraded nondisruptively to clients.

2.3.3 Storage Subsystem: Disks, Flash Cache, and Flash Pool

Spinning disk drives are the slowest components in the whole storage system. The typical response times for spinning disks are a few milliseconds. The performance of disk drives varies depending on the disk type and rotation speed: 7.2K RPM SATA disks have higher latency than 10K RPM SAS disks. Solid-state disks significantly reduce the latency at the storage subsystem. Ultimately, the type of disk needed for a specific application depends on capacity and performance requirements as well as the workload characteristics. For more information about disk drives, see TR-3838: Storage Subsystem Configuration Guide.

With the introduction of Flash Cache and Flash Pool, it is possible to combine the performance of solid-state flash technology with the capacity of spinning media. Flash Cache operates as an additional layer of read cache for the entire system. It caches recently read, or "hot," data for future reads. Flash Pool serves as a read cache similar to Flash Cache at the aggregate level. Flash Pool is also capable of offloading random overwrites that are later destaged to disk to improve write performance.

For more information about Flash Cache, see TR-3832: Flash Cache Best Practices Guide.

2.4 Data Storage and Retrieval

2.4.1 Cluster Operations

In clustered Data ONTAP, data stored or accessed does not need to reside on the node connected to the client. Data can be accessed directly or indirectly across the cluster through application requests, generally referred to as operations, often shortened to simply “ops." Operations can take on many forms, such as the commonly known read and write operation, and lesser known types often categorized as “metadata operations” or “other."

Generally speaking, clustered Data ONTAP is composed of four major architectural components:

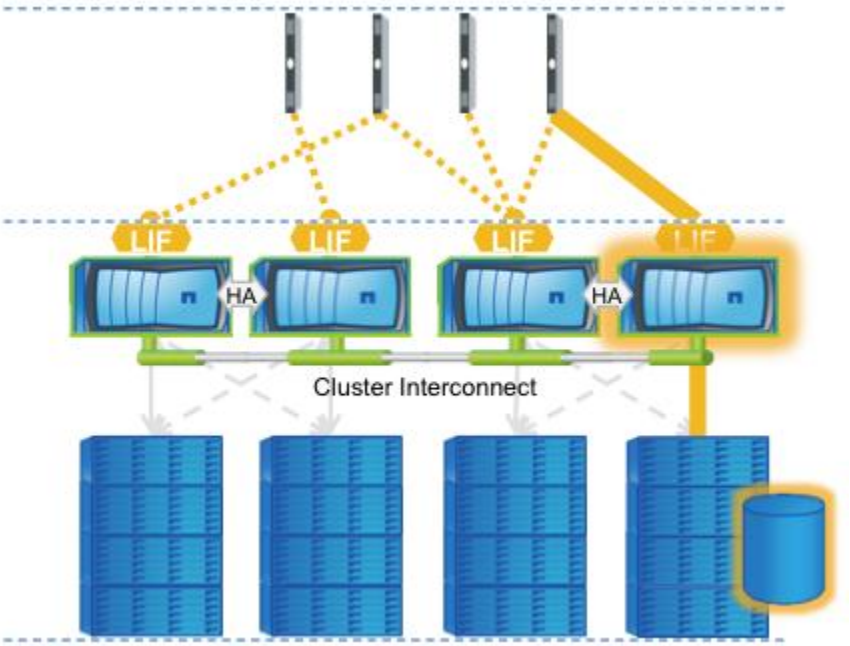
- **Network.** Transports the operation request and response to and from the application.
- **Cluster interconnect (indirect access only).** Transports the operation to the node that has responsibility to execute the operation.
- **Data access and layout.** Optimizes the execution of the operation requested in context with all other operations taking place (otherwise known as the WAFL® [Write Anywhere File Layout] system).
- **Disk.** Stores data to permanent media.

Operations can traverse each of these components across a cluster. The average amount of time an operation takes to traverse these components is the latency or response time metric.

Direct Data Access

Direct data access occurs when a client connected to a node accesses data stored on disks directly connected to that node. When accessing data in this fashion there is no traversal of the cluster interconnect. Note in Figure 4 that data flows directly to the disk component shown below the cluster interconnect component floating above. Different protocols behave differently and have different features when it comes to clustered Data ONTAP. The section on Protocol Considerations later discusses these differences.

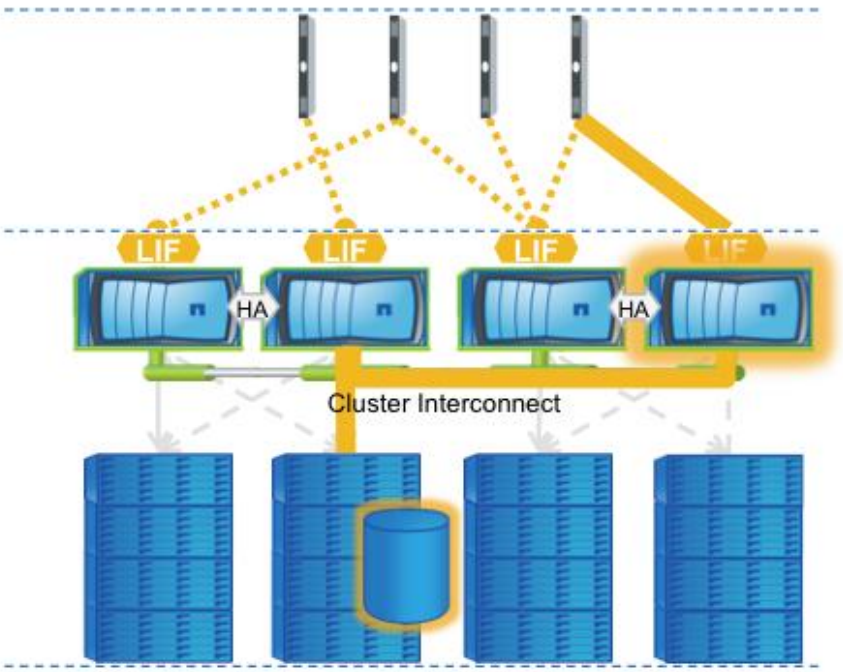
Figure 4) Direct data access on local node.



Indirect Data Access

Indirect data access occurs when a client accesses one node but the data is stored physically on another node. The node with which the client is communicating identifies where the data is stored and accesses the other node using the cluster interconnect and then serves the data back to the client. Indirect data access allows data to live physically on any node without the need to force clients to mount more than a single location to access the data.

Figure 5) Indirect data access to remote node.



Protocol Considerations

Accessing data directly on the node where it is stored reduces the amount of resources necessary to serve data and is ultimately the "shortest path" to where the data lives. Some of the protocols supported by clustered Data ONTAP have the ability to automatically provide direct data access. Independent of protocols, the management features of clustered Data ONTAP can be used to alter the data access path.

Certain protocols have the capability to automatically direct traffic to the node with direct data access. In the case of NAS protocols, NFS version 4 (NFSv4) can direct clients to local nodes through a variety of capabilities. NFSv4 referrals point the client to the directly attached node during mount. Another capability with NFSv4.1 is parallel NFS (pNFS). pNFS enables clients to connect to any node in the cluster for metadata work while performing direct data operations. To learn more about NFS capabilities in clustered Data ONTAP, read TR-4067: Clustered Data ONTAP NFS Best Practices and Implementation Guide and TR-4063: Parallel Network File System Configuration and Best Practices for clustered Data ONTAP. Similarly, the SMB 2.0 and 3.0 protocols support a feature called autolocation. This capability automatically directs a client to the direct node when mounting a share. More information is available in the Data ONTAP documentation.

In SAN environments, the ALUA protocol enables optimal pathing to a LUN. Even if volumes are moved around in the cluster, the host will always access the LUN through the optimal path. To learn more about using SAN with clustered Data ONTAP, read TR-4080: Best Practices for Scalable SAN in Clustered Data ONTAP.

2.4.2 Node Operations

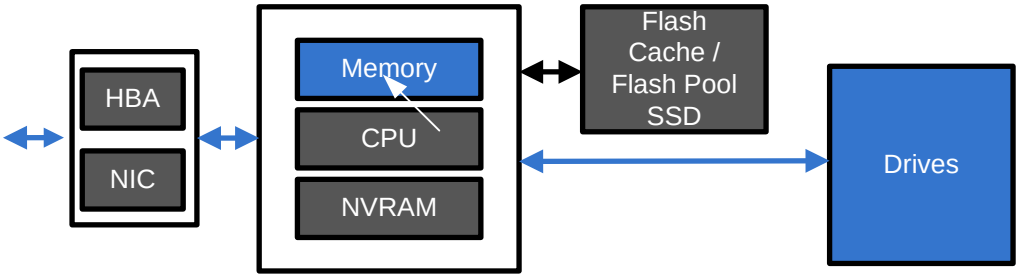
After an operation has been directed to the proper node, that node becomes responsible for completing the read or write operation. In this section, we examine how reads and writes are completed on a node and how the components within the storage system are used.

Reads

Recall the storage system architecture presented in section 2.3, Cluster Node System Architecture Overview; read operations can be serviced from memory, flash-based cache, or spinning disk drives. The workload characteristics and capabilities of the system determine where reads are serviced and how fast. Knowing where reads are serviced can help set expectations as to the overall performance of the system. In the following diagrams, components and links in blue highlight the activity described.

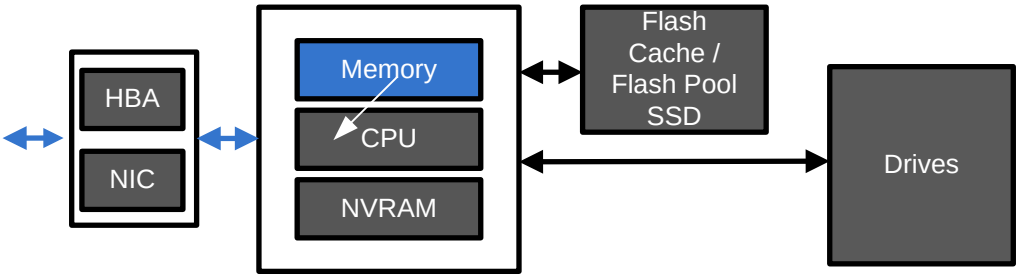
In the simple yet slowest case, read requests that are not cached anywhere are forced to come from disk. After being read from disk, the data is kept in main memory.

Figure 6) Read from disk.



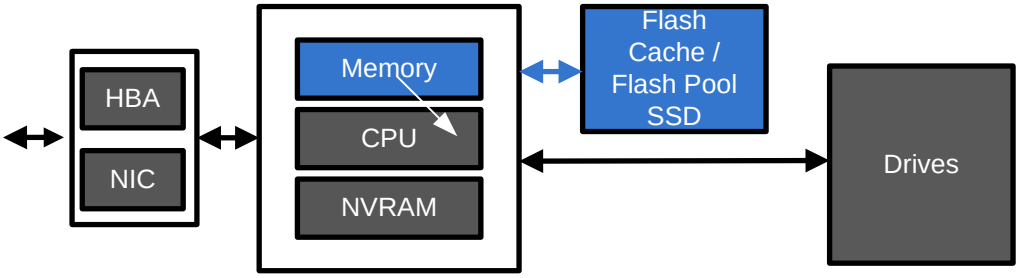
If this data is read again soon, it is possible for the data to be cached in main memory, making subsequent access extremely fast because no disk access would be required.

Figure 7) Read from memory.



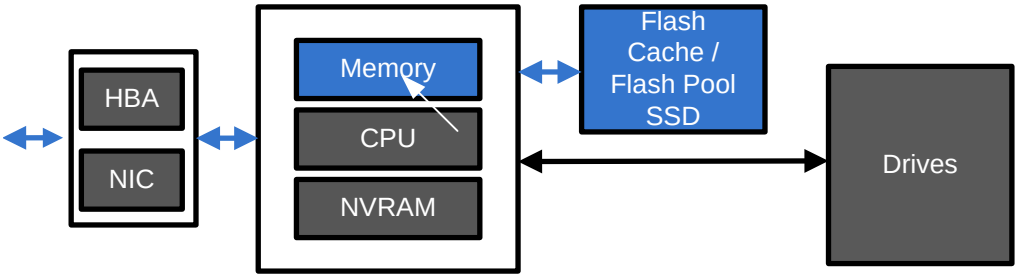
When more room is needed in the main memory cache, as is common with working sets larger than the buffer cache, the data is evicted. If Flash Cache or Flash Pool is in the system, that block could be inserted into the flash-based cache if it meets certain requirements. In general, only randomly read data and metadata are inserted into flash-based caches.

Figure 8) Write to flash.



After data is inserted, subsequent reads of this block unable to be serviced from the buffer cache would be served from the flash-based cache until they are evicted from the flash-based cache. Flash access times are significantly faster than those of disk, and adding cache in random read–intensive workloads can reduce read latency dramatically.

Figure 9) Read from flash.

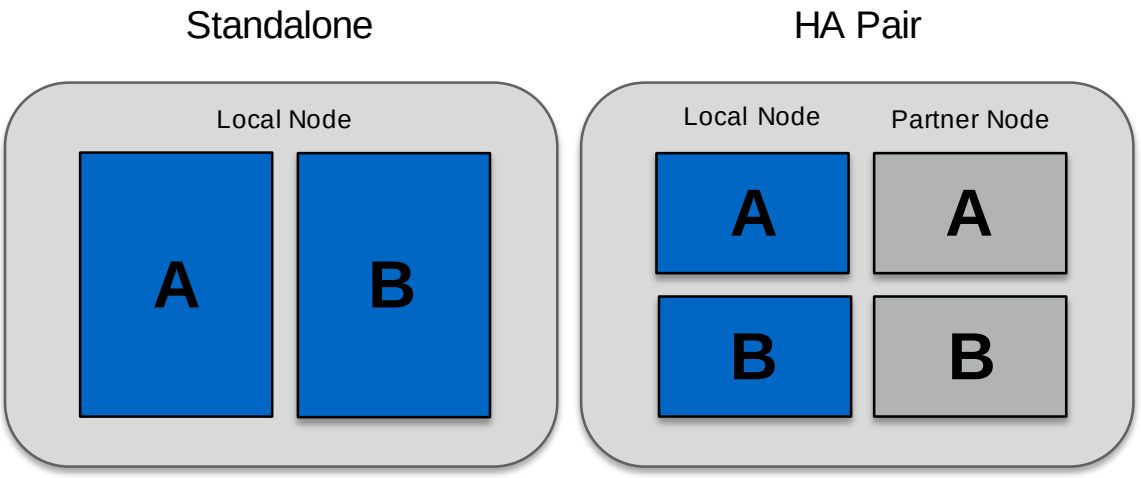


Incoming reads are continually being checked for access patterns. For some data access patterns, such as sequential access, Data ONTAP predicts which blocks a client may want to access prior to the client ever requesting. This "read-ahead" mechanism preemptively reads blocks off disk and caches them in main memory. These read operations are serviced at faster RAM speeds instead of waiting for disk when the read request is received.

Writes

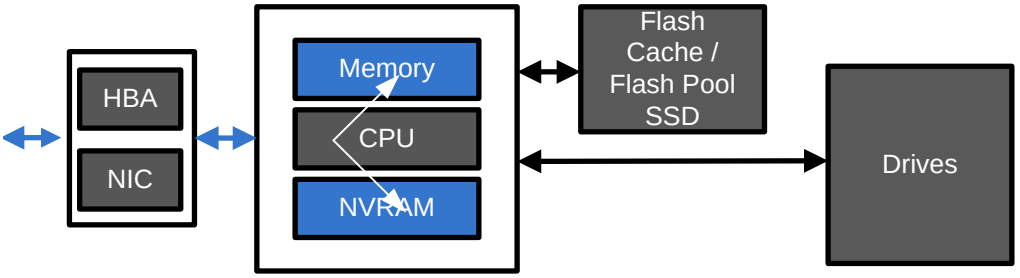
Next, consider how data is written to the storage system. For most storage systems, writes must be placed into a persistent and stable location prior to acknowledging to the client or host that the write was successful. Waiting for the storage system to write an operation to disk for every write could introduce significant latency. To solve this problem, NetApp storage systems use battery-backed RAM to create nonvolatile RAM (NVRAM) to log incoming writes. NVRAM is divided in half, and only one half is used at a time to log incoming writes. When controllers are in highly available pairs, half of the NVRAM is used to mirror the remote partner node's log, while the other half is used for logging local writes. The part that is used for logging locally is still split in half, just like a single node.

Figure 10) NVRAM segmenting: standalone node and HA pair configurations.



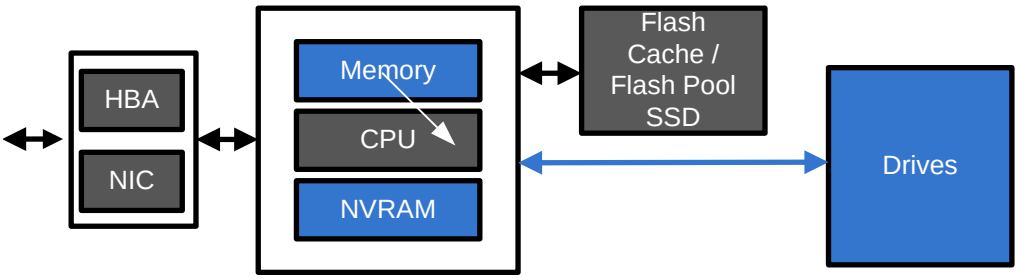
When a write enters a NetApp system, the write is logged into NVRAM and is buffered in main memory. After the data is logged in persistent NVRAM, the client is acknowledged. NVRAM is accessed only in the event of a failure.

Figure 11) Accepting a write.



At a later point in time, called a consistency point, the data buffered in main memory is efficiently written to disk. Consistency points can be triggered for a number of reasons, including time passage, NVRAM fullness, or system-triggered events such as a Snapshot® copy.

Figure 12) Consistency point.



In general, writes take a minimal amount of time, on the order of low milliseconds to submilliseconds. If the disk subsystem is unable to keep up with the client workload and becomes too busy, write latency can begin to increase. When writes are coming in too fast for the back-end storage, both sides of the NVRAM can fill up and cause a scenario called a back-to-back CP. This means that both sides are filled up, a CP

is occurring, and another CP will immediately follow on the current CP's completion. This scenario affects performance because the system can't immediately acknowledge the write as persistent because NVRAM is full and must wait until the operation can be logged. Improving the storage subsystem can most often alleviate the back-to-back CP scenario. Increasing the number of disks, moving some of the workload to other nodes, and considering flash-based caching can help solve write performance issues. Remember, only randomly overwritten data is written to the SSD portion of a Flash Pool aggregate and that Flash Cache is only a read cache but that offloading any type of operation can reduce disk utilization.

2.5 Controlling Workloads: Introduction to Quality of Service

NetApp storage quality of service (QoS) gives the storage administrator the ability to monitor and control storage object workloads that deliver consistent performance to meet application workload service objectives.

Consistent Performance

Storage QoS is a clustered Data ONTAP feature designed to help address the need for consistent workload performance. In environments without a QoS capability, unexpected workload increases will cause utilization to reach levels that may be unacceptable, with some low-priority workloads taking more than their share of resources while higher priority work is denied the resources it needs. The storage QoS feature allows specific maximums to be set for groups of workloads, setting an upper boundary on the amount of throughput they may consume. With this capability, you can retain resources for important work by restricting access to the less important workloads.

Workload Isolation

Similarly, some workloads may need to be isolated from other workloads in the cluster. Rogue workloads can consume an enormous amount of resources and reduce the available amount for others. The ability to monitor and then isolate rogue workloads creates value for the administrator in dynamic environments. Storage QoS accomplishes these tasks with a simple command line interface that allows settings to be configured for the cluster on the fly, without requiring an extended planning process or a complicated interface.

2.5.1 Storage QoS Concepts

Before discussing examples and use cases for storage QoS, it is important to understand some basic QoS concepts and terminology.

Workload

A workload is the set of I/O requests sent to one or more storage objects. In clustered Data ONTAP, QoS workloads include I/O operations and data throughput, and they are measured in IOPS and MB/s, respectively. IOPS workload measurement includes all client I/O, including metadata and disk I/O, regardless of I/O block size. I/O related to system processes is not counted in the IOPS measurement.

Storage Objects

A storage object is the entity on the controller to be assigned to a QoS policy group for monitoring and control. QoS storage objects can be any of the following:

- Storage virtual machines (SVMs), formerly called Vservers.
- FlexVol® volumes
- LUNs
- Files

Policies

QoS policies are behaviors to apply to a QoS policy group and its storage objects. In clustered Data ONTAP 8.2 or newer, you can define a QoS policy to impose a throughput limit on the storage objects in the QoS policy group. This throughput limit is applied collectively to the group. QoS policies may be configured to control IOPS or MB/s throughput. In addition, the QoS policy may be configured to `none` to allow the storage administrator to monitor the workload throughput of the storage objects in the QoS policy group without limiting the workload throughput.

Limits

As previously discussed, the storage administrator can control the workload throughput using IOPS or MB/s limits. When the workload throughput exceeds the QoS policy limit, the workload is reduced at the protocol layer. The storage administrator should expect the response time for I/O requests to increase while QoS throttles the workload. Occasionally, some applications may time out. This behavior is no different than when a system runs out of performance headroom. Throttling a workload in the protocol stack prevents it from consuming incremental cluster resources, thus freeing up resources for the other workloads deployed on the cluster.

When the QoS policy is configured to throttle IOPS or MBPS, the specified policy value is a hard limit. The storage administrator should be aware that the workload IOPS or MBPS throughput may exceed the value set in the QoS policy by up to 10% while the I/O operations are queued and throttled. As a general rule, the lower the QoS policy limits, the higher the deviation from the limit while the policy takes effect. I/O operations queued as a result of hitting the QoS policy limit do not affect cluster resources. QoS policies are applicable to all supported protocols, including NFS, SMB, SAN, iSCSI, and FCoE. Starting in clustered Data ONTAP 8.3, NFS 4.1 is supported.

Note: QoS is not compatible with NFS 4 prior to clustered Data ONTAP 8.3.

QoS is not compatible with pNFS in clustered Data ONTAP.

When to Use MB/s

For large block I/O workloads, NetApp recommends configuring the QoS policy using MB/s.

When to Use IOPS

For transactional workloads, NetApp recommends configuring the QoS policy using IOPS.

Policy Groups

QoS policy groups are collections of storage objects (that is, SVMs, volumes, LUNs, or files) to enable the storage administrator to monitor and control workload throughput. One QoS policy (behavior) can be assigned to a QoS policy group. The storage administrator can monitor storage object workloads by assigning the storage objects to a policy group without applying a QoS policy.

Note: Only one QoS policy may be applied to a QoS policy group.

Storage objects assigned to QoS policy groups are SVM (Vserver) scoped. This means that each QoS policy group may have only storage objects assigned to it from a single SVM. QoS policy groups support assignment of several FlexVol volumes, LUNs, and files within the same SVM. The I/O limits are applied collectively across the storage objects in a policy group and are not applied at an individual storage object level. Individual storage objects within a policy group are expected to consume resources using a fair-share methodology.

Note: The QoS policy throughput limit is applied to the *combined* throughput of all storage object workloads assigned to the policy group.

Nested storage objects cannot be assigned to the same or a different QoS policy group. For example, a VMDK file and its parent volume may not both be assigned to a QoS policy group.

Note: Nested storage objects may not both be assigned to the same or a different QoS policy group.

QoS policy group membership remains unchanged as storage objects are moved within the cluster. However, as previously discussed, storage objects cannot be nested. For example, if a VMDK file, which is part of a policy group, is moved to a different datastore (volume), which is already part of a policy group, then the VMDK file will no longer be assigned to the policy group.

Some environments may utilize NetApp FlexCache® functionality to enhance performance. When a FlexVol volume leveraging FlexCache is assigned to a QoS policy group, the FlexCache volume workload is included in the QoS policy group workload.

Monitor

Assigning storage objects to a QoS policy group without a QoS policy, or modifying an existing QoS policy limit to `none`, gives the storage administrator the ability to monitor the workload placed on those storage objects without limiting the workload throughput. In this configuration the storage administrator can monitor workload latency, IOPS, and data throughput. Storage QoS measures latency from the network interface to and from the disk subsystem.

Creating a QoS policy group to monitor workload latency and throughput:

```
Cluster1::> qos policy-group create -policy-group monitor_workload -vserver vserver1
```

Assigning volumes to a QoS policy group:

```
Cluster1::> vol modify -vserver vserver1 -volume vol1 -qos-policy-group monitor_workload
  (volume modify)

Volume modify successful on volume: vol1

Cluster1::> vol modify -vserver vserver1 -volume vol2 -qos-policy-group monitor_workload
  (volume modify)

Volume modify successful on volume: vol2
```

Assigning an SVM to a QoS policy group:

```
Cluster1::> vserver modify -vserver vserver2 -qos-policy-group vserver2_qos_policy_group
```

Displaying the QoS policy group configuration and the number of workloads assigned to the policy group:

```
Cluster1::> qos policy-group show
Name             Vserver     Class        Wklds Throughput
---------------- ----------- ------------ ----- ------------
monitor_workload vserver1    user-defined 2     0-INF
vol1_qos_policy  vserver1    user-defined 0     0-500IOPS
vol2_qos_policy  vserver1    user-defined 0     0-100MB/S
3 entries were displayed.
```

Note: QoS policy groups that do not have a throughput limit are shown with `0-INF`. This represents an infinite QoS policy limit.

Viewing the QoS policy group latency statistics:

```
cluster1::> qos statistics latency show
Policy Group     Latency    Network    Cluster    Data     Disk     QoS
-------------- ---------- ---------- ---------- -------- -------- -------
-total-              16ms         6ms         2ms       3ms       4ms     1ms
```

Viewing the QoS policy group performance statistics:

```
cluster1::> qos statistics performance show
```

```
Policy Group            IOPS      Throughput   Latency
-------------------- -------- --------------- ----------
-total-                12224       47.75MB/s   512.45us
rogue_policy            7216       28.19MB/s   420.00us
prevent_policy          5008       19.56MB/s    92.45us
```

Autovolumes

Starting in clustered Data ONTAP 8.3, you can view the QoS statistics for volumes in the cluster without setting a QoS policy group. This functionality is called autovolumes. Autovolumes track the volume-level QoS statistics and requires advanced privileges.

```
cluster1::> qos statistics volume latency show
Policy Group     Latency    Network    Cluster     Data      Disk     QoS
-------------- ---------- ---------- ---------- -------- -------- ------
vol1-wid12         170ms       15ms        0ms      10ms    120ms     0ms
vol3-wid302         30ms        5ms        0ms      25ms      0ms     0ms
cluster1::> qos statistics volume performance show
Policy Group            IOPS      Throughput    Latency
-------------------- -------- --------------- ----------
vol3-wid302              120         127MB/s   170.34ms
vol7-wid1441              30          10KB/s    30.13ms
```

Note: Autovolume commands require advanced privileges.

For more information on the QoS policy group monitoring commands, see the Clustered Data ONTAP 8.3 Commands: Manual Page Reference.

Control

QoS policy groups with a policy can control and limit the workloads of the storage objects assigned to the policy group. This capability gives the storage administrator the ability to manage and, when appropriate, throttle storage object workloads. In clustered Data ONTAP 8.2 or newer, the storage administrator can control I/O and data throughput. When a policy is configured, the storage administrator can continue to monitor the latency, IOPS, and data throughput workloads of storage objects.

Creating a QoS policy group to control workload IOPS:

```
Cluster1::> qos policy-group create -policy-group vol1_qos_policy_group -max-throughput 500iops
            -vserver vserver1
```

Creating a QoS policy group to control workload data throughput:

```
Cluster1::> qos policy-group create -policy-group vol2_qos_policy_group -max-throughput 1000MBPS
            -vserver vserver1
```

Assigning volumes to QoS policy groups:

```
Cluster1::> vol modify -vserver vserver1 -volume vol1 -qos-policy-group vol1_qos_policy_group
  (volume modify)

Volume modify successful on volume: vol1

Cluster1::> vol modify -vserver vserver1 -volume vol2 -qos-policy-group vol2_qos_policy_group
  (volume modify)

Volume modify successful on volume: vol2
```

Assigning LUNs to QoS policy groups:

```
Cluster1::> lun modify -vserver vserver1 -lun lun1 -vol vol2
            -qos-policy-group lun_qos_policy_group
```

Assigning files to QoS policy groups:

```
Cluster1::> volume file modify -vserver vserver1 -vol vol2 –file log.txt
            -qos-policy-group file_qos_policy_group
```

Displaying the QoS policy group configuration and the number of workloads assigned to the policy group:

```
Cluster1::> qos policy-group show
Name              Vserver     Class         Wklds Throughput
----------------- ----------- ------------ ----- ------------
monitor_workload vserver1    user-defined 0     0-INF
vol1_qos_policy_group
                  vserver1    user-defined 1     0-500IOPS
vol2_qos_policy_group
                  vserver1    user-defined 1     0-100MB/S
3 entries were displayed.
```

For more information on the QoS policy group monitoring commands, see the Clustered Data ONTAP 8.3 Commands: Manual Page Reference.

Storage QoS Summary

The storage QoS capability in NetApp clustered Data ONTAP 8.2 or newer enables customers to increase utilization of storage resources by consolidating multiple workloads in a single shared storage infrastructure, while minimizing the risk of workloads affecting each other's performance. Administrators can prevent tenants and applications from consuming all available resources in the storage infrastructure, improving the end-user experience and application uptime. In addition, predefining service-level objectives allows IT to provide different levels of service to different stakeholders and applications, ensuring that the storage infrastructure continues to meet the business needs.

Storage QoS adds new capabilities for the storage administrator to monitor and control user workloads. Following is a brief summary of the QoS functionality delivered starting in clustered Data ONTAP 8.2:

- Monitor and manage storage object workloads
- Control I/O and data throughput workloads on SVMs, volumes, LUNs, and files
- Provide multiprotocol support, including SMB, SAN, iSCSI, FCoE, NFS
- Provision policy groups in Workflow Automation (WFA) 2.1 and newer
- Provide QoS support for V-Series

However, there are a few caveats to remember when you consider QoS:

- QoS is not supported on Infinite Volumes.
- Alerts may not be configured for QoS.
- QoS does not provide workload guarantees.
- QoS is not supported with pNFS.

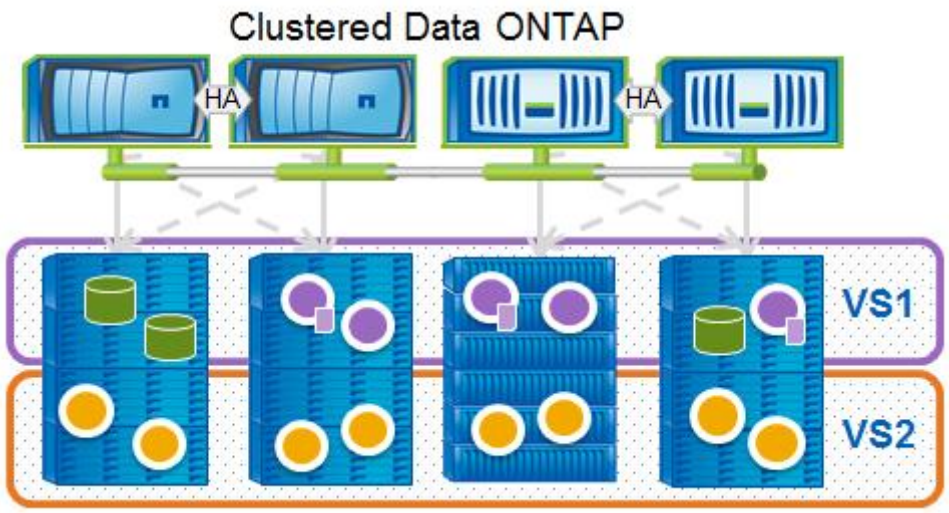
Table 1) QoS limits.

QoS Feature Area	Maximum	
	Per Node	Per Cluster
QoS policy groups supported	3,500	3,500
Number of controllers supported by QoS	1	8
Storage objects assigned to a QoS policy group	10,000	10,000

2.5.2 Examples of Using Storage QoS

Storage QoS will have many applications for the storage administrator. The following are a few scenarios that illustrate QoS capabilities. The first use case is an example in which the storage administrator throttles a "rogue" workload that is affecting other workloads. The second scenario describes how a storage administrator may prevent runaway (or rogue) workloads by proactively setting QoS policies. The final use case looks at managing workloads so that service providers can meet their service-level agreements (SLAs).

Figure 13) Storage QoS.



Reactively Respond

In this scenario the storage administrator has not applied any storage objects to a QoS policy group. By default, Data ONTAP treats all storage objects on a best-effort basis. However, one of the storage objects (that is, a volume) has a rogue workload affecting the performance of other workloads on the system. Using Data ONTAP `statistics` and `qos statistics` commands, the storage administrator can identify the rogue workload. After the rogue workload is identified, the storage administrator can use storage QoS to isolate the workload by assigning it to a QoS policy group and applying a throughput limit.

Throttle Rogue Workloads

After identifying the rogue workload, the storage administrator creates a new QoS policy group and sets the I/O throughput limit to 1,000 IOPS.

```
Cluster1::> qos policy-group create vol1_rogue_qos_policy -max-throughput 1000iops
            -vserver vserver1
```

To view the QoS policy group configuration, the storage administrator may use the `qos policy-group show` command.

```
Cluster1::> qos policy-group show -policy-group vol1_rogue_qos_policy

              Policy Group Name: vol1_rogue_qos_policy
                        Vserver: vserver1
                           Uuid: a20df2c2-c19a-11e2-b0e1-123478563412
             Policy group class: user-defined
                Policy Group ID: 102
             Maximum Throughput: 1000IOPS
            Number of Workloads: 0
              Throughput Policy: 0-1000IOPS
```

Next, the offending volume is assigned to the QoS policy group to begin throttling the rogue workload. The `modify` option of storage objects is used to assign an existing storage object to a QoS policy group.

```
Cluster1::> volume modify -vserver vserver1 -volume vol1 -qos-policy-group vol1_rogue_qos_policy
```

The storage administrator can verify that the volume has been assigned to the QoS policy group using the `volume show` command.

```
Cluster1::> volume show -vserver vserver1 -volume vol1 -fields qos-policy-group
vserver  volume qos-policy-group
-------- ------ --------------------
vserver1 vol1   vol1_rogue_qos_policy
```

Proactively Prevent Runaway Workloads

This is a scenario in which the storage administrator proactively sets a QoS policy group for the storage objects to prevent the impact of new, and possibly runaway, workloads. This situation may arise in a large virtualized environment in which the storage administrator needs to prevent a development or test application from affecting other production applications.

Apply Limits Before a Problem Occurs

The first step is to create a QoS policy group and apply a throughput limit.

```
Cluster1::> qos policy-group create -policy-group vmdk_13_qos_policy_group
            -max-throughput 100iops -vserver vserver1
```

After the QoS policy group has been created, the storage objects are assigned to the policy group. It is important to remember that the QoS limit is applied to the aggregate throughput of all storage objects in the QoS policy group.

```
Cluster1::> volume file modify -vserver vserver1 -vol vol2 –file vmdk-13.vmdk
            -qos-policy-group vmdk_13_qos_policy_group
```

Lastly, the storage administrator should monitor the storage object workload and adjust the policy as needed. Changes to the policy throughput limit can be completed quickly without affecting other workloads.

```
Cluster1::> qos statistics performance show
Policy Group            IOPS      Throughput   Latency
--------------------- -------- --------------- ----------
-total-                    867       47.75MB/s   512.45us
vol1_rogue_qos_policy      769       28.19MB/s   420.00us
vmdk_13_qos_policy_group    98       19.56MB/s    92.45us
```

After reviewing the required IOPS resources for vmdk-13 with engineering, the storage administrator agrees to increase the policy throughput limit to 200 IOPS.

```
Cluster1::> qos policy-group modify -policy-group vmdk_13_qos_policy_group
            -max-throughput 200iops
```

Isolate Tenants with Per-SVM Throughput Limits

In our final use case we look at a service provider who needs to isolate customer workloads to meet the service-level agreements. A new SVM is created in the cluster for each customer, and the service provider must enable workloads to be controlled based on the SLA level. This service provider has three SLA levels—Bronze, Silver, and Gold—corresponding to the maximum data throughput allowed.

Table 2) SLA levels.

SLA Level	Data Throughput
Bronze	100MBPS
Silver	200MBPS
Gold	400MBPS

After the service provider determines the SLA throughput limits, the storage administrator can create the storage QoS policy group with the appropriate limit and assign the SVM storage object to the policy group. For this example, we use three fictional service provider customers—Acme, Bravos, and Trolley—who have purchased the Bronze, Silver, and Gold service levels, respectively.

Create a policy group with the appropriate throughput limit (determined by service level) and assign the SVM for each customer to the policy group:

```
Cluster1::> qos policy-group create -policy-group acme_svm_bronze -max-throughput 100MBPS
            -vserver acme_svm

Cluster1::> qos policy-group create -policy-group bravos_svm_silver -max-throughput 200MBPS
            -vserver bravos_svm

Cluster1::> qos policy-group create -policy-group trolley_svm_gold -max-throughput 400MBPS
            -vserver trolley_svm
```

Apply the SVM for each customer to the QoS policy group:

```
Cluster1::> vserver modify -vserver acme_svm -qos-policy-group acme_svm_bronze

Cluster1::> vserver modify -vserver bravos_svm -qos-policy-group bravos_svm_silver

Cluster1::> vserver modify -vserver trolley_svm -qos-policy-group trolley_svm_gold
```

2.6 Performance Management with Clustered Data ONTAP

Ensuring the performance of a system is essential throughout its deployed lifecycle. Some applications have more stringent performance requirements than others, but performance is never a nonrequirement. Performance management starts with the first request for workload characteristics and continues until the system is decommissioned. Being able to understand workloads, identify problems, and relate them back to the system's operation is essential to achieving performance goals.

This section introduces the capabilities of Data ONTAP and other NetApp software to complete performance management functions, including looking at statistics and using features to alter the performance of the system or workloads.

2.6.1 Basic Workload Characterization

As mentioned earlier, the workload characteristics and system architecture ultimately define the performance of the system. Also mentioned earlier were the storage QoS capabilities available in clustered Data ONTAP 8.3. You can use the statistics generated by the QoS Command line interface (CLI) to monitor and gain a basic understanding of workloads in the system. These insights can then be used to confirm initial sizing estimations, refine sizing forecasts, or simply set performance baselines and expectations. When reviewing this data, keep in mind the relationships introduced in section 2.2.

The following command shows the workload characteristics of the busiest workloads on the system:

```
Cluster1::> qos statistics workload characteristics show
Workload          ID     IOPS      Throughput      Request size    Read  Concurrency
--------------- ------ -------- ---------------- --------------- ------- -----------
-total-              -     5076        37.88MB/s            7825B     65%           2
```

```
volume_a          14368      4843        37.82MB/s           8189B      68%            2
...
```

Although this is just an example, the preceding data shows that the system is processing about 5,000 IOPS, roughly 8KB in size, with 65% being reads.

Entire policy group characteristics can be viewed by eliminating the workload part of the command, as in the following:

```
Cluster1::> qos statistics characteristics show
```

These examples are basic, and more statistics than are presented here are available in Data ONTAP.

Overview of Application Workload Characteristics

Ultimately every production workload is unique due to the many variables that contribute to individual behavior. However, to properly set expectations and for instructional purposes, some basic generalizations about application workloads are presented here.

Write Work

As mentioned earlier, clustered Data ONTAP is highly optimized to efficiently perform writes by taking advantage of NVRAM, system memory, and consistency point logic to optimize the on-disk layout of blocks written. This reduces the effects of writing to and later reading from slower disk storage mediums. Thus sequential and random writes are, for all practical purposes, instantly recorded in memory, permanently logged in local and partner NVRAM, and the response immediately sent to the application. Then based on time thresholds or NVRAM usage thresholds, writes will be flushed to a slower persistent medium while Data ONTAP continues to service user operations. Exceptions can occur when unexpected situations are encountered, such as CPU or disk utilization issues causing resource constraints, excessive loads (along with concurrency) causing back-to-back CPs, or file system disk layout issues resulting in unnecessary disk I/O. None of these exceptions should occur in a properly operating and designed system.

Sequential Read Work

The Data ONTAP read-ahead engine detects common sequential workload read patterns to efficiently cache data before it is requested from disk. This in combination with the previously written layout optimizations contributes to greatly reducing delays associate with disks. Thus workloads with highly sequential read patterns, given adequate resources, should experience low service times by avoiding costly disk accesses.

Random Read Work

Some workloads are inherently more difficult to handle than others. Repeated random access for reasonably sized working sets are rarely a problem provided most of the data fits in caches. These workloads usually experience a large percentage of cache hits, and thus fewer disk reads, and experience low average service times. However, caches are shared resources and can be oversubscribed when shared by too many workloads. In addition, random read workloads with large working sets and even larger datasets make it very difficult or even impossible to predict what data will be needed. Thus, under some unusual circumstances, this causes the storage system to frequently reach for data on a slow disk medium, increasing service times.

Indirect I/O Work

When considering indirect workloads, it is tempting to conclude that the cluster interconnect is a potential source of additional service delay (or latency). However, when observed under normal operating conditions, the actual effects are minimal and contribute negligible delays in service time.

2.6.2 Observing and Monitoring Performance

Monitoring performance avoids potential problems and helps determine whether the system can handle additional load. Latency is used as a leading indicator for performance issues. In other words, if there is no latency issue, there is no performance issue. Corroborating metrics for latency include throughput and resource utilizations. This means that if unacceptable latency is observed alongside increased throughput, a workload may have grown and thus be saturating a resource in the system, confirmed by resource utilization. Low throughput is not necessarily a problem, because clients and applications may simply not be requesting that work be done. Ideally, workload objects are the best to monitor because they likely correspond to an application or tenant. Other abstractions are also useful to monitor, including the volume abstraction.

Data ONTAP collects statistics that can be monitored through graphical tools as well as through the cluster CLI and APIs. The following subsections introduce ways to monitor performance metrics using NetApp tools and on-box features. In advanced or more complex environments, these CLI commands and related APIs from the NetApp SDK can be used to create custom monitoring tools.

OnCommand Performance Manager (OPM)

OnCommand Performance Manager (OPM), an integrated component of OnCommand Unified Manager, is designed for clustered Data ONTAP. OPM Version 1.1 is made available with clustered Data ONTAP 8.3. OPM is designed to eliminate much of the complexity surrounding performance management and will automatically alert administrators when significant performance incidents occur.

Best Practice
It is highly recommended that OPM monitor all clustered Data ONTAP systems.

Simplicity is achieved through minimizing performance configuration settings and automation of typically complex tasks. Upon pointing OPM to a cluster management interface, OPM will automatically:

- Discover storage objects of interest
- Establish performance baselines and thresholds
- Retain 90 days of performance data at five-minute intervals
- Detect performance incidents and alert administrator through integration with OnCommand Unified Manager and e-mail
- Identify root cause performance degradation by correlating source of resource contention
- Suggest remediation tasks to resolve issue
- Export retained data for external monitoring, reporting, and auditing

As a simple example, the screen shot in Figure 14 shows a previously encountered incident where the volume under consideration was identified by OPM as a victim.

Figure 14) OPM victim workload with annotations 1 and 2.

The sample screen shot in Figure 14 depicts an OPM victim incident with:

- Increased workload latency (or response time). The latency graph plots the point in time (red dot) when the metric exceeded the automatically established threshold (gray bands). The line color of the graph line also changes to red and remains for the duration of the incident.
- Lower workload throughput (or operations) correlates with incident detection.

More information about OPM 1.1 can be found at the official NetApp Support site documentation portal in the OnCommand Performance Manager 1.1 User Guide.

More information about OPM integration with OnCommand Unified Manager can be found in the OnCommand Unified Manager 6.2 Administration Guide. More information about OPM incident detection and analysis appears later in this document in Performance Management with OnCommand Performance Manager.

Storage QoS CLI

Starting in clustered Data ONTAP 8.2, the QoS throughput policy is configured by setting either an I/O limit (IOPS) or a data throughput limit (bytes per second). Table 3 provides a list of available QoS CLI labels.

Table 3) QoS throughput labels.

QoS Storage Unit	QoS CLI Labels
IOPS	IOPS, iops, io/s
Bytes/second	Mb/s, MB/s, mb/s, MB/S, MBPS, mbps, B/s, B/S, b/s, bps

Note: The data throughput limit can only be specified in bytes/second, including megabytes/second.

Observing Throughput and Latency

Workload-level statistics are available after storage objects are assigned to a QoS policy group. These statistics can be displayed using the QoS statistics CLI commands.

As discussed earlier, throughput and latency are important observable metrics. Similar to workload characteristics, the throughput and latency of the system, policy group, or workloads can be determined by using the following command:

```
Cluster1::> qos statistics workload performance show
Workload           ID     IOPS       Throughput    Latency
---------------- ------ -------- ---------------- ----------
-total-             -     5060        37.97MB/s   492.00us
volume_a        14368     4847        37.86MB/s   510.00us
...
```

More detailed latency information is also available by looking at the output from the following command:

```
Cluster1::> qos statistics workload latency show
Workload            ID    Latency    Network    Cluster      Data       Disk       QoS
---------------- ------ ---------- ---------- ---------- ---------- ---------- ----------
-total-              -   608.00us   270.00us         0ms   148.00us   190.00us         0ms
volume_a         14368   611.00us   270.00us         0ms   149.00us   192.00us         0ms
```

The output describes the latency encountered at the various components in the system discussed in previous sections. Using this output, it's possible to observe where most of the latency is coming from for a specific workload:

- **Latency.** Refers to the total latency observed.
- **Network.** The amount of latency introduced by the network-level processing in Data ONTAP.
- **Cluster.** The amount of latency introduced by the cluster interconnect.
- **Data.** The amount of latency introduced by the system, except latency from the disk subsystem.
- **Disk.** The amount of latency introduced by the disk subsystem. Note that any reads that were serviced by the WAFL cache will not have a disk latency component because those operations did not go to disk.
- **QoS.** The amount of latency introduced by queuing by QoS if throughput limits have been established.

Observing Resource Utilizations

QoS also enables users to view disk and CPU utilizations for a policy group or workload. This output can help indicate which workloads are utilizing resources the most and can aid in identifying the workloads that could be considered bullies. For instance, if workload A is a very important workload and has high latencies from the previously introduced output, and in the resource utilizations output you notice workload B is using a lot of a system resource, the contention between workload A and workload B for the resource could be the source of the latency. Setting a limit or moving workload B could help alleviate workload A's latency issue. Resource utilizations are provided on a per-node basis.

```
Cluster1::> qos statistics workload resource cpu show -node Node-01
Workload           ID   CPU
---------------- ----- -----
-total- (100%)      -   29%
volume_a        14368   12%
System-Default      1   10%
...
```

```
Cluster1::> qos statistics workload resource disk show -node Node-01
Workload           ID  Disk  No. of Disks
---------------- ------ ----- ------------
-total-             -    4%           29
volume_a        14368    5%           22
System-Default      1    1%           23
...
```

Viewing Cluster-Level and Node-Level Periodic Statistics

Clustered Data ONTAP includes statistics beyond those presented in the QoS CLI statistics. One command to get an overall view into the cluster's state is `statistics show-periodic`. The output from this command provides details about the number of operations being serviced and additional clusterwide resource utilizations. Looking at an individual node's state is also possible.

Note: Use Ctrl-C to stop scrolling statistics and print a summary.

This command should be run in "advanced" privilege to get more information:

```
TestCluster::> set -privilege advanced

Warning: These advanced commands are potentially dangerous; use them only when directed to do so
by NetApp personnel.
Do you want to continue? {y|n}: y
```

The following example shows clusterwide performance. The output is too wide to fit in the document, so it is divided between two output blocks.

```
TestCluster::*> statistics show-periodic
cluster:summary: cluster.cluster: 6/7/2013 18:27:39
  cpu  cpu    total                      fcache    total    total data     data     data cluster …
  avg busy      ops  nfs-ops cifs-ops      ops     recv     sent busy     recv     sent    busy …
 ---- ---- -------- -------- -------- -------- -------- -------- ---- -------- -------- ------- …
  58%  91%    17687    17687        0        0    156MB    133MB  59%   68.7MB   47.3MB       4% …
  65%  92%    18905    18905        0        0    199MB    184MB  84%    103MB   74.9MB       6% …
  54%  86%    17705    17705        0        0    152MB    132MB  58%   68.9MB   47.2MB       4% …
cluster:summary: cluster.cluster: 6/7/2013 18:27:47
  cpu  cpu    total                      fcache    total    total data     data     data cluster …
  avg busy      ops  nfs-ops cifs-ops      ops     recv     sent busy     recv     sent    busy …
 ---- ---- -------- -------- -------- -------- -------- -------- ---- -------- -------- ------- …
Minimums:
  54%  86%    17687    17687        0        0    152MB    132MB  58%   68.7MB   47.2MB       4% …
Averages for 3 samples:
  59%  89%    18099    18099        0        0    169MB    150MB  67%   80.3MB   56.5MB       4% …
Maximums:
  65%  92%    18905    18905        0        0    199MB    184MB  84%    103MB   74.9MB       6% …
```

```
… continued

… cluster  cluster     disk     disk     pkts     pkts
…    recv     sent     read    write     recv     sent
…  -------- -------- -------- -------- -------- --------
…  87.6MB   86.3MB   96.4MB    139MB    87861    75081
…  96.2MB    109MB    108MB    261MB   127944   111190
…  84.0MB   85.4MB   69.6MB    101MB    87563    75402

… cluster  cluster     disk     disk     pkts     pkts
…    recv     sent     read    write     recv     sent
…  -------- -------- -------- -------- -------- --------

…  84.0MB   85.4MB   69.6MB    101MB    87563    75081

…  89.3MB   93.8MB   91.6MB    167MB   101122    87224

…  96.2MB    109MB    108MB    261MB   127944   111190
```

The following command shows a single node. The output is similar to the previous example, but it is for a single node.

```
TestCluster::*> statistics show-periodic -object node -instance node -node Node-01
```

Note: When reviewing CPU information in Data ONTAP, CPU AVG is a better indicator of overall CPU utilization compared to CPU BUSY.

Monitoring Workload Performance from Command Line Interface (CLI)

Clustered Data ONTAP provides a large set of commands for monitoring performance. This section describes how to use a small set of commands to gain insight into workload performance on a clustered Data ONTAP system. It is important to note that as a performance primer, the information presented here is for monitoring and instructional purposes only. This should not be mistaken for troubleshooting workflows or advanced diagnostics.

The recommended method for monitoring performance on a clustered Data ONTAP system is to use volume workloads. Clustered Data ONTAP makes this easy through the use of the newly introduced autovolumes feature (see the section on autovolumes). At this point, it is worth noting a distinction between volume objects versus volume workloads. A volume object is the target of a volume workload. A volume workload object contains performance metrics to record the time (latency) and resource utilization (processing) of operations occurring across the cluster. In clustered Data ONTAP systems, an operation enters the cluster on one node, can get transported across the cluster interconnect, and can get serviced on a different node. The total time it takes to complete the operation is otherwise known as operation residency time and is recorded in the volume workload as a latency metric. The total amount of resource used to complete the operation is recorded as utilization metrics. The total amount of work completed over a given time is recorded as throughput metrics. The metrics are expressed in average units per second when depicting throughput and in average units per operation when depicting latency. Volume objects are irrelevant in this monitoring scenario and are used for alternative workflows beyond the scope of this document.

Thus volume workloads are the primary mechanism for gaining visibility into the service times provided to applications using a volume. The recommended commands to use for CLI performance monitoring are summarized in Table 4.

Table 4) Recommended performance-monitoring commands.

Command	Description
qos statistics volume performance show	View volume workload operations per second, data throughput, and clusterwide latency
qos statistics volume characteristics show	View volume workload operation payload size and concurrency
qos statistics volume latency show	View clusterwide latency contribution at the cluster component level
qos statistics volume resource cpu show	View CPU resource consumption attributed to volume workload
qos statistics volume resource disk show	View disk resource utilization attributed to volume workload

The following examples provide general guidance on interpreting command output for each of the individual commands listed in Table 4.

```
ontaptme-fc-cluster::*> qos statistics volume performance show
Workload            ID     IOPS       Throughput    Latency
--------------- ------ -------- ---------------- ----------
-total-              -     4124        29.80MB/s  1198.00us
bobvol1-wid13..  13891     2474         6.77MB/s  1015.00us
bobvol5-wid9864   9864     1650        23.03MB/s  1472.00us
```

Use the preceding command to view overall latency on volume workloads and get a sense of work performed on the cluster.

- **Workload.** Concatenation of volume name and internal workload identifier.

- **ID.** Internal workload identifier.
- **IOPS.** Average number of operations processed every second.
- **Throughput.** Average amount of payload data moved into and out of the workload every second.
- **Latency.** Average operation residency time into and out of the workload.
- **-total-.** For throughput metrics, the sum of averages of all workloads on the cluster. For latency, the average of average latencies on the cluster.

```
ontaptme-fc-cluster::*> qos statistics volume characteristics show
Workload              ID     IOPS       Throughput Request Size Read Concurrency
--------------- ------ -------- ---------------- ------------ ---- -----------
-total-                -     4179         30.95MB/s         7766B  45%           5
bobvol1-wid13..   13891     2417          6.58MB/s         2856B  35%           3
bobvol5-wid9864    9864     1761         24.37MB/s        14513B  59%           2
```

More can be learned about the offered load presented to the volume using the QoS volume characteristics command shown earlier. In particular, the concurrency calculation shows the level of application offered load in relation to cluster consumption of that load. Both these factors individually are highly complex and do not provide enough information to draw any major conclusions. However, concurrency does provide information about application behavior such as the request arrival rate:

- **Workload.** Concatenation of volume name and internal workload identifier.
- **ID.** Internal workload identifier.
- **IOPS.** Average number of operations processed every second.
- **Request size.** Calculation of throughput divided by IOPS. Given that all the metrics available are averages, this is the best that can be done. For more detailed request size information, a histogram will be required.
- **Read.** Percentage of workload that is read operations. The remaining percentage is write operations for SAN protocols and is the sum of writes and metaoperations (or other) for NAS protocols.
- **Concurrency.** Product of latency and IOPS; see "Little's Law: A Relationship of Throughput, Latency, and Concurrency." This is the number of operations resident in the cluster being serviced at a given point in time.
- **-total-.** For throughput and concurrency metrics, the sum of averages of all workloads on the cluster. For remaining metrics, the average of averages on the cluster.

In the following example, the output is too wide to fit in the document, so it is divided between two output blocks.

```
ontaptme-fc-cluster::*> qos statistics volume latency show
Workload              ID    Latency    Network    Cluster       Data       Disk
--------------- ------ ---------- ---------- ---------- ---------- ----------
-total-                -  1391.00us    74.00us    41.00us    56.00us  1203.00us
bobvol5-wid9864    9864     2.17ms    90.00us   162.00us    69.00us     1.83ms
bobvol1-wid13..   13891  1126.00us    69.00us        0ms    51.00us   988.00us
```

```
..continued
        QoS       NVRAM
 ---------- ----------
        0ms    17.00us
        0ms    14.00us
        0ms    18.00us
```

The volume latency command breaks down the latency contribution of the individual clustered Data ONTAP components (see section 2.4.1 above). Among the monitoring commands presented here, this is possibly the most useful in that it provides visibility into workload internals across the cluster. It is worth repeating that latency is the equivalent of operation residency time and is the sum of operation wait time

and execution time for a given component. In the preceding example, it can be seen that workload bobvol5 is an indirect workload averaging 2.17ms latency. The largest portion of that latency is attributed to the 1.82ms disk contribution. The remaining component contributions are most likely CPU execution time and network and cluster interconnect transmission delay and account for very little of the total latency. This command does not provide enough information to know every detail. However, when considering a singular workload under normal operating conditions, execution time will almost always be far less significant in relation to wait time (in this example disk wait time):

- **Workload.** Concatenation of volume name and internal workload identifier.
- **ID.** Internal workload identifier.
- **Latency.** Overall average operation residency time into and out of the cluster.
- **Network.** Average latency contribution of the server (or client) facing transport component. This includes the delay introduced by front-end SAN or NAS transport stacks.
- **Cluster.** Average latency contribution of the cluster interconnect transport component. This includes the delay introduced by all cluster interconnect transport protocols.
- **Data.** Average latency contribution of the clustered Data ONTAP proprietary file system known as WAFL. After the operation has been transported by the underlying protocols, it will be processed by WAFL and response returned back to the protocols for delivery as rapidly as possible.
- **Disk.** Average latency contribution of the physical disks.
- **QoS.** Applicable only when QoS policy limits are in place. When actively limiting, this is the average latency penalty incurred to enforce the user-defined policy limit.
- **NVRAM.** Average latency incurred to replicate write operations in NVRAM to the high-availability (HA) and/or cluster partner.
- **-total-.** The average of average latencies on the cluster.

```
ontaptme-fc-cluster::*> qos statistics volume resource cpu show -node ontaptme-fc-cluster-03
Workload           ID   CPU
--------------- ----- -----
-total- (100%)      -    5%
bobvol1-wid13.. 13891    3%
bobvol5-wid9864  9864    1%
```

Some workloads are more expensive than others in terms of CPU utilization due to application-specific behavior. The preceding volume resource cpu command displays the specific CPU utilization for a given volume workload. Note that this in no way represents the total physical node-level CPU utilization. That is a different topic of discussion because there are many internal Data ONTAP processes that can be running not accounted for here. It should also be noted that indirect workloads are present here to account for the transport protocol CPU overhead (see bobvol5-wid9864 earlier):

- **Workload.** Concatenation of volume name and internal workload identifier.
- **ID.** Internal workload identifier.
- **CPU.** Processor resource utilization attributed to the workload.
- **-total-.** Sum of all the workload CPU utilization metrics.

```
ontaptme-fc-cluster::*> qos statistics volume resource disk show -node ontaptme-fc-cluster-03
Workload            ID   Disk Number of HDD Disks
--------------- ------ ------ ------------------
-total-              -    15%                 12
bobvol5-wid9864   9864    26%                  7
```

In a similar fashion to CPU, physical disks are a limited shared resource where some workloads are more expensive than others. That is where the similarities end, though. In the preceding command context, disk utilization represents the amount of time a disk is busy servicing requests on behalf of the volume workload. This is indeed a major contributing factor to disk component latency described in section 2.4.1, "Cluster Operations." Disk utilization (and thus disk latency) can widely vary among workloads due to many factors previously discussed such as data access patterns, working set size, free space, and cache

resources. Unlike the volume resource cpu command, the volume resource disk command only includes workloads that are local to the node because it is fundamentally impossible for disk (or aggregate) utilization to be split across multiple nodes:

- **Workload.** Concatenation of volume name and internal workload identifier.
- **ID.** Internal workload identifier.
- **Disk.** Average disk utilization attributed to the workload.
- **-total-.** Average disk utilization of all disks on node.

2.6.3 Managing Workloads with Data Placement

QoS is a very valuable tool to manage workloads within the cluster; however, the location and access path of data in the cluster can also play a role in performance, as was mentioned earlier. Clustered Data ONTAP has features that allow data to be moved, cached, and duplicated across nodes in the cluster to help manage performance.

DataMotion for Volumes

Independent of protocol, volumes can be moved and mirrored in the storage layer. Using volume move (vol move), volumes can be moved to the node handling the most client access to increase direct access. Using the same method, volumes can be moved to different disk types or nodes with different hardware to achieve different performance characteristics. Volume moves should be used to proactively manage performance and not when encountering performance problems, because volume move requires resources to perform the movement.

3 Performance Management with OnCommand Performance Manager

OnCommand Performance Manager (OPM), an integrated component of OnCommand Unified Manager, is designed to monitor cluster performance by regularly collecting cluster statistics, analyzing volume workload performance, establishing baseline metrics, and automatically calculating critical thresholds. When volume workload measurements exceed threshold limits, OPM will generate a performance incident, identify the shared resource under contention, and identify the workload consuming an unfair share of the global resource.

3.1 OnCommand Performance Manager Scenario Overview

This section walks through an OPM-detected performance incident using a scenario created in a lab environment. In this scenario, there will be three workloads under consideration:

- **Victim workload.** The volume workload targeted for performance degradation. The victim workload experiences increased response times (or latency) that exceed the OPM-established thresholds.
- **Bystander workload.** This volume workload is constantly present throughout the scenario. However, the bystander does not share the resource under contention and remains unaffected.
- **Bully/shark workload.** The volume workload introduced that intentionally consumes a large share of a shared cluster resource. In this case the bully workload will over-utilize disk I/O capacity, putting the aggregate component under contention.

OPM references cluster storage objects by name. To make the scenario easier to follow, the following table can be used as a guide to better understand the cluster configuration.

Table 5) Cluster configuration key for OPM incident scenario.

Storage Object Name	Role
ontaptme-fc-cluster	Cluster name
bobvol1	Victim volume workload
bobvol5	Bystander volume workload
bobvol2	Bully volume workload
aggr3	Aggregate used by bobvol1 and bobvol2

At the start of this scenario the victim volume workload and the bystander volume workload offer load at their steady-state rates. Then the bully workload is introduced, causing performance degradations to the extent that OPM generates an incident.

Thus the sequence of events in this scenario is as follows:

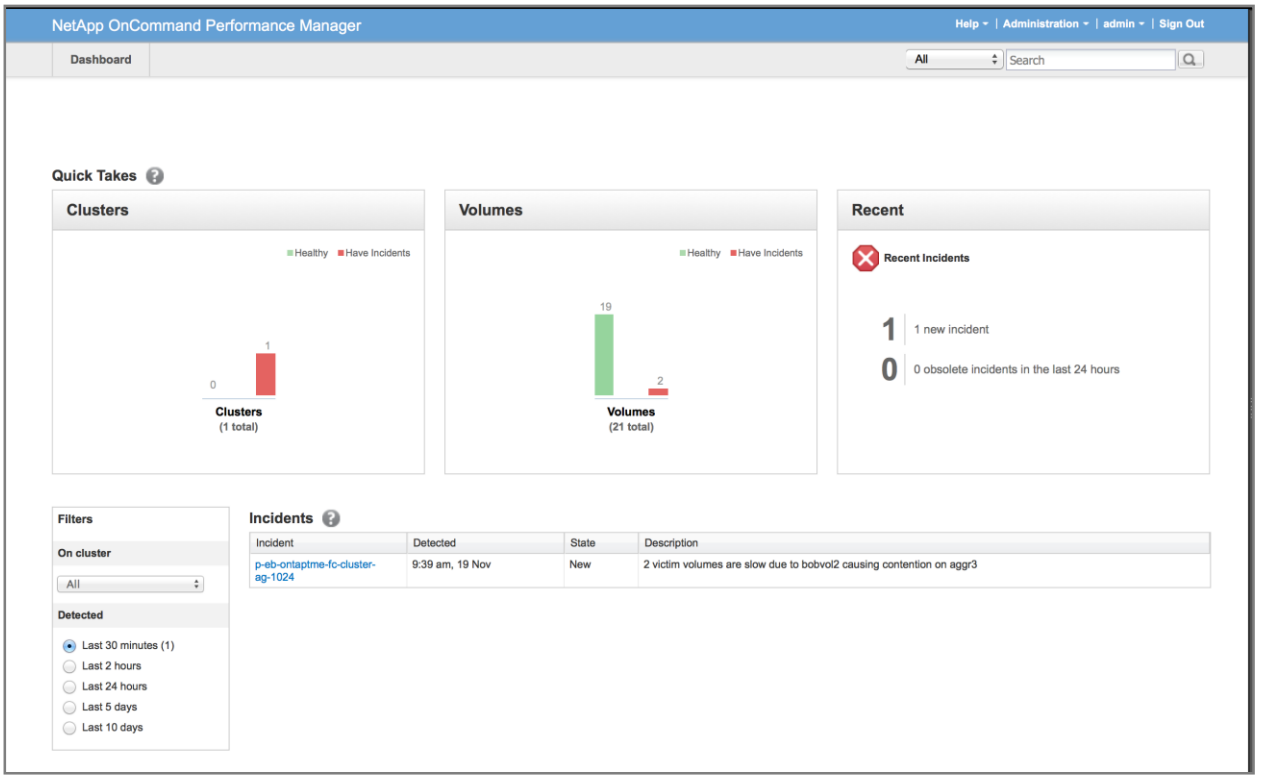
1. Start victim workload.
2. Start bystander workload.
3. Allow baselines and alert thresholds to establish.
4. Start bully workload.
5. Wait for incident to be detected.
6. View OPM incident details and analysis.

3.1.1 OnCommand Performance Manager Incident Detection

The first unsolicited indication OPM provides when a performance incident occurs is by e-mail if configured to do so. (Note: Since OPM is integrated with OnCommand Unified Manager, notifications can be sent to OnCommand Unified Manager as well.) In the e-mail body, links to the incident details view are provided.

In addition to an e-mail alert, an incident indication appears in the main dashboard view as depicted in Figure 15) OPM dashboard.

Figure 15) OPM dashboard.



In the dashboard view there are several panels listing a count of the clusters and volumes involved in recent incidents. The most significant of these panels is the panel labeled “Incidents” (see Figure 16).

Figure 16) OPM dashboard incident panel.

Incidents

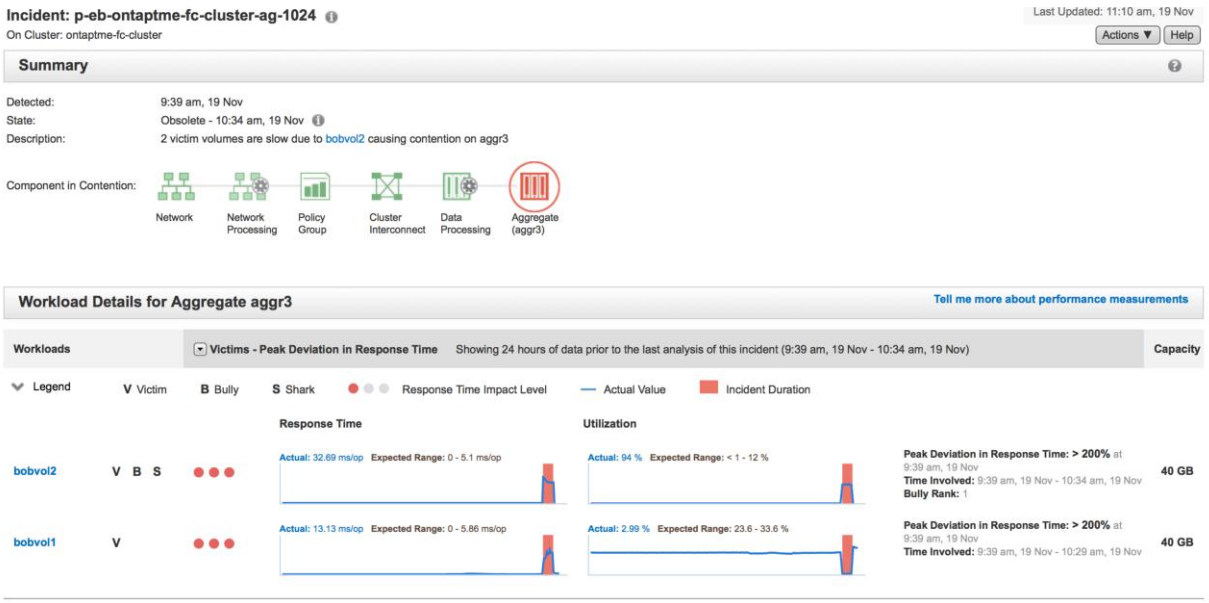
Incident	Detected	State	Description
p-eb-ontaptme-fc-cluster-ag-1024	9:39 am, 19 Nov	New	2 victim volumes are slow due to bobvol2 causing contention on aggr3

The dashboard incident panel shows an incident was detected at 9:39 a.m. It shows that two volumes are involved, victim and bully, and identifies the cluster resource name under contention. The incident identifier in the first column provides a link to the incident details view where additional information is provided.

3.1.2 OnCommand Performance Manager Incident Details

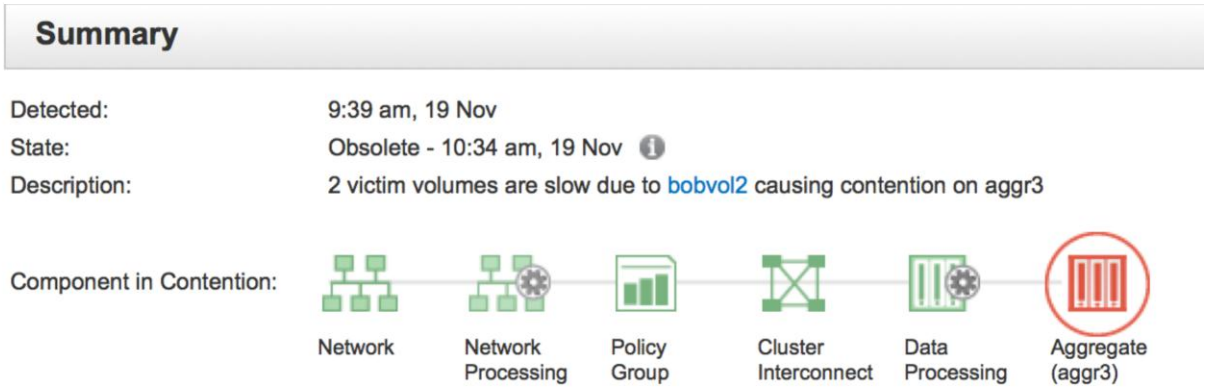
The incident details view is where OPM shows the volume workloads and cluster resources involved in the incident. In the incident details view, OPM provides a "Summary" panel with a graphical depiction of the cluster resource components where the resource under contention is highlighted in red. A second "Workload Details" panel is present with graphs plotting workload latency and workload resource utilization (see Figure 17).

Figure 17) OPM incident details.



In the incident detail "Summary" panel the cluster component resource under contention is highlighted in red (see Figure 18). Here, OPM shows the aggregate component named "aggr3" is under contention and that the incident ended at 10:34 a.m. This indicates that incident took place for approximately 55 minutes.

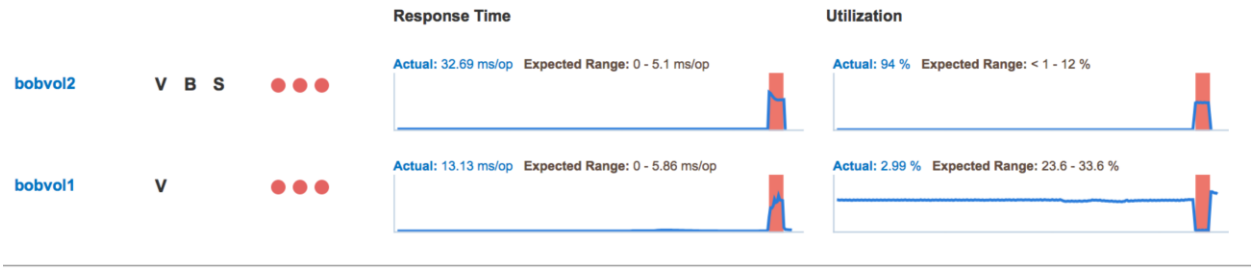
Figure 18) OPM incident details summary cluster components.



Below the "Summary" panel appears the "Workload Details" section (see Figure 19), again identifying the aggregate resource under contention in the section title. Below this appear two line items representing workloads involved in the incident. The first workload listed is "bobvol2" and is identified as victim, bully,

and shark. In this particular incident, the bully workload also exceeded its established threshold and is labeled a victim. The second workload listed, “bobvol1," is simply a victim. From this view it is clear to see where bully and victim workloads both experience increased latencies and where the bully workload, bobvol2, increases resource utilization from 1% to 94%, at the expense of the victim workload, whose resource utilization decreases from 23.6% to 2.99%.

Figure 19) OPM incident details workload details.

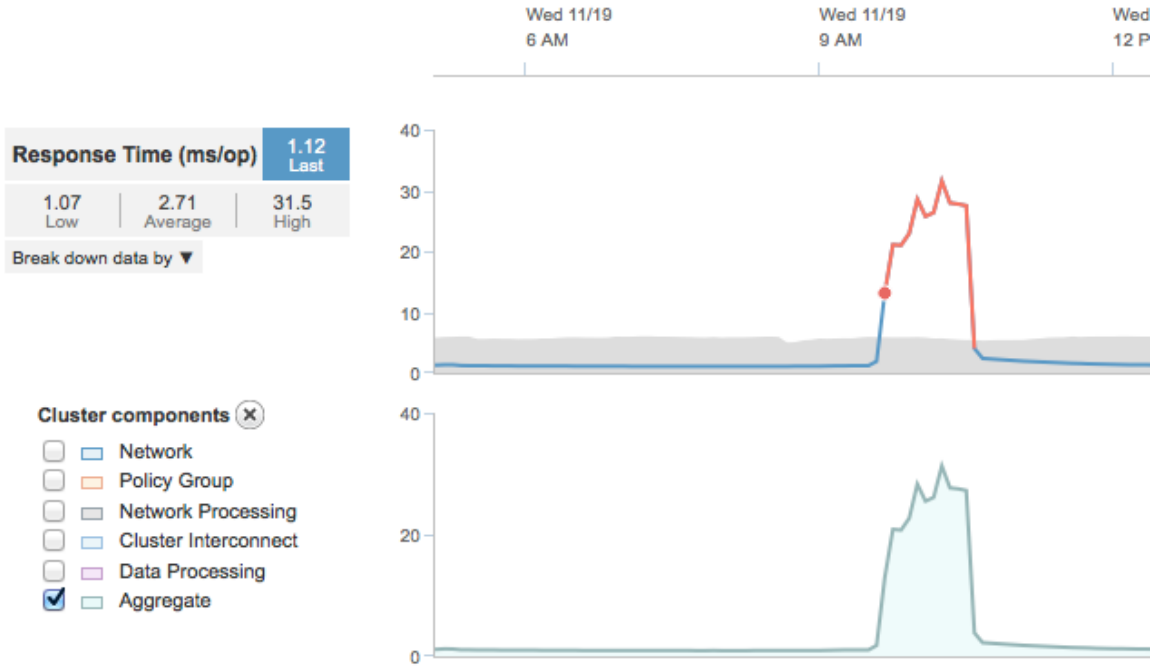


OPM also offers remediation recommendations (not shown). Among the recommendations are suggestions to relocate the volume (see section on Managing Workloads with Data Placement) or cap the workload using a policy group limit (see section on Examples of Using Storage QoS).

3.1.3 OnCommand Performance Manager Victim Volume Workload Monitoring

Given the analytics from OPM, additional information on latency contribution is rarely needed. However, OPM provides views of latency broken down by component and operation type (see Figure 20). By selecting the component of interest and visually comparing the shape of the curves, it can be seen that read latency increases perfectly track aggregate latency increases, which correlates directly to overall response time. Latency contribution of any component can be added to or removed from view by selecting the checkboxes under “Cluster Components” shown to the left in Figure 20.

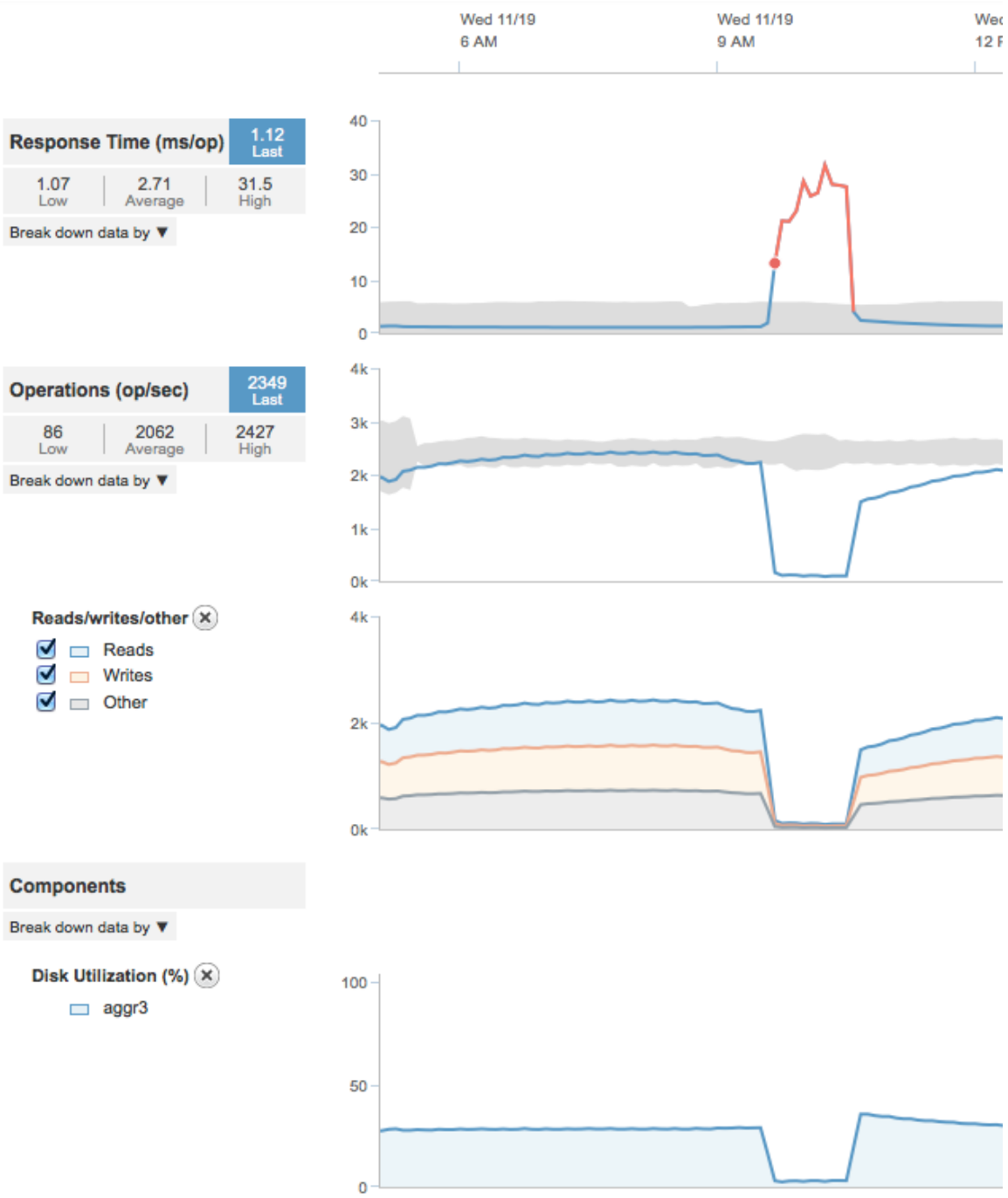
Figure 20) OPM victim volume response time correlated with aggregate response time.



More details about the victim workload incident can be seen in Figure 21. From the information presented here, the correlation between decreased disk utilization and increased response times clearly affected all operation types for this workload. This is perfectly consistent with the incident details presented earlier in Figure 19.

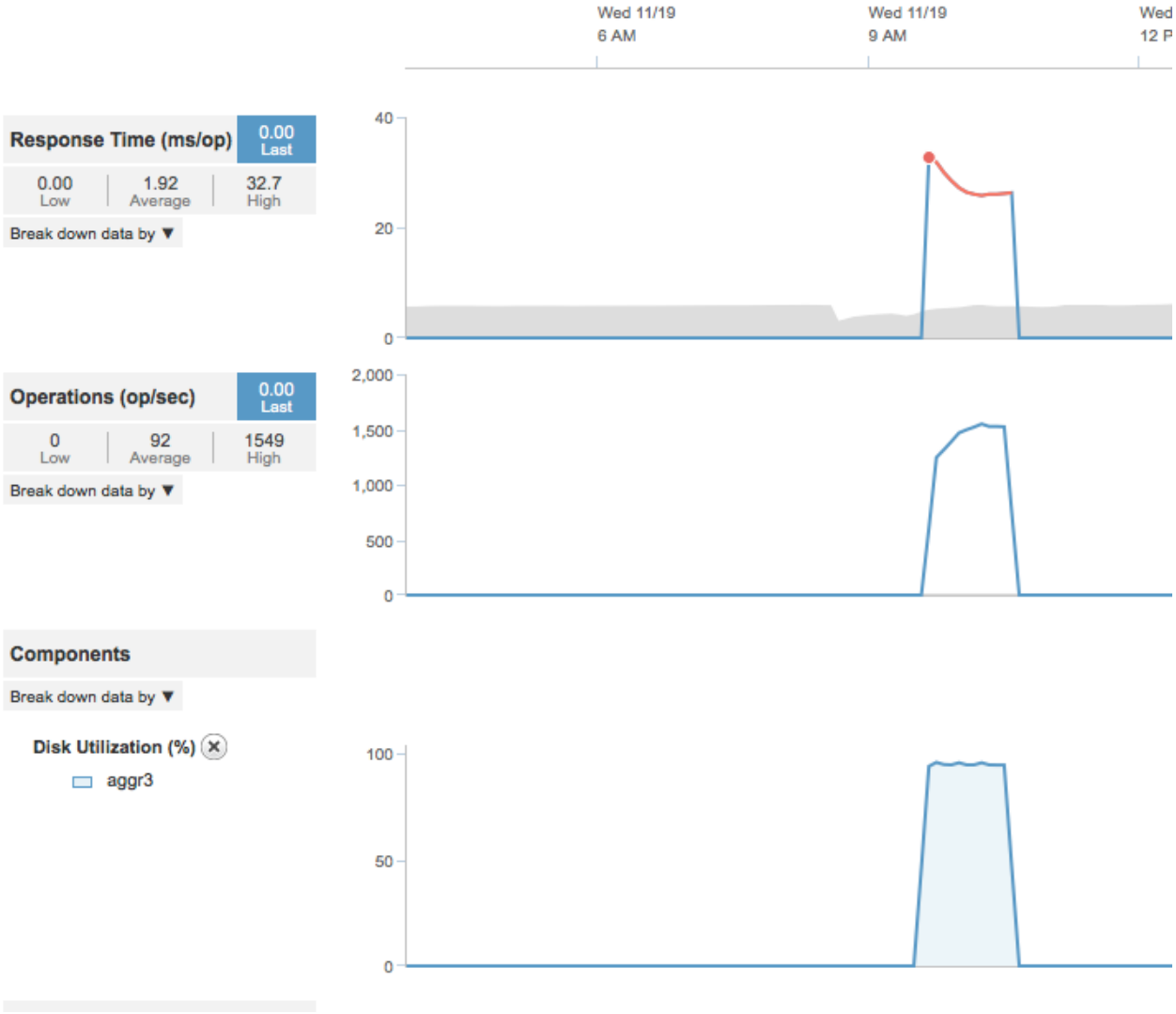
3.1.4 OnCommand Performance Manager Bully Volume Workload

Figure 21) OPM victim volume response time correlated with op rate, op type, and disk.



The same details can be seen on the bully workload, though in this case it is more interesting to see that disk utilization increases directly in proportion to operation rate (see Figure 22).

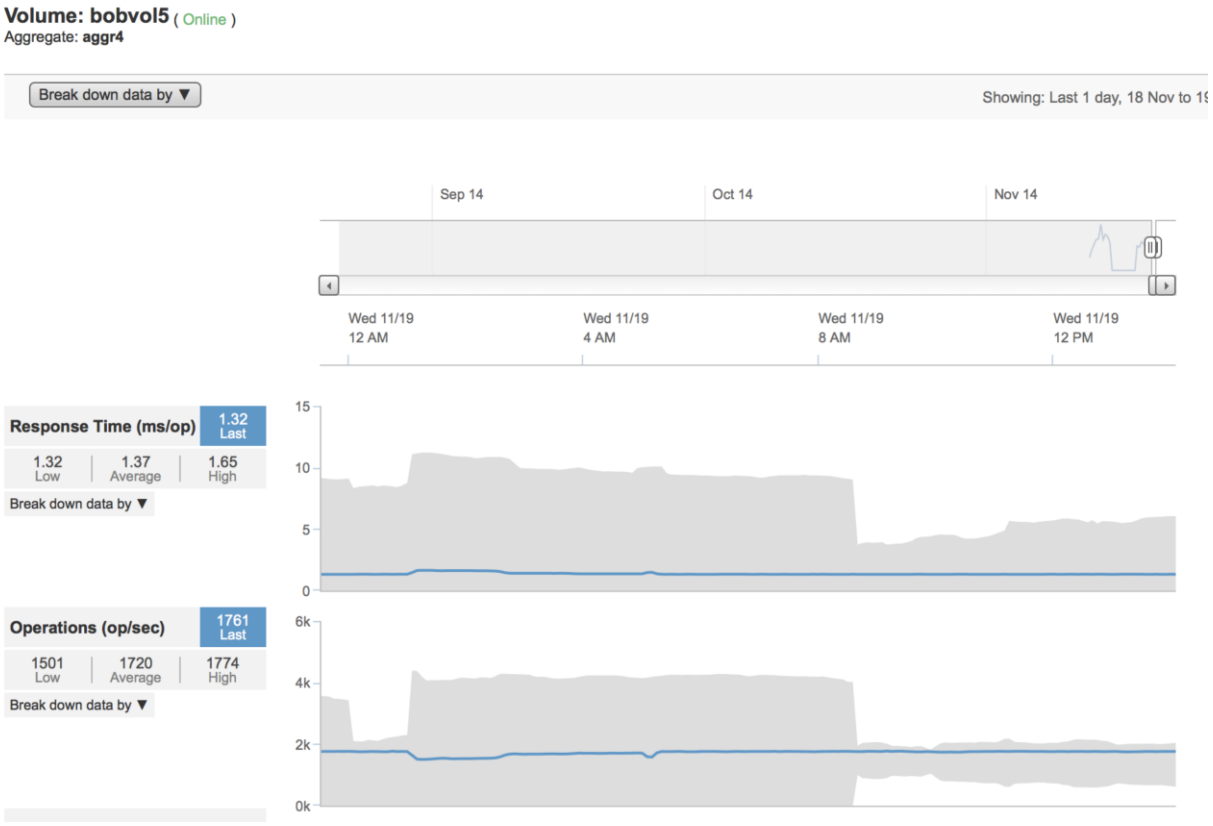
Figure 22) OPM bully workload op rate correlation to disk utilization.



3.1.5 OnCommand Performance Manager Bystander Volume Workload

During the performance incident duration, from 9:39 a.m. to 10:34 a.m., the bystander workload does not experience any performance impact. This is primarily because it is not sharing the contended resource ("aggr3"). Note that other resources on the cluster are shared, such as CPU; however, those are not in contention. Thus, performance is not affected and remains consistent and within the established thresholds. This can be confirmed through browsing the bystander volume workload as shown in Figure 23.

Figure 23) OPM bystander volume workload performance graph.



3.1.6 OnCommand Performance Manager 1.1 Interoperability

OPM supports monitoring clustered Data ONTAP systems. OPM does not support legacy 7-Mode Data ONTAP systems. For the most current interoperability information, see section 3.1.7.

OPM 1.1 can be deployed in two environments where there are browser, system, and hardware requirements. For the most current information, see one of the following documents:

- Installed as a VMware® virtual appliance (vAPP), see OnCommand Performance Manager 1.1 Installation and Administration Guide for VMware Virtual Appliances.
- Installed as a Red Hat Enterprise Linux® application, see OnCommand Performance Manager 1.1 Installation and Setup Guide for Red Hat Enterprise Linux.

3.1.7 OnCommand Performance Manager 1.1 Interoperability Matrix Tool (IMT)

See 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.

Additional Resources

- TR-4015: SnapMirror Configuration and Best Practices Guide for Clustered Data ONTAP 8.2
  http://www.netapp.com/us/media/tr-4015.pdf
- TR-4063: Parallel NFS File System Configuration and Best Practices for Data ONTAP Cluster-Mode
  http://www.netapp.com/us/media/tr-4063.pdf
- TR-4067: Clustered Data ONTAP NFS Implementation Guide
  http://www.netapp.com/us/media/tr-4067.pdf
- TR-3982: NetApp Clustered Data ONTAP 8.3: An Introduction
  http://www.netapp.com/us/media/tr-3982.pdf
- TR-4080: Best Practices for Scalable SAN in Clustered Data ONTAP 8.3
  http://www.netapp.com/us/media/tr-4080.pdf
- TR-3832: Flash Cache Best Practices Guide
  http://www.netapp.com/us/media/tr-3832.pdf
- TR-4070: Flash Pool Design and Implementation Guide
  http://www.netapp.com/us/media/tr-4070.pdf
- TR-3838: Storage Subsystem Configuration Guide
  http://www.netapp.com/us/media/tr-3838.pdf
- OnCommand Performance Manager 1.1 Release Notes
  https://library.netapp.com/ecm/ecm_download_file/ECMP1552963
- OnCommand Performance Manager 1.1 Installation and Administration Guide for VMware Virtual Appliance
  https://library.netapp.com/ecm/ecm_download_file/ECMP1609318
- OnCommand Performance Manager 1.1 Installation and Setup Guide for Red Hat Enterprise Linux
  https://library.netapp.com/ecm/ecm_download_file/ECMP1609318
- OnCommand Performance Manager 1.1 User Guide
  https://library.netapp.com/ecm/ecm_download_file/ECMP1552961

Contact Us

Let us know how we can improve this technical report.

Contact us at docfeedback@netapp.com.

Include TECHNICAL REPORT 4211 in the subject line.

Addendum

8.3 Clustered Data ONTAP Upgrade Recommendations

There no known issues regarding performance when upgrading from 8.2 clustered Data ONTAP to 8.3 at the time of this writing. In all cases the recommendation is to contact your NetApp sales representative and/or utilize the NetApp Upgrade Advisor on the NetApp Support site:

http://support.netapp.com/NOW/asuphome/

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.