NETAPP TECHNICAL REPORT

Managing Performance Advisor Data

Harish Kumar, NetApp
November 2009 | TR-3751

ABSTRACT

Performance Advisor collects a large number of useful performance metrics from appliances. Using it, you can monitor the performance of appliances as well as analyze performance problems, if any. Although there is a trade-off with Performance Advisor between the information collected and the space required to store the information, this document describes how to optimize the storage space used to store performance information.

TABLE OF CONTENTS

1 INTRODUCTION

Performance Advisor collects performance information from appliances and organizes them in groups of performance metrics called performance counter groups. Each counter group has two important properties: the sampling interval and the maximum number of records. Individual records are collected from the appliance at regular intervals, according to its sampling interval. Once the number of records reaches the maximum number of records, the oldest records are overwritten to store the new records.

This document provides the following information:

- Describes how much space is used over a period of time and gives an example for a typical setup and storage controller configuration
- Provides an equation to compute the space required to store performance data for a typical setup and storage controller configuration
- Provides an equation to compute the daily changes to help in backup planning for a typical setup and storage controller configuration
- Provides ways to reduce the collection of data
- Describes other controls related to Performance Advisor data
- Describes default counter groups, their sampling interval, maximum records, and the counters whose data is collected

2 TYPICAL PERFORMANCE DATA USAGE

Storage space used by Performance Advisor to store performance data retrieved from appliances can be confusing to users because the growth is not linear. Instead, the space occupied increases in distinct phases. To explain this further, this section describes a typical setup of an appliance and the manner in which Performance Advisor uses space to store performance data collected for this appliance over a period that spans for more than a year.

2.1 SETUP DETAILS

The following table describes the setup of the appliance used.

Data ONTAP® version	7.2.3
Number of vFiler™ units	2
Number of aggregates	4
Number of volumes	33
Number of qtrees	0
Number of disks	51
Number of LUNs	31
Number of CPUs	2
Number of network interfaces	2
Number of FCP targets	1

2.2 SPACE INCREASE CHART

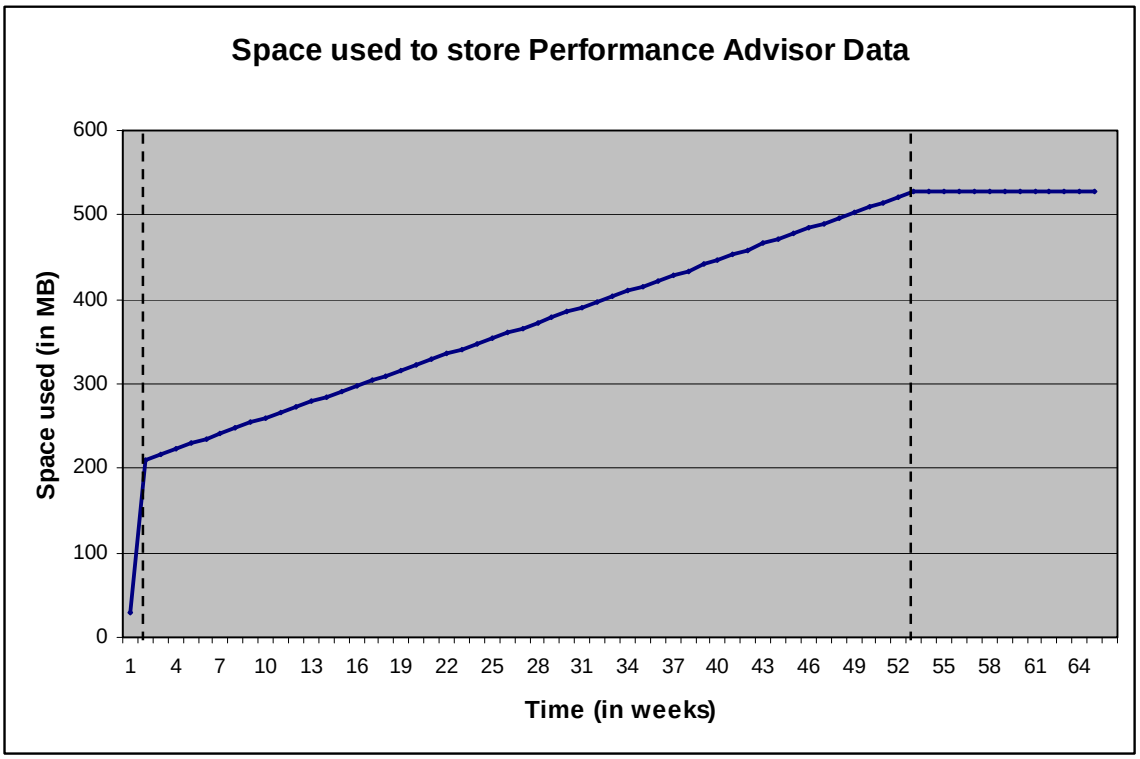

Figure 1) Space used to store performance data of an appliance over time.

Figure 1 shows the amount of space used to store performance data collected from the appliance. As seen from the chart, there are three phases of space growth, each of which is explained in more detail below.

PHASE 1 (STARTUP)

During the first week, performance data is collected for each of the counter groups. The space used for storing the samples collected for all counter groups during this period is not recycled; hence, the space occupied increases rapidly throughout the week. In the above example, the space used during the first week is a little more than 200MB. This value is 7 times the value given by the equation in section 3.2.

PHASE 2 (LINEAR)

During this phase, space usage increases much more slowly compared to the first week. In this phase, space is used to store new data only for those counter groups whose collection period is more than a week (example: System Archive). For other counter groups whose retention period is one week (example: Volume Basic), the space used for storing the data during the first week is recycled, requiring no additional space. In the above example, the space used increases from 200MB to 520MB in this phase. For details about the retention period of counter groups, see the appendix.

PHASE 3 (CAPPED)

During this phase, space usage remains constant. The space used currently will be recycled to store data being collected from the appliance. In the above example, the space used stays at a constant 528MB.

ASSUMPTIONS IN THE ABOVE DESCRIPTION

No new objects are created.

None of the existing objects are deleted.

No default counter groups except “Qtree Basic” are disabled.

Retention period and sampling intervals of the default counter groups are left unchanged.

No custom views are created.

3 PERFORMANCE ADVISOR SPACING TOOLS

3.1 PERFORMANCE ADVISOR SPACE CALCULATOR

Before installing Operations Manager, the user can determine the space Performance Advisor will consume to store the performance metrics collected from the appliances. The same thing applies when one or more new appliances are added to Operations Manager installation. The following equation gives a formula to compute the approximate space that Performance Advisor will use for storing performance data over a period of one year (after a year, the space used will remain constant):

5060160 + 339108480 * no_filers + 161280 * no_cpus + 241920 * no_fcp_targets + 564480 * no_aggrs + 645120 * no_luns + 967680 * no_nw_ifs + 967680 * no_vfilers + 1451520 * no_disks + 3314304 * no_vols

In the above equation, 5060160 is the space required to store the timestamps. Each of the objects has a multiplier, which is proportional to the number of counters being collected for that object as well as the retention period. As an example, applying this formula to the above setup, the result is 528.5MB.

In an existing installation of Operations Manager, the number of objects can be obtained from the `dfm diag` command. The assumptions mentioned in section 2.2 also apply to this equation.

3.2 PERFORMANCE ADVISOR DATA CHANGE CALCULATOR

Similar to the requirement in section 3.1, users also need to know the number of changes to the performance data every day when planning to back up performance data elsewhere. Based on the changes to the data, the size of the secondary storage can be calculated. For this, the following equation gives a formula to compute the approximate daily changes:

104256 + 1369728 * no_filers + 23040 * no_cpus + 34560 * no_fcp_targets + 80640 * no_aggrs + 92160 * no_luns + 138240 * no_nw_ifs + 138240 * no_vfilers + 207360 * no_disks + 473472 * no_vols

As an example, applying this formula to the above setup, the result is around 30MB.

Backups based on Snapshot™ should be considered if the backup times are very long or if disaster recovery is planned. The assumptions mentioned in section 2.2 also apply to this equation.

4 MECHANISMS TO REDUCE PERFORMANCE DATA COLLECTION

This section describes those controls that can be applied either at the appliance level or at the counter group level.

4.1 DISABLING PERFORMANCE ADVISOR DATA COLLECTION OF A GIVEN APPLIANCE

Performance Advisor data collection can be turned off for selected appliances. These could be those appliances for which collection of performance data is not mandatory, say, for appliances in test environments. To disable performance data collection for a given appliance, the appliance option `perfAdvisorTransport` should be set to `disabled`.

EXAMPLE

To disable performance data collection of appliance `toaster`, the command would be

```
dfm host set toaster perfAdvisorTransport=disabled.
```

NOTES

The above command stops subsequent performance data collection for the appliance. Performance data already collected will not be purged.

The above command can also be used to stop further performance data collection of all appliances in a resource group by specifying the group instead of the appliance.

4.2 DISABLING DATA COLLECTION FOR A COUNTER GROUP ACROSS ALL APPLIANCES

Performance Advisor collects a variety of information by default. An example is data related to WAFL®. If the user does not want some performance data to be collected, data collection for that counter group can be disabled for all appliances using the `perfAdvisorDisabledCounterGroups` global option. The counter group names are available in the appendix.

EXAMPLE

To disable data collection of the `"Wafl basic"` counter group across all appliances, the command would be

```
dfm option set perfAdvisorDisabledCounterGroups="Wafl basic".
```

NOTES

If data collection for more than one counter group needs to be disabled, all those counter groups should be specified as a single string, separated by commas.

The above command stops subsequent performance data collection for the specified counter groups. Performance data already collected for these counter groups will not be purged.

The above command requires that the DataFabric® Manager server service be restarted.

4.3 DISABLING DATA COLLECTION FOR ONE OR MORE COUNTER GROUPS FOR A GIVEN APPLIANCE

If the user does not want some performance data to be collected for a given appliance, the data collection for the related counter groups can be disabled using the `perfAdvisorDisabledCounterGroups` host option.

EXAMPLE

To disable data collection of FCP- and iSCSI-related metrics of an appliance named `toaster`, the command would be

```
dfm host set toaster perfAdvisorDisabledCounterGroups="FCP Basic,iSCSI Basic".
```

NOTES

If data collection for more than one counter group needs to be disabled, all those counter groups should be specified as a single string, separated by commas.

The above command stops subsequent performance data collection for the specified counter groups. Performance data already collected for these counter groups will not be purged.

4.4 REDUCING THE RETENTION PERIOD OF A COUNTER GROUP ACROSS ALL APPLIANCES

The space used to store Performance Advisor data can be reduced by decreasing the retention period using the `dfm perf data modify` command. This command changes the maximum records stored for each counter group. Once the maximum record limit is reached, the oldest records will be overwritten to store new records.

EXAMPLE

To reduce the maximum number of records stored by the `System Archive` counter group from one year to one month, the command would be

```
dfm perf data modify –f "System Archive" collectionPeriod="30 days".
```

4.5 REDUCING THE SAMPLING INTERVAL OF A COUNTER GROUP ACROSS ALL APPLIANCES

The frequency at which performance data is collected from appliances can be modified. By lengthening the sampling interval and retaining the same retention period, the maximum number of records stored can be reduced.

EXAMPLE

To collect data stored in the `System Archive` counter group every five minutes instead of every minute, the command would be

```
dfm perf data modify –f "System Archive" sampleRate="5 minutes".
```

NOTES

The above command changes the sampling interval for all appliances for the specified counter group. It is not possible to do it for selected appliances.

5 OTHER CONTROLS FOR MANAGING STORAGE

5.1 CHANGING THE LOCATION OF PERFORMANCE DATA

By default, performance data is stored under the `<dfm-installation-directory>/perfdata` directory. For example, if Operations Manager is installed under the `/opt/NTAPdfm` directory, performance data will be stored in the `/opt/NTAPdfm/perfdata` directory. This directory can be modified by setting the `perfArchiveDir` option to the new directory.

EXAMPLE

To set the `perfArchiveDir to /dfm_perf_data` directory, the command is

```
dfm option set perfArchiveDir="/dfm_perf_data".
```

This option can be set to a path that points to NetApp® storage (either using an NFS export/CIFS share or a LUN). With this configuration, Operations Manager can be used to monitor the space of the volume on which the performance data resides. Thresholds can also be set on this volume to alert the user when the space usage reaches a predefined value. These alerts can also be used to execute scripts configured by the user. Even without thresholds, the “Volume Space Breakout” graph gives useful information about the volume, such as the “daily growth rate” and “days to fill.”

If the performance data and the database information of Operations Manager are stored on a LUN created on a NetApp appliance, database backups of this Operations Manager installation that include the performance data can be created using Snapshot copies of this LUN.

NOTES

The directory should exist before running this command.

This command requires the DataFabric Manager server service to be restarted.

If there is existing performance data, move it to the new folder before starting the DataFabric Manager server service.

5.2 GETTING DETAILS OF SPACE BY COUNTER GROUPS

The `"dfm perf data list"` CLI shows the various counter groups and their details. The details provided are as follows:

Names of all the default counter groups

The sampling interval for each of the default counter groups

The period for which the data will be collected (which is the multiple of sampling interval and maximum records)

The maximum number of records that will be stored for each counter group; once this number is reached, the oldest records will be overwritten to store new records

The space occupied by each of the default counter groups across all appliances

To get even more granular details, the "`dfm perf data describe`" CLI can be used. This gives the details of a given counter group for a given appliance.

For example, to get details of the `System Archive` counter group for appliance `toaster`, the command would be

```
dfm perf data describe "System Archive" toaster.
```

APPENDIX COUNTER GROUP DETAILS

Counter Group Name	Sampling Interval (in Seconds)	Collection Period (in Days)	Counter Names
Aggregate Basic	60	7	aggregate:cp_read_blocks
			aggregate:cp_reads
			aggregate:total_transfers
			aggregate:user_read_blocks
			aggregate:user_reads
			aggregate:user_write_blocks
			aggregate:user_writes
CIFS Basic	60	7	cifs:cifs_ops
			cifs:cifs_latency
CPU Archive	60	365	processor:domain_busy
			processor:processor_busy
			cpu:cpu_real
CPU Basic	60	7	processor:processor_busy
Disk Basic	60	7	disk:cp_reads
			disk:user_read_blocks
			disk:user_write_blocks
			disk:user_read_latency
			disk:user_write_latency
			disk:cp_read_latency
			disk:user_read_chain
			disk:user_write_chain
			disk:cp_read_chain
			disk:user_writes
			disk:user_reads
			disk:total_transfers
			DISK:guaranteed_writes
			DISK:guaranteed_reads
			disk:disk_busy
			disk:cp_read_blocks
FCP Basic	60	7	fcp:fcp_read_data
			fcp:fcp_write_data
			fcp:fcp_ops
iSCSI Basic	60	7	iscsi:iscsi_ops
			iscsi:iscsi_write_data

			iscsi:iscsi_read_data
LUN Basic	60	7	lun:read_data
			lun:avg_latency
			lun:write_data
			lun:total_ops
			lun:read_ops
			lun:other_ops
			lun:write_ops
Network Interface Basic	60	7	ifnet:send_mcasts
			ifnet:send_packets
			ifnet:send_errors
			ifnet:total_mcasts
			ifnet:recv_mcasts
			ifnet:total_errors
			ifnet:total_packets
			ifnet:total_data
			ifnet:recv_data
			ifnet:recv_errors
			ifnet:recv_packets
			ifnet:send_data
NFS Basic	60	7	nfsv3:nfsv3_write_latency
			nfsv3:nfsv3_write_ops
			nfsv3:nfsv3_read_ops
			nfsv3:nfsv3_ops
			nfsv3:nfsv3_read_latency
Perf Archive	600	365	perf:data_disk_util_percnt_histo
			perf:ops_per_sec_histo
			perf:parity_disk_util_percnt_histo
QOS Basic	60	7	priorityqueue:sys_weight
			prisched:wake_on_mesg_rate
			priorityqueue:nvlog_limit_full
			priorityqueue:nvlog_used_max
			priorityqueue:nvlog_limit
			priorityqueue:sys_reads
			priorityqueue:sys_read_limit
			priorityqueue:max_user_reads
			volume:pq_sys_weight
			volume:pq_sys_sched_total
			volume:pq_max_user_reads
			volume:pq_nvlog_used_max
			priorityqueue:usr_sched_total
			priorityqueue:usr_read_limit
			priorityqueue:sys_sched_total
			system:pq_nvlog_limit_full
			prisched:wake_on_sig_rate
			priorityqueue:usr_weight
			prisched:queued_max
			prisched:bypass_rate

			prisched:queued
			volume:pq_max_sys_reads
			prisched:preempt_rate
			prisched:delayed_messages
			prisched:delayed_io_blocked
			prisched:schedslow
			system:pq_sys_weight
			system:pq_usr_sched_total
			system:pq_sys_sched_total
			system:pq_usr_read_limit
			system:pq_max_user_reads
			system:pq_usr_weight
			system:pq_nvlog_limit
			system:pq_nvlog_used_max
			system:pq_sys_read_limit
			volume:pq_usr_weight
			volume:pq_usr_sched_total
			volume:pq_usr_read_limit
			volume:pq_sys_read_limit
			volume:pq_nvlog_limit
			volume:pq_nvlog_limit_full
			system:pq_max_sys_reads
Qtree Basic	60	7	qtree:total_ops
			Qtree:internal_ops
			Qtree:cifs_ops
			Qtree:nfs_ops
System Archive	60	365	system:avg_processor_busy
			system:load_write_bytes_ratio
			system:load_total_mbps
			iscsi:iscsi_ops
			system:load_inbound_mbps
			iscsi:iscsi_write_ops
			system:total_ops
			system:nas_throughput
			system:disk_throughput
			fcp:fcp_read_latency
			iscsi:iscsi_read_ops
			fcp:fcp_write_latency
			fcp:fcp_throughput
			iscsi:iscsi_read_latency
			iscsi:iscsi_write_latency
			iscsi:iscsi_throughput
			nfsv4:nfsv4_avg_latency
			nfsv3:nfsv3_ops
			nfsv4:nfsv4_read_ops
			nfsv4:nfsv4_write_ops
			cifs:cifs_ops
			fcp:fcp_read_ops

			system:load_outbound_mbps
			fcp:fcp_write_ops
			nfsv3:nfsv3_avg_op_latency
			fcp:fcp_ops
			system:nfs_ops
			system:net_data_sent
			system:net_data_recv
			system:nc_urls
			system:iscsi_ops
			system:http_ops
			system:fcp_ops
			system:disk_data_written
			system:disk_data_read
			system:cpu_busy
			system:cifs_ops
			nfsv3:nfsv3_write_ops
			nfsv3:nfsv3_write_latency
			system:dafs_ops
			nfsv3:nfsv3_read_latency
			nfsv4:nfsv4_ops
			system:disk_data_written_mbps
			nfsv3:nfsv3_read_ops
			cifs:cifs_latency_base
			system:disk_data_read_mbps
			system:total_processor_busy
			cifs:cifs_latency
			fcp:fcp_read_data
			fcp:fcp_write_data
			iscsi:iscsi_read_data
			iscsi:iscsi_write_data
			system:load_read_bytes_ratio
System Basic	60	7	system:net_data_recv
			system:cifs_ops
			system:dafs_ops
			system:disk_data_read
			system:disk_data_written
			system:fcp_ops
			system:http_ops
			system:iscsi_ops
			system:net_data_sent
			system:cpu_busy
			system:nfs_ops
Target Basic	60	7	target:write_ops
			target:read_ops
			target:other_ops
vFiler Basic	60	7	vfiler:vfiler_read_bytes
			vfiler:vfiler_write_ops
			vfiler:vfiler_write_bytes

			vfiler:vfiler_read_ops
			vfiler:vfiler_net_data_sent
			vfiler:vfiler_total_ops
			vfiler:vfiler_nw_throughput
			vfiler:vfiler_data_transferred
			vfiler:vfiler_net_data_recv
			vfiler:vfiler_misc_ops
			vfiler:vfiler_cpu_busy_base
			vfiler:vfiler_cpu_busy
Volume Basic	60	7	volume:cifs_read_latency
			volume:nfs_write_latency
			volume:san_write_latency
			volume:san_other_ops
			volume:cifs_write_latency
			volume:cifs_other_latency
			volume:other_latency
			volume:san_other_latency
			volume:flexcache_read_ops
			volume:san_read_ops
			volume:flexcache_other_ops
			volume:read_ops
			volume:total_ops
			volume:flexcache_write_ops
			volume:write_data
			volume:san_read_latency
			volume:cifs_write_ops
			volume:nfs_read_latency
			volume:nfs_read_ops
			volume:sys_ops
			volume:nfs_other_latency
			volume:san_write_ops
			volume:cifs_other_ops
			volume:cifs_read_ops
			volume:nfs_write_ops
			volume:throughput
			volume:user_ops
			volume:read_latency
			volume:read_data
			volume:other_ops
			volume:write_ops
			volume:avg_latency
			volume:nfs_other_ops
			volume:write_latency
WAFL Basic	60	7	wafl:wafl_cp_total
			wafl:wafl_cp_blocked
			wafl:wafl_cp_normal
			wafl:cp_count