



# **ONTAP 9.16.1 performance counter mapping**

ONTAP 9.16.1 performance counter mapping

NetApp  
November 15, 2024

# Table of Contents

- ONTAP 9.16.1 performance counter mapping . . . . . 1
- Overview . . . . . 2
- copy\_manager . . . . . 4
  - Table Row IDs . . . . . 4
  - Properties . . . . . 4
  - Counters . . . . . 5
  - Property/Counter Content Changes . . . . . 9
  - Table Aliases . . . . . 10
- disk . . . . . 11
  - Table Row IDs . . . . . 11
  - Properties . . . . . 11
  - Counters . . . . . 12
  - Property/Counter Content Changes . . . . . 19
  - Table Aliases . . . . . 19
- external\_cache . . . . . 20
  - Table Row IDs . . . . . 20
  - Properties . . . . . 20
  - Counters . . . . . 20
  - Property/Counter Content Changes . . . . . 21
  - Table Aliases . . . . . 22
- fcp . . . . . 23
  - Table Row IDs . . . . . 23
  - Properties . . . . . 23
  - Counters . . . . . 23
  - Property/Counter Content Changes . . . . . 28
  - Table Aliases . . . . . 29
- fcp\_lif . . . . . 30
  - Table Row IDs . . . . . 30
  - Properties . . . . . 30
  - Counters . . . . . 31
  - Property/Counter Content Changes . . . . . 39
  - Table Aliases . . . . . 39
- fcvi . . . . . 40
  - Table Row IDs . . . . . 40
  - Properties . . . . . 40
  - Counters . . . . . 40
  - Property/Counter Content Changes . . . . . 50
  - Table Aliases . . . . . 50
- headroom\_aggregate . . . . . 51
  - Table Row IDs . . . . . 51
  - Properties . . . . . 51
  - Counters . . . . . 51
  - Property/Counter Content Changes . . . . . 53

|  |    |
|--|----|
| Table Aliases .....                    | 53 |
| headroom_cpu .....                     | 54 |
| Table Row IDs .....                    | 54 |
| Properties .....                       | 54 |
| Counters .....                         | 54 |
| Property/Counter Content Changes ..... | 58 |
| Table Aliases .....                    | 58 |
| host_adapter .....                     | 59 |
| Table Row IDs .....                    | 59 |
| Properties .....                       | 59 |
| Counters .....                         | 59 |
| Property/Counter Content Changes ..... | 60 |
| Table Aliases .....                    | 60 |
| iscsi_lif .....                        | 61 |
| Table Row IDs .....                    | 61 |
| Properties .....                       | 61 |
| Counters .....                         | 62 |
| Property/Counter Content Changes ..... | 68 |
| Table Aliases .....                    | 68 |
| iwarp .....                            | 69 |
| Table Row IDs .....                    | 69 |
| Properties .....                       | 69 |
| Counters .....                         | 69 |
| Property/Counter Content Changes ..... | 69 |
| Table Aliases .....                    | 70 |
| lif .....                              | 71 |
| Table Row IDs .....                    | 71 |
| Properties .....                       | 71 |
| Counters .....                         | 71 |
| Property/Counter Content Changes ..... | 72 |
| Table Aliases .....                    | 72 |
| lun .....                              | 73 |
| Table Row IDs .....                    | 73 |
| Properties .....                       | 73 |
| Counters .....                         | 73 |
| Property/Counter Content Changes ..... | 75 |
| Table Aliases .....                    | 75 |
| namespace .....                        | 76 |
| Table Row IDs .....                    | 76 |
| Properties .....                       | 76 |
| Counters .....                         | 76 |
| Property/Counter Content Changes ..... | 82 |
| Table Aliases .....                    | 82 |
| nfs_v4_diag .....                      | 83 |
| Table Row IDs .....                    | 83 |

|                                  |     |
|----------------------------------|-----|
| Properties                       | 83  |
| Counters                         | 83  |
| Property/Counter Content Changes | 86  |
| Table Aliases                    | 86  |
| nic_common                       | 87  |
| Table Row IDs                    | 87  |
| Properties                       | 87  |
| Counters                         | 88  |
| Property/Counter Content Changes | 93  |
| Table Aliases                    | 93  |
| nvmf_lif                         | 94  |
| Table Row IDs                    | 94  |
| Properties                       | 94  |
| Counters                         | 95  |
| Property/Counter Content Changes | 102 |
| Table Aliases                    | 102 |
| nvmf_rdma_port                   | 103 |
| Table Row IDs                    | 103 |
| Properties                       | 103 |
| Counters                         | 104 |
| Property/Counter Content Changes | 105 |
| Table Aliases                    | 105 |
| nvmf_tcp_port                    | 106 |
| Table Row IDs                    | 106 |
| Properties                       | 106 |
| Counters                         | 107 |
| Property/Counter Content Changes | 108 |
| Table Aliases                    | 108 |
| object_store_client_op           | 109 |
| Table Row IDs                    | 109 |
| Properties                       | 109 |
| Counters                         | 109 |
| Property/Counter Content Changes | 110 |
| Table Aliases                    | 111 |
| object_store_server              | 112 |
| Table Row IDs                    | 112 |
| Properties                       | 112 |
| Counters                         | 112 |
| Property/Counter Content Changes | 123 |
| Table Aliases                    | 123 |
| path                             | 124 |
| Table Row IDs                    | 124 |
| Properties                       | 124 |
| Counters                         | 124 |
| Property/Counter Content Changes | 126 |

|  |     |
|--|-----|
| Table Aliases .....                    | 126 |
| processor .....                        | 127 |
| Table Row IDs .....                    | 127 |
| Properties .....                       | 127 |
| Counters .....                         | 127 |
| Property/Counter Content Changes ..... | 128 |
| Table Aliases .....                    | 129 |
| qos .....                              | 130 |
| Table Row IDs .....                    | 130 |
| Properties .....                       | 130 |
| Counters .....                         | 131 |
| Property/Counter Content Changes ..... | 133 |
| Table Aliases .....                    | 133 |
| qos_detail .....                       | 134 |
| Table Row IDs .....                    | 134 |
| Properties .....                       | 134 |
| Counters .....                         | 134 |
| Property/Counter Content Changes ..... | 135 |
| Table Aliases .....                    | 135 |
| qos_detail_volume .....                | 136 |
| Table Row IDs .....                    | 136 |
| Properties .....                       | 136 |
| Counters .....                         | 136 |
| Property/Counter Content Changes ..... | 137 |
| Table Aliases .....                    | 137 |
| qos_resource .....                     | 138 |
| Table Row IDs .....                    | 138 |
| Properties .....                       | 138 |
| Counters .....                         | 138 |
| Property/Counter Content Changes ..... | 138 |
| Table Aliases .....                    | 139 |
| qos_volume .....                       | 140 |
| Table Row IDs .....                    | 140 |
| Properties .....                       | 140 |
| Counters .....                         | 140 |
| Property/Counter Content Changes ..... | 143 |
| Table Aliases .....                    | 143 |
| qtree .....                            | 144 |
| Table Row IDs .....                    | 144 |
| Properties .....                       | 144 |
| Counters .....                         | 144 |
| Property/Counter Content Changes ..... | 145 |
| Table Aliases .....                    | 145 |
| raid .....                             | 146 |
| Table Row IDs .....                    | 146 |

|                                  |     |
|----------------------------------|-----|
| Properties                       | 146 |
| Counters                         | 146 |
| Property/Counter Content Changes | 147 |
| Table Aliases                    | 147 |
| rewind_context                   | 148 |
| Table Row IDs                    | 148 |
| Properties                       | 148 |
| Counters                         | 148 |
| Property/Counter Content Changes | 149 |
| Table Aliases                    | 149 |
| smb2                             | 150 |
| Table Row IDs                    | 150 |
| Properties                       | 150 |
| Counters                         | 150 |
| Property/Counter Content Changes | 153 |
| Table Aliases                    | 153 |
| svm_cifs                         | 155 |
| Table Row IDs                    | 155 |
| Properties                       | 155 |
| Counters                         | 156 |
| Property/Counter Content Changes | 168 |
| Table Aliases                    | 169 |
| svm_nfs_v3                       | 170 |
| Table Row IDs                    | 170 |
| Properties                       | 170 |
| Counters                         | 170 |
| Property/Counter Content Changes | 188 |
| Table Aliases                    | 188 |
| svm_nfs_v4                       | 189 |
| Table Row IDs                    | 189 |
| Properties                       | 189 |
| Counters                         | 189 |
| Property/Counter Content Changes | 204 |
| Table Aliases                    | 204 |
| svm_nfs_v41                      | 205 |
| Table Row IDs                    | 205 |
| Properties                       | 205 |
| Counters                         | 205 |
| Property/Counter Content Changes | 225 |
| Table Aliases                    | 225 |
| svm_nfs_v42                      | 226 |
| Table Row IDs                    | 226 |
| Properties                       | 226 |
| Counters                         | 226 |
| Property/Counter Content Changes | 234 |

|  |     |
|--|-----|
| Table Aliases .....                    | 234 |
| svm_vscan .....                        | 235 |
| Table Row IDs .....                    | 235 |
| Properties .....                       | 235 |
| Counters .....                         | 235 |
| Property/Counter Content Changes ..... | 236 |
| Table Aliases .....                    | 236 |
| system .....                           | 237 |
| Table Row IDs .....                    | 237 |
| Properties .....                       | 237 |
| Counters .....                         | 238 |
| Property/Counter Content Changes ..... | 243 |
| Table Aliases .....                    | 243 |
| token_manager .....                    | 244 |
| Table Row IDs .....                    | 244 |
| Properties .....                       | 244 |
| Counters .....                         | 244 |
| Property/Counter Content Changes ..... | 245 |
| Table Aliases .....                    | 245 |
| volume .....                           | 246 |
| Table Row IDs .....                    | 246 |
| Properties .....                       | 246 |
| Counters .....                         | 247 |
| Property/Counter Content Changes ..... | 255 |
| Table Aliases .....                    | 255 |
| vscan .....                            | 256 |
| Table Row IDs .....                    | 256 |
| Properties .....                       | 256 |
| Counters .....                         | 256 |
| Property/Counter Content Changes ..... | 257 |
| Table Aliases .....                    | 257 |
| wafl .....                             | 258 |
| Table Row IDs .....                    | 258 |
| Properties .....                       | 258 |
| Counters .....                         | 258 |
| Property/Counter Content Changes ..... | 260 |
| Table Aliases .....                    | 260 |
| wafl_comp_aggr_bin .....               | 261 |
| Table Row IDs .....                    | 261 |
| Properties .....                       | 261 |
| Counters .....                         | 261 |
| Property/Counter Content Changes ..... | 262 |
| Table Aliases .....                    | 262 |
| wafl_comp_aggr_vol_bin .....           | 263 |
| Table Row IDs .....                    | 263 |

|                                  |     |
|----------------------------------|-----|
| Properties                       | 263 |
| Counters                         | 263 |
| Property/Counter Content Changes | 265 |
| Table Aliases                    | 265 |
| wafI_hya_per_aggregate           | 266 |
| Table Row IDs                    | 266 |
| Properties                       | 266 |
| Counters                         | 266 |
| Property/Counter Content Changes | 274 |
| Table Aliases                    | 275 |
| wafI_hya_sizer                   | 276 |
| Table Row IDs                    | 276 |
| Properties                       | 276 |
| Counters                         | 276 |
| Property/Counter Content Changes | 281 |
| Table Aliases                    | 282 |
| Legal notices                    | 283 |
| Copyright                        | 283 |
| Trademarks                       | 283 |
| Patents                          | 283 |
| Privacy policy                   | 283 |
| Open source                      | 283 |



# ONTAP 9.16.1 performance counter mapping

# Overview

This mapping documentation can help you continue meeting your ONTAP performance and monitoring objectives. It provides a list of the performance counter tables accessible through the ONTAP 9.15.1 REST API with a description of every counter. Each table also includes details about how the performance counters are related to the equivalent values available with the earlier ONTAPI API. This mapping information is designed to assist and support the migration of your ONTAP automation code from ONTAPI to the ONTAP REST API. More details about the documentation and how to use it are presented below.

## Transition from objects to tables

The ONTAPI API, also referred to as the Zephyr API (ZAPI), is available through the NetApp Network Manageability SDK. ONTAPI presents the ONTAP performance data as a collection of objects each with a set of counter properties. The REST API provides the equivalent data in a table format for a subset of the ONTAPI objects. Navigation begins in the sidebar with a list of the performance counter tables that are accessible through the REST API.

## Consistent description of each table

The description of every table is based on a consistent format. After a brief introduction and an indication of the related ONTAPI object, the following sections are included.

- Table row IDs

Each counter table is accessed through a unique ID. This section provides a list of the one or more ID formats supported for the table.

- Properties

This section includes the names of the ONTAPI string counters and the equivalent REST property for each.

- Counters

There is also a mapping from the numeric ONTAPI counters to the equivalent REST counters.

- Content changes

Any changes from the ONTAPI string counters to the respective REST properties are described here.

- Table aliases

This section includes a description of any aliases for the aggregated tables.

## Searching the documentation

You can search this mapping documentation using the search bar at the top of the page. If you're interested in a specific property or counter, you should enclose the search term in double quotes. Alternatively, you can download a PDF version of the entire mapping document and search the PDF locally. Click *Available PDFs* at the top left to download the PDF.

## Additional resources

There are additional resources available to help as you migrate your automation code to the ONTAP REST API.

- [ONTAP 9.16.1 ONTAPI-to-REST mapping](#)
- [ONTAP automation: Performance counters](#)

### **Help us improve the documentation**

NetApp is committed to supporting the modernization of your ONTAP applications as you transition to the REST API. You can provide feedback by becoming a [GitHub contributor](#) or by sending an email to [doccomments@netapp.com](mailto:doccomments@netapp.com). Also follow us on Twitter [@NetAppDoc](#) for notifications about the NetApp documentation.

# copy\_manager

The ONTAP copy subsystem (OCS) is the ONTAP component that processes copy-offload requests from multiple protocols. It enables a host to offload data transfer between a source and a destination within the storage array. Based on the scope of this transfer, these requests can be processed by one or more of the OCS copy engines. These include the single instance storage clone copy engine (SCE) for intra-volume copies, the byte copy engine (BCE) for inter-volume copies and the SpinNP copy engine (SpinCE) for copying across nodes in a cluster. The copy manager table reports counters that track copy-offload requests processed by the OCS copy engines and related errors.

**Classic Object:** copy\_manager

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment  |
|---|------------------|--|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.                             |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                               |
|------------------------|---------------|---|
| node_name              | node.name     | Name of the constituent Copy Manager row  |
| node_uuid              | node.uuid     | UUID for the constituent Copy Manager row |
| instance_name          | name          | Name of the Copy Manager SVM row          |
| vserver_name           | svm.name      | Name of the Copy Manager SVM row          |
| vserver_uuid           | svm.uuid      | UUID for the Copy Manager SVM row         |

# Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter      | REST Counter                     | Description  |
|------------------------------|----------------------------------|--|
| copy_reqs                    | copy_requests                    | Sum of Copy Requests received by OCM.                            |
| abort_reqs                   | abort_requests                   | Sum of Abort Requests received by OCM.                           |
| status_reqs                  | status_requests                  | Sum of Status Requests received by OCM.                          |
| copy_notify_reqs             | copy_notify_requests             | Sum of Copy Notify Requests received by OCM.                     |
| copy_revoke_reqs             | copy_revoke_requests             | Sum of Copy Revoke Requests received by OCM.                     |
| intranode_intravol_copy_reqs | intranode_intravol_copy_requests | Sum of Intra-node Intra-volume Copy Requests received.           |
| intranode_intervol_copy_reqs | intranode_intervol_copy_requests | Sum of Intra-node Inter-volume Copy Requests received.           |
| internode_copy_reqs          | internode_copy_requests          | Sum of Inter-node Copy Requests received.                        |
| copy_authchecks              | copy_authchecks                  | Sum of Copy Requests received requesting an authorization check. |
| KB_copied                    | KB_copied                        | Sum of kilo-bytes copied.  |
| KB_hole_punched              | KB_hole_punched                  | Sum of kilo-bytes that were hole-punched.                        |
| copy_notify_failures         | copy_notify_failures             | Sum of Copy Notify failures.                                     |
| copy_revoke_failures         | copy_revoke_failures             | Sum of Copy Revoke failures.                                     |
| copy_authcheck_failures      | copy_authcheck_failures          | Sum of Copy Authorization failures.                              |
| copy_failures                | copy_failures                    | Sum of Copy Request failures.                                    |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>          | <b>Description</b>   |
|--------------------------------|------------------------------|--|
| copy_fail_no_resource          | copy_fail_no_resource        | Sum of copy engine failures for: out of resources.               |
| copy_fail_authfailure          | copy_fail_authfailure        | Sum of copy engine failures for: authentication failure.         |
| copy_fail_isdir                | copy_fail_is_directory       | Sum of copy engine failures for: WAFL, is directory.             |
| copy_fail_nospace              | copy_fail_no_space           | Sum of copy engine failures for: WAFL, no space.                 |
| copy_fail_diskquota            | copy_fail_disk_quota         | Sum of copy engine failures for: WAFL, disk quota.               |
| copy_fail_jukebox              | copy_fail_could_grow         | Sum of copy engine failures for: WAFL, low space, could grow.    |
| copy_fail_offline              | copy_fail_offline            | Sum of copy engine failures for: WAFL, is offline.               |
| copy_fail_staleFH              | copy_fail_stale_filehandle   | Sum of copy engine failures for: WAFL, stale filehandle.         |
| copy_fail_io                   | copy_fail_io                 | Sum of copy engine failures for: WAFL, input/output error.       |
| copy_fail_readonly             | copy_fail_read_only          | Sum of copy engine failures for: WAFL, destination is read only. |
| copy_fail_dataunrecov          | copy_fail_data_unrecoverable | Sum of copy engine failures for: WAFL, data is unrecoverable.    |
| copy_fail_bad_msgparm          | copy_fail_bad_msg_parameter  | Sum of copy engine failures for: OCM, invalid msg parameter.     |
| copy_fail_vol_moving           | copy_fail_volume_moving      | Sum of copy engine failures for: WAFL, data is moving.           |
| copy_fail_no_volume            | copy_fail_no_volume          | Sum of copy engine failures for: WAFL, volume not available.     |
| copy_fail_delayed              | copy_fail_delayed            | Sum of copy engine failures for: WAFL, request was delayed.      |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                              | <b>Description</b>   |
|--------------------------------|--|--|
| copy_fail_must_retry           | copy_fail_must_retry                             | Sum of copy engine failures for: WAFL, retry request encountered.                            |
| copy_fail_op_not_permitted     | copy_fail_op_not_permitted                       | Sum of copy engine failures for: WAFL, operation not permitted.                              |
| copy_fail_no_access            | copy_fail_no_access                              | Sum of copy engine failures for: WAFL, access error encountered.                             |
| copy_fail_virus_detected       | copy_fail_virus_detected                         | Sum of copy engine failures for: WAFL, virus detected.                                       |
| copy_fail_data_underrun        | copy_fail_data_underrun                          | Sum of copy engine failures for: WAFL, not enough data copied.                               |
| copy_fail_data_overrun         | copy_fail_data_overrun                           | Sum of copy engine failures for: WAFL, too much data copied.                                 |
| copy_fail_locking_issue        | copy_fail_locking_issue                          | Sum of copy engine failures for: WAFL, locking error encountered.                            |
| copy_fail_fencing_issue        | copy_fail_fencing_issue                          | Sum of copy engine failures for: WAFL, fencing error encountered.                            |
| copy_fail_other                | copy_fail_other                                  | Sum of copy engine failures for: WAFL, other WAFL error.                                     |
| ocs_copy_count_curr            | ontap_copy_subsystem_current_copy_count          | Current number of copy requests being processed by the ONTAP copy subsystem.                 |
| ocs_copy_count_hwm             | ontap_copy_subsystem_copy_count_peak             | Highest number of concurrent copy requests encountered by the ONTAP copy subsystem.          |
| sce_copy_count_curr            | system_continuous_engineering_current_copy_count | Current number of copy requests being processed by the System Continuous Engineering.        |
| sce_copy_count_hwm             | system_continuous_engineering_copy_count_peak    | Highest number of concurrent copy requests encountered by the System Continuous Engineering. |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                         | <b>Description</b>   |
|--------------------------------|---|--|
| bce_copy_count_curr            | block_copy_engine_current_copy_count        | Current number of copy requests being processed by the Block Copy Engine.        |
| bce_copy_count_hwm             | block_copy_engine_copy_count_peak           | Highest number of concurrent copy requests encountered by the Block Copy Engine. |
| spince_copy_count_curr         | spince_current_copy_count                   | Current number of copy requests being processed by the SpinCE.                   |
| spince_copy_count_hwm          | spince_copy_count_peak                      | Highest number of concurrent copy requests encountered by the SpinCE.            |
| bce_copy_reqs                  | block_copy_engine_copy_requests             | Sum of Block Copy Engine Copy Requests received.                                 |
| sce_copy_reqs                  | system_continuous_engineering_copy_requests | Sum of Sislone Copy Engine Copy Requests received.                               |
| sce_kb_copied                  | system_continuous_engineering_KB_copied     | Sum of Sislone Copy Engine KB copied.  |
| bce_kb_copied                  | block_copy_engine_KB_copied                 | Sum of Block Copy Engine KB copied.  |
| spince_kb_copied               | spince_KB_copied                            | Sum of Spinnp Copy Engine KB copied.   |
| bce_ops                        | bce_ops                                     | Number of byte copy engine requests array with respect to size of operation.     |
| bce_time_latency               | bce_time_latency                            | Byte copy engine average latency array with respect to size of operation.        |
| sce_ops                        | sce_ops                                     | Number of sislone copy engine requests array with respect to size of operation.  |
| sce_time_latency               | sce_time_latency                            | Sislone copy engine average latency array with respect to size of operation.     |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>         | <b>Description</b>   |
|--------------------------------|-----------------------------|--|
| spince_ops                     | spince_ops                  | Number of spinnp copy engine requests array with respect to size of operation. |
| spince_time_latency            | spince_time_latency         | Spinnp copy engine average latency array with respect to size of operation.    |
| copy_latency_hist              | copy_latency_histogram      | Latency histogram for copy operations  |
| bce_latency_hist               | bce_latency_histogram       | Latency histogram for byte copy engine copy operations                         |
| sce_latency_hist               | sce_latency_histogram       | Latency histogram for sisclone copy engine copy operations                     |
| spince_latency_hist            | spince_latency_histogram    | Latency histogram for spinnp copy engine copy operations                       |
| bce_read_hist                  | bce_read_histogram          | Latency histogram for byte copy engine read operations                         |
| bce_write_hist                 | bce_write_histogram         | Latency histogram for byte copy engine write operations                        |
| bce_hp_hist                    | bce_hole_punch_histogram    | Latency histogram for byte copy engine hole punch operations                   |
| spince_read_hist               | spince_read_histogram       | Latency histogram for spinnp copy engine read operations                       |
| spince_write_hist              | spince_write_histogram      | Latency histogram for spinnp copy engine write operations                      |
| spince_hp_hist                 | spince_hole_punch_histogram | Latency histogram for spinnp copy engine hole punch operations                 |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| <b>REST Counter</b> | <b>Description</b>   |
|---------------------|--|
| bce_ops             | The labels have changed from bce_ops_512KB to 512KB.<br>Also bce_ops_greater_than_32MB label has changed to >32MB.               |
| bce_time_latency    | The labels have changed from bce_latency_512KB to 512KB.<br>Also bce_latency_greater_than_32MB label has changed to >32MB.       |
| sce_ops             | The labels have changed from sce_ops_512KB to 512KB.<br>Also sce_ops_greater_than_32MB label has changed to >32MB.               |
| sce_time_latency    | The labels have changed from sce_latency_512KB to 512KB.<br>Also sce_latency_greater_than_32MB label has changed to >32MB.       |
| spince_ops          | The labels have changed from spince_ops_512KB to 512KB.<br>Also spince_latency_greater_than_32MB label has changed to >32MB.     |
| spince_time_latency | The labels have changed from spince_latency_512KB to 512KB.<br>Also spince_latency_greater_than_32MB label has changed to >32MB. |

## Table Aliases

This section describes aliases for aggregated tables.

# disk

CM table for exporting disk performance counters. The alias name for disk:raid\_group is disk\_raid\_group.

**Classic Object:** disk

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                            |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {raid_group}:{raid_group_id}                | raid_group       | This represents the construction of the row ID field in raid_group combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property      | Description   |
|------------------------|--------------------|---|
| node_name              | node.name          | Node Name   |
| instance_name          | name               | Name of the disk  |
| unique_id              | unique_id          | Unique ID of the disk   |
| physical_disk_uuid     | physical_disk_id   | Unique ID of the physical disk  |
| physical_disk_name     | physical_disk_name | Name of the physical disk   |
| raid_name              | raid.name          | Name of the disk in raid terminology                                  |
| raid_group             | raid_group         | Name of the RAID group to which the disk belongs, in raid terminology |

| Classic String Counter | REST Property | Description   |
|------------------------|---------------|---|
| raid_group_id          | raid_group_id | ID of the RAID group to which the disk belongs, in raid terminology |
| raid_type              | raid.type     | Raid type of this disk  |
| disk_speed             | speed         | Disk RPM  |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter           | REST Counter                      | Description  |
|-----------------------------------|-----------------------------------|--|
| disk_capacity                     | capacity                          | Disk capacity in MB  |
| zns_wrio_msg_wait_latency_count   | zns_wrio_msg_wait.latency_count   | Total number of ZNS disk write I/O message waited per second |
| zns_wrio_msg_wait_average_latency | zns_wrio_msg_wait.average_latency | The average wait time taken for ZNS disk write I/O message   |
| time_spent_high_priority_queue    | time_spent.high_priority_queue    | Time spent in high priority queue                            |
| time_spent_guaranteed_queue       | time_spent.guaranteed_queue       | Time spent in guaranteed queue                               |
| time_spent_guaranteed_other_queue | time_spent.guaranteed_other_queue | Time spent in guaranteed other queue                         |
| time_spent_cscan_high_queue       | time_spent.cscan_high_queue       | Time spent in cscan high queue                               |
| time_spent_cscan_medium_queue     | time_spent.cscan_medium_queue     | Time spent in cscan medium queue                             |
| time_spent_cscan_medium_b_queue   | time_spent.cscan_medium_b_queue   | Time spent in cscan medium b queue                           |
| time_spent_cpio_queue             | time_spent.cpio_queue             | Time spent in cpio queue                                     |
| time_spent_deadline_queue         | time_spent.deadline_queue         | Time spent in deadline queue                                 |
| time_spent_verify_queue           | time_spent.verify_queue           | Time spent in verify queue                                   |

| <b>Classic Numeric Counter</b>      | <b>REST Counter</b>                 | <b>Description</b>  |
|-------------------------------------|-------------------------------------|---|
| zns_wrio_msg_wait_latency_histogram | zns_wrio_msg_wait.latency_histogram | This histogram provides the wait latency information for ZNS disk write I/O messages in queue                         |
| total_transfers                     | total_transfer_count                | Total number of disk operations involving data transfer initiated per second  |
| user_read_chain                     | user_read_chain                     | Average number of blocks transferred in each user read operation  |
| user_reads                          | user_read_count                     | Number of disk read operations initiated each second for retrieving data or metadata associated with user requests    |
| user_write_chain                    | user_write_chain                    | Average number of blocks transferred in each user write operation   |
| user_writes                         | user_write_count                    | Number of disk write operations initiated each second for storing data or metadata associated with user requests      |
| user_writes_in_skip_mask            | user_writes_in_skip_mask            | Number of disk write I/Os that were executed as part of a skip-mask write   |
| user_skip_write_ios                 | user_skip_write_ios                 | Number of disk skip-write operations initiated each second for storing data or metadata associated with user requests |
| cp_read_chain                       | cp_read_chain                       | Average number of blocks transferred in each consistency point read operation during a CP                             |
| cp_reads                            | cp_read_count                       | Number of disk read operations initiated each second for consistency point processing                                 |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>   |
|--------------------------------|------------------------|--|
| guarenteed_read_chain          | guarenteed_read_chain  | Average number of blocks transferred in each guaranteed read operation                             |
| guaranteed_read_chain          | guaranteed_read.chain  | Average number of blocks transferred in each guaranteed read operation                             |
| guarenteed_reads               | guarenteed_reads       | Number of disk read operations initiated each second for raid reconstruct or scrubbing activities  |
| guaranteed_reads               | guaranteed_reads       | Number of disk read operations initiated each second for raid reconstruct or scrubbing activities  |
| guarenteed_write_chain         | guarenteed_write_chain | Average number of blocks transferred in each guaranteed write operation                            |
| guaranteed_write_chain         | guaranteed_write.chain | Average number of blocks transferred in each guaranteed write operation                            |
| guarenteed_writes              | guarenteed_writes      | Number of disk write operations initiated each second for raid reconstruct or scrubbing activities |
| guaranteed_writes              | guaranteed_writes      | Number of disk write operations initiated each second for raid reconstruct or scrubbing activities |
| guaranteed_msgs                | guaranteed_msgs        | Number of guaranteed message operations per second   |
| user_read_latency              | user_read_latency      | Average latency per block in microseconds for user read operations                                 |
| user_read_blocks               | user_read_block_count  | Number of blocks transferred for user read operations per second                                   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>  |
|--------------------------------|--------------------------|---|
| read_ops                       | read_ops                 | Number of read operations per second  |
| split_ops                      | split_ops                | Numbers of operations split into child operations per second  |
| split_child_ops                | split_child_ops          | Number of child operations resulting from a split operation per second                              |
| split_child_gap_ops            | split_child_gap_ops      | Number of child operations per second resulting from a break in continuity of blocks requested      |
| split_child_hard_gap_ops       | split_child_hard_gap_ops | Number of child operations per second resulting from a hard break in continuity of blocks requested |
| read_data                      | read_data                | Amount of data read per second  |
| user_write_latency             | user_write_latency       | Average latency per block in microseconds for user write operations                                 |
| operation_latency              | operation_latency        | Average latency per operation in microseconds for read and write operations                         |
| user_write_blocks              | user_write_block_count   | Number of blocks transferred for user write operations per second                                   |
| write_ops                      | write_ops                | Number of write operations per second   |
| write_data                     | write_data               | Amount of data written per second.  |
| skip_blocks                    | skip_blocks              | Number of blocks skipped in skip-mask write operations per second                                   |
| bit_buckets                    | bit_buckets              | Number of blocks using a bit bucket per second  |
| dummy_buffers                  | dummy_buffers            | Number of blocks using a dummy buffer per second  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>          | <b>Description</b>  |
|--------------------------------|------------------------------|---|
| cp_read_latency                | cp_read_latency              | Average latency per block in microseconds for consistency point read operations |
| cp_read_blocks                 | cp_read_blocks               | Number of blocks transferred for consistency point read operations per second   |
| total_blocks                   | total_block_count            | Total number of blocks transferred for user operations per second               |
| total_data                     | total_data                   | Total throughput for user operations per second                                 |
| guarenteed_read_latency        | guarenteed_read_latency      | Average latency per block in microseconds for guaranteed read operations        |
| guaranteed_read_latency        | guaranteed_read.latency      | Average latency per block in microseconds for guaranteed read operations        |
| guarenteed_read_blocks         | guarenteed_read_blocks       | Number of blocks transferred for guaranteed read operations per second          |
| guaranteed_read_blocks         | guaranteed_read.block_count  | Number of blocks transferred for guaranteed read operations per second          |
| guarenteed_write_latency       | guarenteed_write_latency     | Average latency per block in microseconds for guaranteed write operations       |
| guaranteed_write_latency       | guaranteed_write.latency     | Average latency per block in microseconds for guaranteed write operations       |
| guarenteed_write_blocks        | guarenteed_write_blocks      | Number of blocks transferred for guaranteed write operations per second         |
| guaranteed_write_blocks        | guaranteed_write.block_count | Number of blocks transferred for guaranteed write operations per second         |
| disk_busy                      | disk_busy_percent            | The utilization percent of the disk   |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                     | <b>Description</b>   |
|--------------------------------|---|--|
| base_for_disk_busy             | base_for_disk_busy                      | Time base for disk_busy calculation  |
| io_pending                     | io_pending                              | Average number of I/Os issued to the disk for which we have not yet received the response        |
| io_pending_histogram           | io_pending_histogram                    | I/O pending histogram.   |
| io_pending_util                | io_pending_util                         | Average number of I/Os issued to the disk for measuring utilization                              |
| cp_io_pending                  | cp_io_pending                           | Average number of CP I/Os issued to the disk for which we have not yet received the response     |
| cp_msg_pending                 | cp_msg_pending                          | Average number of CP messages issued to the disk for which we have not yet received the response |
| io_queued                      | io_queued                               | Number of I/Os queued to the disk but not yet issued   |
| io_queued_histogram            | io_queued_histogram                     | I/O queued histogram.  |
| dlsched_distant                | deadline_scheduler_distant              | Number of requests with non-imminent deadlines issued by deadline scheduler.                     |
| dlsched_immediate              | deadline_scheduler_immediate            | Number of requests with imminent deadlines issued by deadline scheduler.                         |
| dlsched_distant_bg             | deadline_scheduler_distant_background   | Number of background requests with distant deadlines issued by deadline scheduler.               |
| dlsched_immediate_bg           | deadline_scheduler_immediate_background | Number of background requests with imminent deadlines issued by deadline scheduler.              |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                   | <b>Description</b>   |
|--------------------------------|---------------------------------------|--|
| dlsched_max_distant            | deadline_scheduler_maximum_distant    | Number of non-imminent requests allowed on device by deadline scheduler. |
| dlsched_max_background         | deadline_scheduler_maximum_background | Number of background requests allowed on device by deadline scheduler.   |
| dlsched_qtime                  | deadline_scheduler_queue_time         | Estimate of disk queue time used by deadline scheduler.                  |
| dlsched_svctime                | deadline_scheduler_service_time       | Estimate of disk service time used by deadline scheduler.                |
| dlsched_wait                   | deadline_scheduler_wait_time          | Average deadline scheduler wait time by QoS class.                       |
| dlsched_count                  | deadline_scheduler_count              | Count of deadline scheduler messages by QoS class.                       |
| disk_io_latency_histogram      | disk_io_latency_histogram             | Disk I/O latency histogram   |
| read_io_latency_histogram      | read_io_latency_histogram             | Read I/O latency histogram   |
| write_io_latency_histogram     | write_io_latency_histogram            | Write I/O latency histogram  |
| share_io_latency_histogram     | share_io_latency_histogram            | Share Blocks I/O latency histogram                                       |
| unmap_io_latency_histogram     | unmap_io_latency_histogram            | Unmap Blocks I/O latency histogram                                       |
| write_stream_ops               | write_stream_ops                      | Number of write stream operations per second                             |
| write_amplification_factor     | write_amplification_factor            | Average measure of the write amplification factor for SSDs               |
| host_write_blocks              | host_write_block_count                | Number of blocks written by the host as reported by the SSD              |
| physical_write_blocks          | physical_write_block_count            | Number of blocks physically written as reported by the SSD               |
| cpio_rates                     | cp_io_rates                           | Milliseconds spent at each CP I/O rate                                   |

| Classic Numeric Counter | REST Counter          | Description   |
|-------------------------|-----------------------|---|
| medium_errors           | medium_error_count    | Total medium errors reported for drive                              |
| recovered_errors        | recovered_error_count | Total recovered errors reported for drive                           |
| dma_buffered_data       | dma_buffered_data     | Amount of data copied into direct memory access buffers per second. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object  | Alias           |
|-----------------|-----------------|
| disk:raid_group | disk_raid_group |

# external\_cache

This table provides performance metrics and configuration characteristics for a given WAFL External Cache type, such as Flash Cache or the Predictive Cache Statistics simulator. High-level cache behavior can be monitored using these statistics.

**Classic Object:** ext\_cache\_obj

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description      |
|------------------------|---------------|------------------|
| node_name              | node.name     | System node name |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter        | Description  |
|-------------------------|---------------------|--|
| usage                   | usage               | Percentage of blocks in external cache currently containing valid data |
| accesses                | accesses            | External cache accesses per second                                     |
| disk_reads_replaced     | disk_reads_replaced | Estimated number of disk reads per second replaced by cache            |
| hit                     | hit.total           | Number of WAFL buffers served off the external cache                   |

| Classic Numeric Counter | REST Counter           | Description  |
|-------------------------|------------------------|--|
| hit_normal_lev0         | hit.normal_level_zero  | Number of normal level 0 WAFL buffers served off the external cache          |
| hit_metadata_file       | hit.metadata_file      | Number of metadata file buffers served off the external cache                |
| hit_directory           | hit.directory          | Number of directory buffers served off the external cache                    |
| hit_indirect            | hit.indirect           | Number of indirect file buffers served off the external cache                |
| miss                    | miss.total             | External cache misses  |
| miss_normal_lev0        | miss.normal_level_zero | External cache misses accessing normal level 0 buffers                       |
| miss_metadata_file      | miss.metadata_file     | External cache misses accessing metadata file buffers                        |
| miss_directory          | miss.directory         | External cache misses accessing directory buffers                            |
| miss_indirect           | miss.indirect          | External cache misses accessing indirect file buffers                        |
| hit_percent             | hit.percent            | External cache hit rate  |
| inserts                 | inserts                | Number of WAFL buffers inserted into the external cache                      |
| evicts                  | evicts                 | Number of blocks evicted from the external cache to make room for new blocks |
| invalidates             | invalidates            | Number of blocks invalidated in the external cache                           |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

# Table Aliases

This section describes aliases for aggregated tables.

# fcp

An FCP target port is a hardware endpoint that performs data communications over a physical link using the Fibre Channel Protocol (FCP). This table collects diagnostic and performance information for an FCP target port. The table counters can be used to diagnose link and connectivity issues on the port. These table counters also provide an overview of the ports IO performance. The alias name for fcp:node is fcp\_node.

**Classic Object:** fcp\_port

## Table Row IDs

| ID Format                       | Aggregation Type | Comment   |
|---------------------------------|------------------|---|
| {instance_name}:{instance_uuid} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{node.uuid}         | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description  |
|------------------------|---------------|--|
| node_name              | node.name     | Name of the specified node.                          |
| node_uuid              | node.uuid     | UUID of the specified node.                          |
| instance_name          | name          | Name of this FCP target port.                        |
| port_wwpn              | port.wwpn     | World Wide Port Name (WWPN) of this FCP target port. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                             | <b>Description</b>   |
|--------------------------------|---|--|
| reset_count                    | reset_count                                     | Number of physical port resets   |
| active_npiv                    | active_node_port_id_virtualization              | Number of active node port ID virtualization (NPIV) instances on this port   |
| link_down                      | link.down                                       | Number of times the Fibre Channel link was lost  |
| link_up                        | link.up   | Number of times the Fibre Channel link was established   |
| link_speed                     | link.speed                                      | Negotiated link speed  |
| queue_high_wm                  | queue_peak                                      | High watermark of the number of commands processed   |
| itn_create_latency_hist        | initiator_target_nexus_create_latency_histogram | Latency histogram for initiator target nexus (ITN) create operations   |
| isr_latency_hist               | isr.latency_histogram                           | Latency histogram for interrupt response time  |
| isr_latency                    | isr.latency                                     | Average interrupt response time  |
| isr_count                      | isr.count                                       | Number of interrupt responses  |
| queue_full                     | queue_full                                      | Number of times a queue full condition occurred.   |
| threshold_full                 | threshold_full                                  | Number of times the total number of outstanding commands on the port exceeds the threshold supported by this port. |
| queue_depth_hist               | queue_depth_histogram                           | Queue depth histogram for FCP target ports.  |
| avg_itn_create_latency         | average_initiator_target_nexus_create_latency   | Average initiator target nexus (ITN) create latency  |
| max_itn_create_latency         | maximum_initiator_target_nexus_create_latency   | Maximum initiator target nexus (ITN) create latency  |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                           | <b>Description</b>   |
|--------------------------------|---|--|
| min_itn_create_latency         | minimum_initiator_target_nexus_create_latency | Minimum initiator target nexus (ITN) create latency          |
| total_itn_create_ops           | total_initiator_target_nexus_create_ops       | Number of initiator target nexus (ITN) create operations     |
| int_count                      | interrupt_count                               | Number of interrupts   |
| shared_int_count               | shared_interrupt_count                        | Number of shared interrupts                                  |
| spurious_int_count             | spurious_interrupt_count                      | Number of spurious interrupts                                |
| loss_of_sync                   | loss_of_sync                                  | Number of times this port lost sync                          |
| loss_of_signal                 | loss_of_signal                                | Number of times this port lost signal                        |
| prim_seq_err                   | primitive_seq_err                             | Number of primitive sequence errors                          |
| lr_sent_count                  | lr.sent_count                                 | Number of LR primitives transmitted                          |
| lr_received_count              | lr.received_count                             | Number of LR primitives received                             |
| nos_received_count             | nos_received_count                            | Number of NOS primitives received                            |
| ols_received_count             | ols_received_count                            | Number of OLS primitives received                            |
| discarded_frames_count         | discarded_frames_count                        | Number of discarded frames.                                  |
| polled_intr_count              | polled_interrupt.count                        | Number of polled interrupts count.                           |
| polled_intr_preempted          | polled_interrupt.preempted                    | Number of polled interrupts preempted.                       |
| polled_intr_lost               | polled_interrupt.lost                         | Number of polled interrupts that could not be accounted for. |
| polled_intr_yielded            | polled_interrupt.yielded                      | Number of polled interrupts yielded.                         |
| read_ops                       | read_ops                                      | Number of read operations                                    |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>   | <b>Description</b>                                       |
|--------------------------------|-----------------------|--|
| write_ops                      | write_ops             | Number of write operations                               |
| other_ops                      | other_ops             | Number of operations that are not read or write.         |
| total_ops                      | total_ops             | Total number of FCP operations                           |
| read_data                      | read_data             | Amount of data read from the storage system              |
| write_data                     | write_data            | Amount of data written to the storage system             |
| total_data                     | total_data            | Amount of FCP traffic to and from the storage system     |
| average_read_latency           | average_read_latency  | Average latency for read operations                      |
| average_write_latency          | average_write_latency | Average latency for write operations                     |
| average_other_latency          | average_other_latency | Average latency for operations other than read and write |
| nvmf_read_ops                  | nvmf.read_ops         | Number of FC-NVMe read operations                        |
| nvmf_write_ops                 | nvmf.write_ops        | Number of FC-NVMe write operations                       |
| nvmf_caw_ops                   | nvmf.caw_ops          | Number of FC-NVMe CAW operations                         |
| nvmf_other_ops                 | nvmf.other_ops        | Number of NVMF operations that are not read or write.    |
| nvmf_remote_read_ops           | nvmf_remote.read_ops  | Number of FC-NVMe remote read operations                 |
| nvmf_remote_write_ops          | nvmf_remote.write_ops | Number of FC-NVMe remote write operations                |
| nvmf_remote_caw_ops            | nvmf_remote.caw_ops   | Number of FC-NVMe remote CAW operations                  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>              | <b>Description</b>   |
|--------------------------------|----------------------------------|--|
| nvmf_remote_other_ops          | nvmf_remote.other_ops            | Number of NVMF remote operations that are not read or write.       |
| nvmf_remote_total_ops          | nvmf_remote.total_ops            | Total number of remote FC-NVMe operations                          |
| nvmf_total_ops                 | nvmf.total_ops                   | Total number of FC-NVMe operations                                 |
| nvmf_read_data                 | nvmf.read_data                   | Amount of data read from the storage system (FC-NVMe)              |
| nvmf_write_data                | nvmf.write_data                  | Amount of data written to the storage system (FC-NVMe)             |
| nvmf_caw_data                  | nvmf.caw_data                    | Amount of CAW data sent to the storage system (FC-NVMe)            |
| nvmf_total_data                | nvmf.total_data                  | Amount of FC-NVMe traffic to and from the storage system           |
| nvmf_remote_read_data          | nvmf_remote.read_data            | Amount of remote data read from the storage system (FC-NVMe)       |
| nvmf_remote_write_data         | nvmf_remote.write_data           | Amount of remote data written to the storage system (FC-NVMe)      |
| nvmf_remote_caw_data           | nvmf_remote.caw_data             | Amount of remote CAW data sent to the storage system (FC-NVMe)     |
| nvmf_remote_total_data         | nvmf_remote.total_data           | Amount of remote FC-NVMe traffic to and from the storage system    |
| nvmf_avg_read_latency          | nvmf.average_read_latency        | Average latency for read operations (FC-NVMe)                      |
| nvmf_avg_write_latency         | nvmf.average_write_latency       | Average latency for write operations (FC-NVMe)                     |
| nvmf_avg_other_latency         | nvmf.average_other_latency       | Average latency for operations other than read and write (FC-NVMe) |
| nvmf_avg_remote_read_latency   | nvmf.average_remote_read_latency | Average latency for remote read operations (FC-NVMe)               |

| Classic Numeric Counter       | REST Counter                         | Description  |
|-------------------------------|--------------------------------------|--|
| nvmf_avg_remote_write_latency | nvmf.average_remote_write_latency    | Average latency for remote write operations (FC-NVMe)  |
| nvmf_avg_remote_other_latency | nvmf.average_remote_other_latency    | Average latency for remote operations other than read and write (FC-NVMe)  |
| nvmf_command_slots            | nvmf.command_slots                   | Number of command slots that have been used by initiators logging into this port. This shows the command fan-in on the port. |
| request_dropped_unknown_vp    | request_dropped_unknown_virtual_port | Number of requests dropped because there was no matching virtual port identifier.  |
| link_failure                  | link_failure                         | Number of link failures  |
| invalid_transmission_word     | invalid.transmission_word            | Number of invalid transmission words   |
| invalid_crc                   | invalid.crc                          | Number of invalid cyclic redundancy checks (CRC count)   |
| portcfg_flogi_acc             | portcfg.fabric_login_accept          | Number of fabric login (FLOGI) accept frames received  |
| portcfg_flogi_rjt             | portcfg.fabric_login_reject          | Number of port configuration changes due to fabric login (FLOGI) link service (LS) reject frames received                    |
| portcfg_flogi_timeout         | portcfg.fabric_login_timeout         | Number of fabric login (FLOGI) timeouts seen during port configuration   |
| portcfg_topology_change       | portcfg.topology_change              | Number of port configuration changes due to topology changes   |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

# Table Aliases

This section describes aliases for aggregated tables.

| Classic Object | Alias    |
|----------------|----------|
| fcp:node       | fcp_node |

# fcf\_lif

An FCF LIF is a logical interface that connects a SVM to a physical FCF port. This table collects diagnostics and performance information for an FCF LIF. The table counters can be used to debug and diagnose connectivity issues with initiators or fabric on a LIF. The table counters can be used to debug and diagnose connectivity issues with initiators or fabric at a LIF level granularity. The alias names for fcf\_lif:svm and fcf\_lif:port are fcf\_lif\_svm and fcf\_lif\_port respectively.

**Classic Object:** fcf\_lif

## Table Row IDs

| ID Format                       | Aggregation Type | Comment   |
|---------------------------------|------------------|---|
| {instance_name}:{instance_uuid} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {svm.name}:{svm.uuid}           | svm              | This represents the construction of the row ID field in svm combo object, which is a single unique string that identifies a row.  |
| {port.id}:{port.wwpn}           | port             | This represents the construction of the row ID field in port combo object, which is a single unique string that identifies a row. |
| {node.name}:{node.uuid}         | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                               |
|------------------------|---------------|---|
| node_name              | node.name     | System node name                          |
| instance_name          | name          | Name of this logical interface (LIF)      |
| instance_uuid          | unique_id     | Unique id of this logical interface (LIF) |

| Classic String Counter | REST Property | Description  |
|------------------------|---------------|--|
| vserver_name           | svm.name      | Name of the SVM that owns this logical interface (LIF) |
| vserver_uuid           | svm.uuid      | UUID of the SVM that owns this logical interface (LIF) |
| port_wwpn              | port.wwpn     | WWPN for the port                                      |
| port_id                | port.id       | ID for the port  |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter  | REST Counter             | Description  |
|--------------------------|--------------------------|--|
| read_ops                 | read_ops                 | Number of read operations                            |
| write_ops                | write_ops                | Number of write operations                           |
| other_ops                | other_ops                | Number of operations that are not read or write.     |
| total_ops                | total_ops                | Total number of operations.                          |
| read_data                | read_data                | Amount of data read from the storage system          |
| write_data               | write_data               | Amount of data written to the storage system         |
| total_data               | total_data               | Amount of FCP traffic to and from the storage system |
| ls_reject                | link_service_reject      | Number of link service (LS) requests rejected.       |
| command_sequence_error   | command.sequence_error   | Number of sequence errors seen for ELS/CT commands   |
| command_sequence_timeout | command.sequence_timeout | Number of sequence timeouts seen for ELS/CT commands |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                | <b>Description</b>   |
|--------------------------------|------------------------------------|--|
| command_terminate              | command.terminate                  | Number of terminate requests processed for IO commands                           |
| total_logins                   | total_logins                       | Total number of logins   |
| total_logouts                  | total_logouts                      | Total number of logouts  |
| inits_connected                | initiators_connected               | Number of initiators that are currently logged in to the logical interface (LIF) |
| portlogout_disc_timeout        | port_logout_disc.timeout           | Number of port logouts caused by discovery (DISC) timeouts                       |
| portlogout_disc_rjt            | port_logout_disc.reject            | Number of port logouts caused by rejected discovery (DISC) frames                |
| login_affecting_prli           | login_affecting.process_login      | Number of login-affecting process login (PRLI) frames received                   |
| login_affecting_prlo           | login_affecting.process_logout     | Number of login-affecting process logout (PRLO) frames received                  |
| login_affecting_plogi          | login_affecting.port_login         | Number of port login (PLOGI) frames received                                     |
| login_affecting_logo           | login_affecting.logout             | Number of logout (LOGO) frames received  |
| login_affecting_tprlo          | login_affecting.third_party_logout | Number of login-affecting third-party logouts (TPRLOs) received                  |
| avg_read_latency               | average_read_latency               | Average latency for read operations  |
| avg_write_latency              | average_write_latency              | Average latency for write operations   |
| avg_other_latency              | average_other_latency              | Average latency for operations other than read and write                         |
| avg_latency                    | average_latency                    | Average latency for FCP operations   |



| <b>Classic Numeric Counter</b>           | <b>REST Counter</b>                           | <b>Description</b>   |
|--|---|--|
| read_latency_hist                        | read_latency_histogram                        | Latency histogram for read operations  |
| read_cmd_data_proc_latency_hist_8K       | read_cmd_data_proc.latency_histogram_8K       | Histogram for read command processing and data delivery time for I/O less than/equal to 8K                       |
| read_cmd_data_proc_latency_hist_16K      | read_cmd_data_proc.latency_histogram_16K      | Histogram for read command processing and data delivery time for I/O greater than 8K AND less than/equal to 16K  |
| read_cmd_data_proc_latency_hist_32K      | read_cmd_data_proc.latency_histogram_32K      | Histogram for read command processing and data delivery time for I/O greater than 16K AND less than/equal to 32K |
| read_cmd_data_proc_latency_hist_64K      | read_cmd_data_proc.latency_histogram_64K      | Histogram for read command processing and data delivery time for I/O greater than 32K AND less than/equal to 64K |
| read_cmd_data_proc_latency_hist_large_IO | read_cmd_data_proc.latency_histogram_large_io | Histogram for read command processing and data delivery time for I/O greater than 64K                            |
| write_latency_hist                       | write_latency_histogram                       | Latency histogram for write operations   |
| write_cmd_proc_latency_hist_8K           | write_cmd_proc.latency_histogram_8K           | Histogram for write command processing time for I/O less than/equal to 8K  |
| write_cmd_proc_latency_hist_16K          | write_cmd_proc.latency_histogram_16K          | Histogram for write command processing time for I/O greater than 8K AND less than/equal to 16K                   |
| write_cmd_proc_latency_hist_32K          | write_cmd_proc.latency_histogram_32K          | Histogram for write command processing time for I/O greater than 16K AND less than/equal to 32K                  |

| Classic Numeric Counter                  | REST Counter                                  | Description   |
|--|---|---|
| write_cmd_proc_latency_hist_64K          | write_cmd_proc.latency_histogram_64K          | Histogram for write command processing time for I/O greater than 32K AND less than/equal to 64K |
| write_cmd_proc_latency_hist_large_IO     | write_cmd_proc.latency_histogram_large_io     | Histogram for write command processing time for I/O greater than 64K                            |
| write_data_consume_latency_hist_8K       | write_data_consume.latency_histogram_8K       | Histogram for write data consumption time for I/O less than/equal to 8K                         |
| write_data_consume_latency_hist_16K      | write_data_consume.latency_histogram_16K      | Histogram for write data consumption time for I/O greater than 8K AND less than/equal to 16K    |
| write_data_consume_latency_hist_32K      | write_data_consume.latency_histogram_32K      | Histogram for write data consumption time for I/O greater than 16K AND less than/equal to 32K   |
| write_data_consume_latency_hist_64K      | write_data_consume.latency_histogram_64K      | Histogram for write data consumption time for I/O greater than 32K AND less than/equal to 64K   |
| write_data_consume_latency_hist_large_IO | write_data_consume.latency_histogram_large_io | Histogram for write data consumption time for I/O greater than 64K                              |
| read_size_hist                           | read_size_histogram                           | Histogram of read sizes   |
| write_size_hist                          | write_size_histogram                          | Histogram of write sizes  |
| request_dropped_no_map                   | request_dropped.no_map                        | Number of requests dropped because of no valid initiator target nexus (ITN).                    |
| request_dropped_no_resource              | request_dropped.no_resource                   | Number of requests dropped because of lack of driver resources.                                 |
| request_dropped_no_hw_resource           | request_dropped.no_hw_resource                | Number of requests dropped because of lack of hardware resources.                               |

| <b>Classic Numeric Counter</b>    | <b>REST Counter</b>                   | <b>Description</b>  |
|-----------------------------------|---------------------------------------|---|
| request_dropped_no_exch_resource  | request_dropped.no_exchange_resource  | Number of requests dropped because of lack of exchange resources in the firmware.                   |
| request_taskfull_no_exch_resource | request_taskfull.no_exchange_resource | Number of requests responded with a taskfull because of lack of exchange resources in the firmware. |
| request_dropped_not_ready         | request_dropped.not_ready             | Number of requests dropped because the LIF is not ready.  |
| els_plogi_error                   | els.port_login_error                  | Number of times an error was encountered when processing a port login (PLOGI) command               |
| els_flogi_error                   | els.fabric_login_error                | Number of times an error was encountered when processing a fabric login (FLOGI) command             |
| els_fdisc_error                   | els.fabric_discovery_error            | Number of times an error was encountered when processing a fabric discovery (FDISC) command         |
| els_logo_error                    | els.logout_error                      | Number of times an error was encountered when processing a logout (LOGO) command                    |
| els_echo_error                    | els.echo_error                        | Number of times an error was encountered when processing an ECHO command                            |
| els_scr_error                     | els.state_change_register_error       | Number of times an error was encountered when processing a state change registration (SCR) command  |
| els_acc_error                     | els.accept_error                      | Number of times an error was encountered when processing an ELS ACCEPT command                      |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                            | <b>Description</b>  |
|--------------------------------|--|---|
| els_edc_error                  | els.exchange_diag_capability_error             | Number of times an error was encountered while processing an Exchange Diagnostic Capabilities (EDC) command           |
| els_rdf_error                  | els.register_diag_func_error                   | Number of times an error was encountered while processing a Register Diagnostic Functions (RDF) command               |
| ct_gppn_error                  | ct.get_physical_port_num_error                 | Number of times an error was encountered while processing a get physical port number (GPPN) command                   |
| ct_giel_error                  | ct.get_interconnect_element_list_error         | Number of times an error was encountered while processing a get interconnect element list (GIEL) command              |
| ct_gdid_error                  | ct.get_domain_id_error                         | Number of times an error was encountered while processing a get domain identifier (GDID) command                      |
| ct_gpl_error                   | ct.get_port_list_error                         | Number of times an error was encountered while processing a get port list (GPL) command                               |
| ct_gieln_error                 | ct.get_interconnect_element_logical_name_error | Number of times an error was encountered while processing a get interconnect element logical name (GIELN) command     |
| ct_gieil_error                 | ct.get_interconnect_element_info_list_error    | Number of times an error was encountered while processing a get interconnect element information list (GIEIL) command |
| ct_gapnl_error                 | ct.get_attached_port_name_list_error           | Number of times an error was encountered while processing a get attached port name list (GAPNL) command               |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>              | <b>Description</b>  |
|--------------------------------|----------------------------------|---|
| ct_gps_error                   | ct.get_port_state_error          | Number of times an error was encountered while processing a get port state (GPS) command            |
| ct_gfn_error                   | ct.get_fabric_name_error         | Number of times an error was encountered while processing a get fabric name (GFN) command           |
| ct_gfpn_id_error               | ct.get_fabric_port_name_id_error | Number of times an error was encountered while processing a get fabric port name (GFPN) command     |
| ct_ga_nxt_error                | ct.get_all_next_error            | Number of times an error was encountered while processing a get all next (GA_NXT) command           |
| ct_gid_pn_error                | ct.get_id_port_num_error         | Number of times an error was encountered while processing a get port identifier (GID_PN) command    |
| ct_gpn_id_error                | ct.get_port_name_id_error        | Number of times an error was encountered while processing a get port name (GPN_ID) command          |
| ct_gid_pt_error                | ct.get_id_port_error             | Number of times an error was encountered while processing a get port identifiers (GID_PT) command   |
| ct_rft_id_error                | ct.register_fc_type_id_error     | Number of times an error was encountered while processing a register FC-4 types (RFT_ID) command    |
| ct_rff_id_error                | ct.register_fc_feature_id_error  | Number of times an error was encountered while processing a register FC-4 features (RFF_ID) command |

| <b>Classic Numeric Counter</b>           | <b>REST Counter</b>                        | <b>Description</b>  |
|--|--|---|
| ct_rspn_id_error                         | ct.register_sym_port_name_id_error         | Number of times an error was encountered while processing a register symbolic port name (RSPN_ID) command |
| ct_rsnn_nn_error                         | ct.register_sym_node_name_error            | Number of times an error was encountered while processing a register symbolic node name (RSNN_NN) command |
| ct_gazs_error                            | ct.get_active_zone_set_error               | Number of times an error was encountered while processing a get active zone set (GAZS) command            |
| ct_gfez_error                            | ct.get_fabric_enh_zone_error               | Number of times an error was encountered while processing a get fabric enhanced zoning (GFEZ) command     |
| ct_sfez_error                            | ct.set_fabric_enh_zone_error               | Number of times an error was encountered while processing a set fabric enhanced zoning (SFEZ) command     |
| abts_completed                           | abts.completed                             | Number of ABTS successfully completed by the driver   |
| abts_drop_cmd_not_found                  | abts.drop_cmd_not_found                    | Number of ABTS frames dropped because no associated command was found                                     |
| abts_drop_itn_not_ready                  | abts.drop_initiator_target_nexus_not_ready | Number of ABTS frames dropped because no associated ITN was found   |
| abts_drop_no_resource                    | abts.drop_no_resource                      | Number of ABTS frames dropped because there were no resources available for processing                    |
| abts_nvme_fc_spdk_master_enqueue_failure | abts.nvme_fc_spdk_master_enqueue_failure   | Number of NVMe/FC ABTS events that failed to be enqueued to spdk for processing                           |

| Classic Numeric Counter                  | REST Counter                             | Description  |
|--|--|--|
| abts_nvme_fc_spdk_master_enqueue_success | abts.nvme_fc_spdk_master_enqueue_success | Number of NVMe/FC ABTS events successfully enqueued to spdk for processing         |
| gs_cmd_retry                             | gs_cmd_retry                             | Number of FC-GS commands that were retried   |
| ct_pz_rscn_success                       | ct.peer_zone_rscn_success                | Successful RSCNs initiated due to FC Peer Zone changes                             |
| ct_gapz_error                            | ct.get_active_peer_zone_error            | Errors encountered while processing a Get Active Peer Zone (GAPZ) command          |
| ct_aapz_error                            | ct.add_replace_active_peer_zone_error    | Errors encountered while processing an Add/Replace Active Peer Zone (AAPZ) command |
| ct_rapz_error                            | ct.remove_active_peer_zone_error         | Errors encountered while processing a Remove Active Peer Zone (RAPZ) command       |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object | Alias        |
|----------------|--------------|
| fcp_lif:svm    | fcp_lif_svm  |
| fcp_lif:port   | fcp_lif_port |

# fcvi

CM table for exporting FCVI counters. These counters track performance and error statistics pertaining to FCVI Interconnect adapter. These counters will be used by Engineering and Support teams to diagnose FCVI adapter issues.

**Classic Object:** fcvi

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description |
|------------------------|---------------|-------------|
| node_name              | node.name     | Node name.  |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter        | Description  |
|-------------------------|---------------------|--|
| adapter_reset           | adapter_reset       | Number of times FCVI adapter was reset.                |
| soft_reset_cnt          | soft_reset_count    | Number of times soft reset of FCVI adapter got issued. |
| hard_reset_cnt          | hard_reset_count    | Number of times hard reset of FCVI adapter got issued. |
| soft_reset_fail         | soft_reset_failures | Number of times soft reset of FCVI adapter failed.     |
| hard_reset_fail         | hard_reset_failures | Number of times hard reset of FCVI adapter failed.     |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>         | <b>Description</b>  |
|--------------------------------|-----------------------------|---|
| dma_timeout_reset              | dma_timeout_reset           | Number of times DMA timed out while resetting FCVI adapter.                               |
| cpr_reset_initiated_cnt        | cpr_reset.initiated_count   | Number of times CPR reset initiated by the FCVI adapter.                                  |
| cpr_reset_cnt                  | cpr_reset.count             | Number of times CPR reset got issued for FCVI adapter.                                    |
| cpr_reset_failure_cnt          | cpr_reset.failures_count    | Number of times CPR reset of FCVI adapter failed.   |
| cpr_reset_cancelled_cnt        | cpr_reset.cancelled_count   | Number of times CPR reset of FCVI adapter got cancelled.                                  |
| driver_reinit_skipped_cnt      | driver_reinit_skipped_count | Number of times driver re-init was skipped.   |
| fw_error                       | firmware_internal_error     | Number of times FCVI FW detected an unrecoverable internal error.                         |
| vi_error                       | vi_error                    | Number of times FCVI FW detected error with a VI and transitioned it to error state.      |
| recv_frame_error               | received_frame_error        | Number of times FCVI FW dropped an incoming frame.  |
| req_xfer_error                 | request_transfer_error      | Number of times FCVI FW detected an error while transferring data from request queue.     |
| resp_xfer_error                | response_transfer_error     | Number of times FCVI FW detected an error while transferring data to response queue.      |
| no_iu_buf                      | no_iu_buffers               | Number of times incoming connection request was dropped because there were no IU buffers. |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                | <b>Description</b>   |
|--------------------------------|------------------------------------|--|
| no_rcv_buf                     | no_received_buffers                | Number of times incoming data was dropped because no buffers were allocated. |
| link_transitions               | link_transitions                   | Number of times physical link transitioned up and down.                      |
| dropped_adv                    | dropped_advertisements             | Number of times incoming node advertisement packet got dropped.              |
| spurious_interrupts            | spurious_interrupts                | Number of times FCVI driver received spurious interrupts.                    |
| mbox_timeouts                  | mailbox.timeouts                   | Number of times mailbox command timed out.                                   |
| mbox_failures                  | mailbox.failures                   | Number of times mailbox command failed.                                      |
| create_vi_failure              | create_vi_failures                 | Number of times create VI operation failed.                                  |
| destroy_vi_failure             | destroy_vi_failures                | Number of times destroy VI operation failed.                                 |
| mem_reg_failure                | mem_registration_failures          | Number of times memory registration operations failed.                       |
| send_iu_post_failed            | send_iu.post_failures              | Number of times send IU buffer post failed.                                  |
| rcv_iu_post_failed             | receive_iu.post_failures           | Number of times receive IU buffer post failed.                               |
| rcv_iu_buf_avail               | receive_iu.buffers_available       | Available receive IU buffer count  |
| els_resp_failed                | els.response_failures              | Extended link service response fail count                                    |
| els_resp_passed                | els.response_passed                | Extended link service response pass count                                    |
| fcvi_system_adv_rcv_cnt        | system_advertisement.receive_count | System advertisement receive count   |

| <b>Classic Numeric Counter</b>   | <b>REST Counter</b>                    | <b>Description</b>  |
|----------------------------------|--|---|
| fcvi_partner_info_rcv_cnt        | partner_info_receive_count             | Partner info receive count  |
| fcvi_system_adv_drop_cnt         | system_advertisement.drop_count        | System advertisement drop count   |
| fcvi_ct_iocb_resp_fail_cnt       | ct_iocb.response_failures_count        | Common transport IOCB response fail count                               |
| fcvi_ct_iocb_resp_timeout_cnt    | ct_iocb.response_timeout_count         | Common transport IOCB response timeout count                            |
| fcvi_ct_iocb_fail_cnt            | ct_iocb.failures_count                 | Common transport IOCB fail count  |
| fcvi_adisc_fail_cnt              | address_discovery_failures_count       | Address discovery fail count  |
| fcvi_login_resp_cnt              | login_response.count                   | Login response count  |
| fcvi_login_resp_fail_cnt         | login_response.failures_count          | Login response failure count  |
| fcvi_mem_reg_resp_fail_cnt       | mem_register_response_failures_count   | Memory register response failure count                                  |
| fcvi_dereg_mem_resp_fail_cnt     | deregister_mem.response_failures_count | Memory de-register response failure count                               |
| fcvi_mem_reg_resp_cnt            | mem_register_response_count            | Memory register response count  |
| fcvi_dereg_mem_resp_cnt          | deregister_mem_response_count          | Memory de-register response count                                       |
| fcvi_postiu_buffers_fail_cnt     | post_iu.buffers_failures_count         | Post IU buffer failed count   |
| fcvi_postiu_invalid_iu_cnt       | post_iu.invalid_iu_count               | Post IU buffer failed count   |
| fcvi_dereg_mem_fail_cnt          | deregister_mem.failures_count          | Memory de-register failed count   |
| fcvi_dereg_mem_not_required      | deregister_mem.not_required            | Number of times memory de-registering was not required                  |
| fcvi_postiu_buffers_cnt_mismatch | post_iu.buffers_count_mismatch         | Number of times the posted buffer count didn't match the expected value |
| fcvi_exec_firm_mbox_fail_cnt     | exec_firm_mailbox_failures_count       | Number of times firmware execution failed                               |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                | <b>Description</b>   |
|--------------------------------|------------------------------------|--|
| fcvi_set_fw_option_fail_cnt    | set_firmware_option_failures_count | Number of times command to set firmware options failed     |
| fcvi_init_fw_fail_cnt          | init_firmware_failures_count       | Number of times firmware initialization failed             |
| fcvi_set_param_fail_cnt        | set_parameter_failures_count       | Number of times command to set parameters failed           |
| fcvi_load_risc_fail_cnt        | load_risc_failures_count           | Number of times RISC load failed                           |
| fcvi_verify_cksum_fail_cnt     | verify_checksum_failures_count     | Number of times checksum verification failed               |
| fcvi_get_fw_state_cnt          | get_firmware_state_count           | Number of times get firmware state mailbox command failed. |
| fcvi_mbox_cmd_pending_cnt      | mailbox_cmd_pending_count          | Number of pending mailbox commands                         |
| fcvi_get_id_list_cnt           | get_id_list_count                  | Number of logged in ports in firmware database             |
| fcvi_nsdb_changed_cnt          | name_server_db_changed_count       | Number of times name server database changed               |
| fcvi_pdb_changed_cnt           | point_db_changed_count             | Number of times port database changed                      |
| fcvi_rjt_err_cnt               | reject_error_count                 | Link service reject error count                            |
| fcvi_fw_err_cnt                | firmware_error_count               | FW error count   |
| fcvi_frame_err_cnt             | frame_error_count                  | Frame error count  |
| fcvi_invalid_qp_err_cnt        | invalid_qp_error_count             | Invalid QP error count                                     |
| fcvi_num_qp_cnt                | num_qp_count                       | QP count   |
| fcvi_create_qp_fail_cnt        | create_qp_failures_count           | Create QP failure count                                    |
| fcvi_destroy_qp_fail_cnt       | destroy_qp_failures_count          | Destroy QP failure count                                   |
| fcvi_io_failed_cnt             | io_failures_count                  | IO failure count   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                         | <b>Description</b>   |
|--------------------------------|---|--|
| fcvi_num_rmt_device_cnt        | num_rmt_device_count                        | Remote devices count   |
| fcvi_send_link_up_cnt          | send_link_up_count                          | Number of times link up was sent                               |
| fcvi_send_rescan_cnt           | send_rescan_count                           | Number of times rescan was done                                |
| fcvi_fw_not_rdy_cnt            | firmware.not_rdy_count                      | Firmware-not-ready count                                       |
| fcvi_gspn_info_len_err         | gspn_info_len_error_count                   | The remote system information length error count on GSPN query |
| fw_link_failure                | firmware.link_failure_count                 | Firmware reported link failure count                           |
| fw_loss_of_sync                | firmware.loss_of_sync_count                 | Firmware reported loss of sync count                           |
| fw_loss_of_signal              | firmware.loss_of_signal_count               | Firmware reported loss of signal count                         |
| fw_prim_proto_error            | firmware.prim_protocol_error_count          | Firmware reported protocol error count                         |
| fw_invalid_xmit_words          | firmware.invalid_transmit_word_count        | Firmware reported invalid transmit word count                  |
| fw_invalid_crc                 | firmware.invalid_crc_count                  | Firmware reported invalid CRC count                            |
| fw_SyStatLipOccurred           | firmware.systat.lip0_occurred_count         | Firmware reported Lip0 count                                   |
| fw_SyStatLinkUp                | firmware.systat.link_up_count               | Firmware reported link-up count                                |
| fw_SyStatLinkDwn_LoopInitTO    | firmware.systat.link_down_loop_init_to      | Firmware reported SyStatLinkDwn_LoopInitTO value               |
| fw_SyStatLinkDwn_LOS           | firmware.systat.link_down_los               | Firmware reported SyStatLinkDwn_LOS value                      |
| fw_SyStatLinkDwn_LossRcvClk    | firmware.systat.link_down_loss_received_clk | Firmware reported SyStatLinkDwn_LossRcvClk value               |
| fw_SyStatLinkDwn_NOS_OLS       | firmware.systat.link_down_nos_ols           | Firmware reported SyStatLinkDwn_NOS_OLS value                  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                | <b>Description</b>                         |
|--------------------------------|------------------------------------|--|
| fw_SyStatLipReset              | firmware.systat.lip_reset          | Firmware reported SyStatLipReset value     |
| fw_SyStatLipF7Rcvd             | firmware.systat.lipf7_received     | Firmware reported SyStatLipF7Rcvd value    |
| fw_SyStatLipF8Rcvd             | firmware.systat.lipf8_received     | Firmware reported SyStatLipF8Rcvd value    |
| fw_SyStatP2P                   | firmware.systat.p2p                | Firmware reported SyStatP2P value          |
| fw_SyStatPortCnfgChg           | firmware.systat.port_config_change | Firmware reported SyStatPortCnfgChg value  |
| fw_SyStatLbitDetected          | firmware.systat.lbit_detected      | Firmware reported SyStatLbitDetected value |
| fw_SyStatFP2P                  | firmware.systat.fp2p               | Firmware reported SyStatFP2P value         |
| fw_SyStatFL                    | firmware.systat.fl                 | Firmware reported SyStatFL value           |
| fw_SyStatPrivateLoop           | firmware.systat.private_loop       | Firmware reported SyStatPrivateLoop value  |
| fw_SyStatN2N                   | firmware.systat.n2n                | Firmware reported SyStatN2N value          |
| fw_SyStatP2pTimeOut            | firmware.systat.p2p_timeout        | Firmware reported SyStatP2pTimeOut value   |
| fw_SyStatP2pTimeOut2           | firmware.systat.p2p_timeout_2      | Firmware reported SyStatP2pTimeOut2 value  |
| fw_SyStatProtocolErr           | firmware.systat.protocol_error     | Firmware reported SyStatProtocolErr value  |
| fw_SyStatLRInitByISP           | firmware.systat.lr_init_by_isp     | Firmware reported SyStatLRInitByISP value  |
| fw_SyStatLRRcvd                | firmware.systat.lrr_cvd            | Firmware reported SyStatLRRcvd value       |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                 | <b>Description</b>                          |
|--------------------------------|-------------------------------------|---|
| fw_SyTsTOLipCnt                | firmware.sy_ts_to_lip_count         | Firmware reported SyTsTOLipCnt value        |
| fw_SyStatRspQFull              | firmware.systat.response_queue_full | Firmware reported SyStatRspQFull value      |
| fw_SyStatATIOQFull             | firmware.systat.atio_queue_full     | Firmware reported SyStatATIOQFull value     |
| fw_SyStatDropAE                | firmware.systat.drop_ae             | Firmware reported SyStatDropAE value        |
| fw_SyStatELSPROTOERR           | firmware.systat.els_protocol_error  | Firmware reported SyStatELSPROTOERR value   |
| fw_SyStatOpnFail               | firmware.systat.open_failures       | Firmware reported SyStatOpnFail value       |
| fw_SyStatTxFrames              | firmware.systat.transmit_frames     | Firmware reported SyStatTxFrames value      |
| fw_SyStatRxFrames              | firmware.systat.receive_frames      | Firmware reported SyStatRxFrames value      |
| fw_SyStatDiscardFrames         | firmware.systat.discard_frames      | Firmware reported SyStatDiscardFrames value |
| fw_SyStatDropFrmFw             | firmware.systat.drop_from_firmware  | Firmware reported SyStatDropFrmFw value     |
| fw_SyStatLIPRcvd               | firmware.systat.lip_received        | Firmware reported SyStatLIPRcvd value       |
| fw_SyStatNOSRcvd               | firmware.systat.nos_received        | Firmware reported SyStatNOSRcvd value       |
| fw_SyStatOLSRcvd               | firmware.systat.ols_received        | Firmware reported SyStatOLSRcvd value       |
| cm_num_listeners_cnt           | num_listeners_count                 | Connection Manager: Number of listeners     |
| cm_num_cm_cnt                  | num_cm_count                        | Connection Manager: CM count                |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>               | <b>Description</b>                                   |
|--------------------------------|-----------------------------------|--|
| cm_connect_sent_cnt            | connect.sent_count                | Connection Manager: Connect sent count               |
| cm_connect_failed_cnt          | connect.failures_count            | Connection Manager: Connect failed count             |
| cm_connect_passed_cnt          | connect.passed_count              | Connection Manager: Connect success count            |
| cm_connect_rcv_cnt             | connect.received_count            | Connection Manager: Connect receive count            |
| cm_connect_rjct_cnt            | connect.reject_count              | Connection Manager: Connect reject count             |
| cm_connect_ignored_cnt         | connect.ignored_count             | Connection Manager: Connect ignore count             |
| cm_discon_sent_cnt             | disconnect.sent_count             | Connection Manager: Disconnect sent count            |
| cm_discon_passed_cnt           | disconnect.passed_count           | Connection Manager: Disconnect success count         |
| cm_discon_failed_cnt           | disconnect.failures_count         | Connection Manager: Disconnect failed count          |
| cm_discon_sent_ignored_cnt     | disconnect.sent_ignored_count     | Connection Manager: Disconnect sent ignore count     |
| cm_discon_rcv_ignored_cnt      | disconnect.received_ignored_count | Connection Manager: Disconnect received ignore count |
| cm_discon_resp_ignored_cnt     | disconnect.response_ignored_count | Connection Manager: Disconnect response ignore count |
| cm_discon_timeout_cnt          | disconnect.timeout_count          | Connection Manager: Disconnect timeout count         |
| cm_discon_rcv_cnt              | disconnect.received_count         | Connection Manager: Disconnect received count        |
| cm_connecting_error_cnt        | connect.connecting_error_count    | Connection Manager: Connecting error count           |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>           | <b>Description</b>  |
|--------------------------------|-------------------------------|---|
| cm_msg_alloc_fail_cnt          | msg_allocation_failures_count | Connection Manager: Message allocation failure count                              |
| cm_connect_cnt                 | connect.count                 | Connection Manager: Connection count  |
| cm_connect_timeout_cnt         | connect.timeout_count         | Connection Manager: Connect timeout count   |
| cm_connect_info_len_err        | connect.info_len_error        | Connection Manager: Number of times we received an invalid connection info length |
| cm_connect_error_cnt           | connect.connect_error_count   | Connection Manager: Connect-error count   |
| cm_ifp_not_found_cnt           | ifp_not_found_count           | Connection Manager: IFP-not-found count   |
| cm_rmt_ifp_not_found_cnt       | rmt_ifp_not_found_count       | Connection Manager: Remote IFP-not-found count                                    |
| drop_adv_cnt                   | drop_advertisement_count      | Connection Manager: Shim layer advertisement-drop count                           |
| send_adv_cnt                   | send_advertisement_count      | Connection Manager: Shim layer's advertisement-sent count                         |
| recv_adv_cnt                   | receive_advertisement_count   | Connection Manager: Shim layer's advertisement-receive count                      |
| num_rmt_ips_cnt                | num_rmt_ips_count             | Connection Manager: Number of remote IP addresses                                 |
| rdma_write_ops                 | rdma.write_ops                | Number of RDMA write I/Os issued per second.                                      |
| rdma_write_throughput          | rdma.write_throughput         | RDMA write throughput in bytes per second.  |
| rdma_write_avg_latency         | rdma.write_average_latency    | Average RDMA write I/O latency.   |
| rdma_write_latency_histogram   | rdma.write_latency_histogram  | Latency of RDMA write operations.   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>       | <b>Description</b>      |
|--------------------------------|---------------------------|-------------------------|
| rdma_write_size_histogram      | rdma.write_size_histogram | Size of I/O operations. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# headroom\_aggregate

Display message service time variance and message inter-arrival time variance for aggregates in a node.

**Classic Object:** resource\_headroom\_aggr

## Table Row IDs

| ID Format                       | Aggregation Type | Comment  |
|---------------------------------|------------------|--|
| {instance_name}:{instance_uuid} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description   |
|------------------------|---------------|---|
| node_name              | node.name     | System node name  |
| instance_name          | name          | This is the name of the headroom_aggregate row.               |
| instance_uuid          | uuid          | This is the unique identifier for the headroom_aggregate row. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter        | Description   |
|-------------------------|---------------------|---|
| service_time            | service_time        | Disk aggregate service time variance statistics.  |
| interarrival_time       | inter_arrival_time  | Disk aggregate inter-arrival time variance statistics.                                    |
| current_utilization     | current_utilization | This is the storage aggregate average utilization of all the data disks in the aggregate. |

| <b>Classic Numeric Counter</b>  | <b>REST Counter</b>             | <b>Description</b>  |
|---------------------------------|---------------------------------|---|
| current_utilization_total       | current_utilization_denominator | This is the denominator value used to compute current utilization.  |
| current_ops                     | current_ops                     | Total number of I/Os processed by the aggregate per second.   |
| current_latency                 | current_latency                 | This is the storage aggregate average latency per message at the disk level.  |
| optimal_point_utilization       | optimal_point.utilization       | The utilization component of the optimal point of the latency/utilization curve.  |
| optimal_point_latency           | optimal_point.latency           | The latency component of the optimal point of the latency/utilization curve.  |
| optimal_point_ops               | optimal_point.ops               | The ops component of the optimal point derived from the latency/utilization curve.  |
| optimal_point_confidence_factor | optimal_point.confidence_factor | The confidence factor for the optimal point value based on the observed resource latency and utilization.   |
| optimal_point_samples           | optimal_point.samples           | The base counter for optimal_point_utilization, optimal_point_latency, optimal_point_confidence_factor, and optimal_point_ops. This is the number of one-minute samples since bootup. |
| ewma_hourly                     | ewma.hourly                     | Hourly exponential weighted moving average.   |
| ewm_std_dev_hourly              | ewm_std_dev.hourly              | Hourly exponential weighted moving standard deviation based on one-minute average values.   |
| ewma_daily                      | ewma.daily                      | Daily exponential weighted moving average.  |

| Classic Numeric Counter | REST Counter              | Description  |
|-------------------------|---------------------------|--|
| ewm_std_dev_daily       | ewm_std_dev.daily         | Daily exponential weighted moving standard deviation based on one-minute average values.   |
| ewma_weekly             | ewma.weekly               | Weekly exponential weighted moving average.  |
| ewm_std_dev_weekly      | ewm_std_dev.weekly        | Weekly exponential weighted moving standard deviation based on one-minute average values.  |
| ewma_monthly            | ewma.monthly              | Monthly exponential weighted moving average.   |
| ewm_std_dev_monthly     | ewm_std_dev.monthly       | Monthly exponential weighted moving standard deviation based on one-minute average values. |
| current_driver_qtime    | current_driver_queue_time | This is the disk driver average queue time for the storage aggregate.                      |
| observations_skipped    | observations_skipped      | The number of observations not processed.  |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter | Description   |
|--------------|---|
| uuid         | This counter now returns only the storage aggregate UUID instead of a combination of the aggregate name and aggregate UUID. |

## Table Aliases

This section describes aliases for aggregated tables.

# headroom\_cpu

This table displays message service time variance and message inter-arrival time variance for WAFL, as well as headroom optimal point information for the CPU resource.

**Classic Object:** resource\_headroom\_cpu

## Table Row IDs

| ID Format                       | Aggregation Type | Comment  |
|---------------------------------|------------------|--|
| {instance_name}:{instance_uuid} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description  |
|------------------------|---------------|--|
| node_name              | node.name     | System node name   |
| instance_name          | name          | This is the name of the headroom_cpu row.                |
| instance_uuid          | uuid          | UUID for the headroom row. This is the UUID of the node. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter   | REST Counter        | Description   |
|---------------------------|---------------------|---|
| current_utilization       | current_utilization | Average processor utilization across all processors in the system.      |
| current_utilization_total | elapsed_time        | Elapsed time since boot.  |
| current_ops               | current_ops         | Total number of operations per second (also referred to as dblade ops). |

| <b>Classic Numeric Counter</b>  | <b>REST Counter</b>             | <b>Description</b>   |
|---------------------------------|---------------------------------|--|
| current_intercluster_ops        | current_intercluster_ops        | Total number of operations per second going to the partner node.   |
| current_latency                 | current_latency                 | Current operation latency of the resource.   |
| service_time                    | service_time                    | Average service time for the CPU resource.   |
| optimal_point_utilization       | optimal_point.utilization       | Utilization component of the optimal point of the latency/utilization curve. This counter can provide an average utilization over a range of time.   |
| optimal_point_latency           | optimal_point.latency           | Latency component of the optimal point of the latency/utilization curve. This counter can provide an average latency over a range of time.   |
| optimal_point_ops               | optimal_point.ops               | Ops component of the optimal point derived from the latency/utilization curve. This counter can provide an average ops over a range of time.   |
| optimal_point_confidence_factor | optimal_point.confidence_factor | Confidence factor for the optimal point value based on the observed resource latency and utilization. The possible values are: 0 - unknown, 1 - low, 2 - medium, 3 - high. This counter can provide an average confidence factor over a range of time. |
| optimal_point_samples           | optimal_point.samples           | Base counter for optimal_point_utilization, optimal_point_latency, optimal_point_confidence_factor, and optimal_point_ops. This is the number of one-minute samples since bootup.  |
| waf_hipri_service_time          | waf_high_priority.service_time  | WAFL high priority service time variance statistics.   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                  | <b>Description</b>   |
|--------------------------------|--------------------------------------|--|
| wافل_lopri_service_time        | wافل_low_priority.service_time       | WAFL low priority service time variance statistics.  |
| wافل_cppri_service_time        | wافل_cp_priority.service_time        | WAFL cp priority service time variance statistics.   |
| wافل_cpro_service_time         | wافل_cpro.service_time               | WAFL cp promoted priority service time variance statistics.  |
| wافل_hipri_interarrival_time   | wافل_high_priority.interarrival_time | WAFL high priority inter-arrival time variance statistics.   |
| wافل_lopri_interarrival_time   | wافل_low_priority.interarrival_time  | WAFL low priority inter-arrival time variance statistics.  |
| wافل_cppri_interarrival_time   | wافل_cp_priority.interarrival_time   | WAFL cp priority inter-arrival time variance statistics.   |
| wافل_cpro_interarrival_time    | wافل_cpro.interarrival_time          | WAFL cp promoted priority inter-arrival time variance statistics.  |
| wافل_mpio_read_service_time    | wافل_mpio.read_service_time          | WAFL MP I/O successful read service time variance statistics.  |
| ewma_hourly                    | ewma.hourly                          | Hourly exponential weighted moving average for current_ops, optimal_point_ops, current_latency, optimal_point_latency, current_utilization, optimal_point_utilization and optimal_point_confidence_factor.               |
| ewm_std_dev_hourly             | ewm_std_dev.hourly                   | Hourly exponential weighted moving standard deviation based on one-minute average values, for current_ops, optimal_point_ops, current_latency, optimal_point_latency, current_utilization and optimal_point_utilization. |



| Classic Numeric Counter | REST Counter       | Description  |
|-------------------------|--------------------|--|
| ewma_daily              | ewma.daily         | Daily exponential weighted moving average for current_ops, optimal_point_ops, current_latency, optimal_point_latency, current_utilization, optimal_point_utilization and optimal_point_confidence_factor.                |
| ewm_std_dev_daily       | ewm_std_dev.daily  | Daily exponential weighted moving standard deviation based on one-minute average values, for current_ops, optimal_point_ops, current_latency, optimal_point_latency, current_utilization and optimal_point_utilization.  |
| ewma_weekly             | ewma.weekly        | Weekly exponential weighted moving average for current_ops, optimal_point_ops, current_latency, optimal_point_latency, current_utilization, optimal_point_utilization and optimal_point_confidence_factor.               |
| ewm_std_dev_weekly      | ewm_std_dev.weekly | Weekly exponential weighted moving standard deviation based on one-minute average values, for current_ops, optimal_point_ops, current_latency, optimal_point_latency, current_utilization and optimal_point_utilization. |
| ewma_monthly            | ewma.monthly       | Monthly exponential weighted moving average for current_ops, optimal_point_ops, current_latency, optimal_point_latency, current_utilization, optimal_point_utilization and optimal_point_confidence_factor.              |

| Classic Numeric Counter         | REST Counter                    | Description   |
|---------------------------------|---------------------------------|---|
| ewm_std_dev_monthly             | ewm_std_dev.monthly             | Monthly exponential weighted moving standard deviation based on one-minute average values, for current_ops, optimal_point_ops, current_latency, optimal_point_latency, current_utilization and optimal_point_utilization. |
| normalized_4k_current_ops       | normalized_4k.current_ops       | Total number of 4k current ops (normalized).  |
| normalized_4k_optimal_point_ops | normalized_4k.optimal_point_ops | Total number of 4k optimal point ops (normalized).  |
| observations_skipped            | observations_skipped            | Number of observations not processed.   |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# host\_adapter

The host\_adapter table reports activity on the Fibre Channel, Serial Attached SCSI, and parallel SCSI host adapters the storage system uses to connect to disks and tape drives.

**Classic Object:** hostadapter

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description      |
|------------------------|---------------|------------------|
| node_name              | node.name     | System node name |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter       | Description   |
|-------------------------|--------------------|---|
| total_reads             | total_read_ops     | Total number of reads on a host adapter                       |
| total_writes            | total_write_ops    | Total number of writes on a host adapter                      |
| bytes_read              | bytes_read         | Bytes read through a host adapter                             |
| bytes_written           | bytes_written      | Bytes written through a host adapter                          |
| max_link_data_rate      | max_link_data_rate | Max link data rate in Kilobytes per second for a host adapter |

| Classic Numeric Counter | REST Counter | Description                              |
|-------------------------|--------------|--|
| rscn_count              | rscn_count   | Number of RSCN(s) received by the FC HBA |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# iscsi\_lif

CM table for exporting iSCSI LIF performance counters The alias names for iscsi\_lif:svm and iscsi\_lif:node are iscsi\_lif\_svm and iscsi\_lif\_node respectively.

**Classic Object:** iscsi\_lif

## Table Row IDs

| ID Format                       | Aggregation Type | Comment   |
|---------------------------------|------------------|---|
| {instance_name}:{instance_uuid} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {svm.name}:{svm.uuid}           | svm              | This represents the construction of the row ID field in svm combo object, which is a single unique string that identifies a row.  |
| {node.name}:{node.uuid}         | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description   |
|------------------------|---------------|---|
| node_name              | node.name     | System node name  |
| instance_name          | name          | Name of this logical interface (LIF) row                        |
| instance_uuid          | uuid          | Unique ID of this logical interface (LIF) row                   |
| current_port           | current_port  | Name of the home port hosting the iSCSI logical interface (LIF) |
| vserver_name           | svm.name      | Name of the SVM owning this logical interface (LIF)             |

| <b>Classic String Counter</b> | <b>REST Property</b> | <b>Description</b>                                   |
|-------------------------------|----------------------|--|
| vserver_uuid                  | svm.uuid             | UUID for the SVM owning this logical interface (LIF) |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>   | <b>Description</b>   |
|--------------------------------|-----------------------|--|
| cmd_transferred                | cmd_transferred       | Command transferred by this iSCSI connection   |
| iscsi_read_ops                 | iscsi_read_ops        | iSCSI read operations per second on this logical interface (LIF)                           |
| iscsi_write_ops                | iscsi_write_ops       | iSCSI write operations per second on this logical interface (LIF)                          |
| iscsi_other_ops                | iscsi_other_ops       | iSCSI other operations per second on this logical interface (LIF)                          |
| total_ops                      | total_ops             | iSCSI total operations per second on this logical interface (LIF)                          |
| scsi_cmd_rcvd                  | scsi_cmd_received     | SCSI command received from iSCSI sessions on this logical interface (LIF)                  |
| nop_out_rcvd                   | nop_out_received      | iSCSI nop_out command received from iSCSI sessions on this logical interface (LIF)         |
| taskmgmt_cmd_rcvd              | taskmgmt_cmd_received | iSCSI task management command received from iSCSI sessions on this logical interface (LIF) |
| login_cmd_rcvd                 | login_cmd_received    | iSCSI login command received from iSCSI sessions on this logical interface (LIF)           |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>  |
|--------------------------------|------------------------|---|
| logout_cmd_rcvd                | logout_cmd_received    | iSCSI logout command received from iSCSI sessions on this logical interface (LIF)       |
| text_cmd_rcvd                  | text_cmd_received      | iSCSI text command received from iSCSI sessions on this logical interface (LIF)         |
| data_out_rcvd                  | data_out_received      | iSCSI data_out Command received from iSCSI sessions on this logical interface (LIF)     |
| recv_data                      | received_data          | Data received from iSCSI session on this logical interface (LIF)                        |
| snack_rcvd                     | snack_received         | iSCSI snack received from iSCSI sessions on this logical interface (LIF)                |
| unknown_pdu_rcvd               | unknown_pdu_received   | unknown iSCSI pdu received from iSCSI sessions on this logical interface (LIF)          |
| scsi_resp_sent                 | scsi_response_sent     | SCSI Command response sent from iSCSI sessions on this logical interface (LIF)          |
| nop_in_sent                    | nop_in_sent            | iSCSI nop_in response sent from iSCSI sessions on this logical interface (LIF)          |
| taskmgmt_resp_sent             | taskmgmt_response_sent | iSCSI task management response sent from iSCSI sessions on this logical interface (LIF) |
| login_resp_sent                | login_response_sent    | iSCSI login response sent from iSCSI sessions on this logical interface (LIF)           |
| logout_resp_sent               | logout_response_sent   | iSCSI logout response sent from iSCSI sessions on this logical interface (LIF)          |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b> | <b>Description</b>  |
|--------------------------------|---------------------|---|
| text_resp_sent                 | text_response_sent  | iSCSI text response sent from iSCSI sessions on this logical interface (LIF)        |
| data_in_sent                   | data_in_sent        | iSCSI data_in response sent from iSCSI sessions on this logical interface (LIF)     |
| sent_data                      | sent_data           | Data sent from iSCSI sessions on this logical interface (LIF)                       |
| r2t_sent                       | r2t_sent            | iSCSI r2t sent from iSCSI sessions on this logical interface (LIF)                  |
| reject_sent                    | reject_sent         | iSCSI reject sent from iSCSI sessions on this logical interface (LIF)               |
| async_msg_sent                 | async_msg_sent      | iSCSI asynchronous message sent from iSCSI sessions on this logical interface (LIF) |
| data_in_blocks                 | data_in.blocks      | Number of blocks sent from iSCSI sessions on this logical interface (LIF)           |
| data_out_blocks                | data_out.blocks     | number of blocks received from iSCSI sessions on this logical interface (LIF)       |
| read_data                      | read_data           | Amount of data read from the storage system in bytes                                |
| write_data                     | write_data          | Amount of data written to the storage system in bytes                               |
| total_data                     | total_data          | Amount of data read/written from/to the storage system in bytes                     |
| recv_packet                    | received_packet     | Packets(PDUs) received at the target side   |
| sent_packet                    | sent_packet         | Packets(PDUs) sent from the target side   |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                         | <b>Description</b>  |
|--------------------------------|---|---|
| recv_error                     | received_error                              | Errors received at the target side from initiator   |
| error_status_cdb               | error_status_cdb                            | Number of CDBs that contain error status from iSCSI sessions on this logical interface (LIF)                |
| success_status_cdb             | success_status_cdb                          | Number of CDBs that contain success status from iSCSI sessions on this logical interface (LIF)              |
| login_failed                   | login_failed                                | Number of login failures from iSCSI sessions on this logical interface (LIF)                                |
| taskmgmt_failed                | taskmgmt_failed                             | Number of task management failures from iSCSI sessions on this logical interface (LIF)                      |
| logout_failed                  | logout_failed                               | Number of logout failures from iSCSI sessions on this logical interface (LIF)                               |
| text_cmd_failed                | text_cmd_failed                             | Number of text command failures from iSCSI sessions on this logical interface (LIF)                         |
| protocol_errors                | protocol_errors                             | Number of protocol errors from iSCSI sessions on this logical interface (LIF)                               |
| header_digest_errors           | header_digest_errors                        | Header digest errors from iSCSI sessions on this logical interface (LIF)                                    |
| data_digest_errors             | data_digest_errors                          | Data digest errors from iSCSI sessions on this logical interface (LIF)                                      |
| outside_cmd_sn_win_pdus        | outside_cmd_sequence_number_w<br>indow_pdus | The PDUs that are out of command sequence number window from iSCSI sessions on this logical interface (LIF) |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>         | <b>Description</b>   |
|--------------------------------|-----------------------------|--|
| login_accept                   | login_accept                | Number of logins accepted from iSCSI sessions on this logical interface (LIF)  |
| login_itns                     | login_itns                  | Number of logins reported to SCSI-target for ITN creation  |
| login_authentication_errors    | login_authentication_errors | Login authentication errors from iSCSI sessions on this logical interface (LIF)  |
| login_authorization_errors     | login_authorization_errors  | Login authorization errors from iSCSI sessions on this logical interface (LIF)   |
| login_negotiation_errors       | login_negotiation_errors    | Login negotiation errors from iSCSI sessions on this logical interface (LIF)   |
| login_other_errors             | login_other_errors          | Other login errors from iSCSI sessions on this logical interface (LIF)   |
| login_redirect                 | login_redirect              | Login redirects from iSCSI sessions on this logical interface (LIF)  |
| logout_normals                 | logout_normals              | Normal logouts from iSCSI sessions on this logical interface (LIF)   |
| logout_others                  | logout_others               | Other logouts from iSCSI sessions on this logical interface (LIF)  |
| avg_read_latency               | average_read_latency        | Average latency for read operations  |
| avg_write_latency              | average_write_latency       | Average latency for write operations   |
| avg_other_latency              | average_other_latency       | Average latency for operations other than read and write (for example, Inquiry, Report LUNs, SCSI Task Management Functions) |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                | <b>Description</b>  |
|--------------------------------|------------------------------------|---|
| avg_latency                    | average_latency                    | Average latency for iSCSI operations                                      |
| avg_login_latency              | average_login_latency              | Average latency for login completions                                     |
| read_latency_hist              | read_latency_histogram             | Latency histogram for read operations                                     |
| write_latency_hist             | write_latency_histogram            | Latency histogram for write operations                                    |
| login_latency_hist             | login_latency_histogram            | Latency histogram for login completions                                   |
| socket_latency_hist            | socket_latency_histogram           | Latency histogram for socket wait time receiving a full PDU               |
| cmdproc_latency_hist           | cmd_proc_latency_histogram         | Latency histogram for command processing within iSCSI before SCSIT submit |
| cmdsub_latency_hist            | cmd_sub_latency_histogram          | Latency histogram for SCSIT command submit (from socket PDU rcv)          |
| do_pdu_latency_hist            | data_out_pdu_latency_histogram     | Latency histogram for a single DATA OUT PDU                               |
| do_scsit_latency_hist          | data_out_scsit_latency_histogram   | Latency histogram for time taken in SCSIT for DATA OUT                    |
| do_iscsi_latency_hist          | data_out_iscsi_latency_histogram   | Latency histogram for time taken in iSCSI after SCSIT callback            |
| di_scsit_latency_hist          | data_in_scsit_latency_histogram    | Latency histogram for time taken in SCSIT for DATA IN                     |
| di_iscsi_latency_hist          | data_in_iscsi_latency_histogram    | Latency histogram for time taken in iSCSI after SCSIT callback            |
| diresp_latency_hist            | data_in_response_latency_histogram | Latency histogram for Data IN SCSIT resp after SCSIT submit               |
| outpdu_latency_hist            | out_pdu_latency_histogram          | Latency histogram for out_pdu fully submitted to socket                   |

| Classic Numeric Counter | REST Counter                        | Description  |
|-------------------------|-------------------------------------|--|
| doresp_latency_hist     | data_out_response_latency_histogram | Latency histogram for Data OUT SCSI response after SCSI submit |
| read_size_hist          | read_size_histogram                 | Histogram of read sizes  |
| write_size_hist         | write_size_histogram                | Histogram of write sizes                                       |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object | Alias          |
|----------------|----------------|
| iscsi_lif:svm  | iscsi_lif_svm  |
| iscsi_lif:node | iscsi_lif_node |

# iwarp

These counters report IP based RDMA activities. RDMA is done through the iWARP protocol.

**Classic Object:** iwarp

## Table Row IDs

| ID Format | Aggregation Type | Comment  |
|-----------|------------------|--|
| {adapter} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description |
|------------------------|---------------|-------------|
| node_name              | node.name     | Node name   |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter    | Description                       |
|-------------------------|-----------------|-----------------------------------|
| iw_write_ops            | write_ops       | Number of RDMA write I/Os issued. |
| iw_read_ops             | read_ops        | Number of RDMA read I/Os issued.  |
| iw_ops                  | ops             | Number of RDMA I/Os issued.       |
| iw_avg_latency          | average_latency | Average RDMA I/O latency.         |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

# Table Aliases

This section describes aliases for aggregated tables.

# lif

These counters report activity of logical interfaces (LIFs). The alias name for lif:svm is lif\_svm.

**Classic Object:** lif

## Table Row IDs

| ID Format                       | Aggregation Type | Comment  |
|---------------------------------|------------------|--|
| {instance_name}:{instance_uuid} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                     |
| {svm.name}:{svm.id}             | svm              | This represents the construction of the row ID field in svm combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description  |
|------------------------|---------------|--|
| node_name              | node.name     | System node name                                   |
| instance_name          | name          | Name of the logical interface (LIF) row            |
| instance_uuid          | unique_id     | Unique ID for the logical interface (LIF) row      |
| vserver_name           | svm.name      | Name of the SVM                                    |
| current_port           | current_port  | Physical port hosting this logical interface (LIF) |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter     | Description                                    |
|-------------------------|------------------|--|
| up_time                 | up_time          | Interface up time                              |
| recv_data               | received_data    | Number of bytes received per second            |
| recv_packet             | received_packets | Number of packets received per second          |
| recv_errors             | received_errors  | Number of received Errors per second           |
| sent_data               | sent_data        | Number of bytes sent per second                |
| sent_packet             | sent_packets     | Number of packets sent per second              |
| sent_errors             | sent_errors      | Number of sent errors per second               |
| total_data              | total_data       | Number of bytes received and sent per second   |
| total_packet            | total_packets    | Number of packets received and sent per second |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object | Alias   |
|----------------|---------|
| lif:svm        | lif_svm |



# lun

This table contains LUN-level SAN counters which are shared between 7-mode and C-mode. These counters are available for every mapped logical unit. The alias name for lun:node is lun\_node.

**Classic Object:** lun

## Table Row IDs

| ID Format  | Aggregation Type | Comment   |
|--|------------------|---|
| {svm.name}:{instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{svm.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {node.name}:{node.uuid}                                | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                                  |
|------------------------|---------------|--|
| node_name              | node.name     | Name of the node                             |
| vserver_name           | svm.name      | Name of the vserver owning this logical unit |
| vserver_uuid           | svm.uuid      | UUID of the SVM owning this logical unit     |
| instance_uuid          | uuid          | UUID of the logical unit                     |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>   | <b>Description</b>  |
|--------------------------------|-----------------------|---|
| avg_read_latency               | average_read_latency  | Average read latency in microseconds for all operations on the LUN                                      |
| avg_write_latency              | average_write_latency | Average write latency in microseconds for all operations on the LUN                                     |
| avg_xcopy_latency              | average_xcopy_latency | Average latency in microseconds for xcopy requests  |
| caw_reqs                       | caw_requests          | Number of compare and write requests  |
| enospc                         | enospc                | Number of operations receiving ENOSPC errors  |
| queue_full                     | queue_full            | Queue full responses  |
| read_align_histo               | read_align_histogram  | Histogram of WAFL read alignment (number sectors off WAFL block start)                                  |
| read_data                      | read_data             | Read bytes  |
| read_ops                       | read_ops              | Number of read operations   |
| read_ops_sent                  | read_ops_sent         | Number of outgoing read operations  |
| read_partial_blocks            | read_partial_blocks   | Percentage of reads whose size is not a multiple of WAFL block size                                     |
| remote_bytes                   | remote_bytes          | I/O to or from a LUN which is not owned by the storage system handling the I/O.                         |
| remote_ops                     | remote_ops            | Number of operations received by a storage system that does not own the LUN targeted by the operations. |
| unmap_reqs                     | unmap_requests        | Number of unmap command requests  |

| Classic Numeric Counter | REST Counter             | Description  |
|-------------------------|--------------------------|--|
| write_align_histo       | write_align_histogram    | Histogram of WAFL write alignment (number of sectors off WAFL block start) |
| write_data              | write_data               | Write bytes  |
| write_ops               | write_ops                | Number of write operations   |
| write_ops_sent          | write_ops_sent           | Number of outgoing write operations  |
| write_partial_blocks    | write_partial_blocks     | Percentage of writes whose size is not a multiple of WAFL block size       |
| writesame_reqs          | writesame_requests       | Number of write same command requests                                      |
| writesame_unmap_reqs    | writesame_unmap_requests | Number of write same commands requests with unmap bit set                  |
| xcopy_reqs              | xcopy_requests           | Total number of xcopy operations on the LUN                                |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object | Alias    |
|----------------|----------|
| lun:node       | lun_node |

# namespace

This table contains NVMF Namespace-level counters. These counters are available for every mapped namespace.

**Classic Object:** namespace

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment  |
|---|------------------|--|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.                             |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter        | REST Property                 | Description                           |
|-------------------------------|-------------------------------|---------------------------------------|
| node_name                     | node.name                     | System node name                      |
| instance_name                 | name                          | Constituent NVMeoF Namespace          |
| instance_uuid                 | uuid                          | Constituent namespace vdisk ID        |
| aggregated_namespace_vdisk_id | aggregated_namespace_vdisk_id | Aggregated namespace vdisk ID         |
| aggregated_namespace_path     | aggregated_namespace_path     | Aggregated namespace path             |
| vserver_name                  | svm.name                      | Name of the SVM owning this namespace |
| vserver_uuid                  | svm.uuid                      | UUID of the SVM owning this namespace |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                    | <b>Description</b>  |
|--------------------------------|--|---|
| waf <sub>l</sub> _aborts_req   | waf <sub>l</sub> .aborts_requested     | Number of WAFL operations that were requested to abort                        |
| waf <sub>l</sub> _aborts_rcv   | waf <sub>l</sub> .aborts_received      | Number of WAFL operations that were aborted                                   |
| waf <sub>l</sub> _rw_errs      | waf <sub>l</sub> .read_write_op_errors | Number of WAFL read/write/compare operations that returned an error           |
| ana_transition_errs            | ana.transition_errors                  | Number of ANA transition errors returned per second.                          |
| ana_inaccess_errs              | ana.inaccessible_errors                | Number of ANA Inaccessible errors returned per second.                        |
| waf <sub>l</sub> _delete_ops   | waf <sub>l</sub> .delete_ops           | Number of WAFL operations that failed after namespace delete/unmap            |
| waf <sub>l</sub> _test_ops     | waf <sub>l</sub> .test_ops             | Number of WAFL operations effected by a test operation                        |
| internal_err                   | internal_errors                        | Number of non-fatal internal errors   |
| sgl_internal_err               | sgl_internal_errors                    | Number of non-fatal internal errors related to scatter gather list.           |
| create_sge_internal_err        | create_sge_internal_errors             | Number of non-fatal internal errors related to creation of SGEs using extents |
| read_ops                       | read_ops                               | Number of read operations   |
| write_ops                      | write_ops                              | Number of write operations  |
| compare_ops                    | compare_ops                            | Number of compare operations  |
| caw_ops                        | caw_ops                                | Number of compare and write operations  |
| other_ops                      | other_ops                              | Number of other operations  |
| remote_read_ops                | remote.read_ops                        | Number of remote read operations  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>   |
|--------------------------------|-------------------------|--|
| remote_write_ops               | remote.write_ops        | Number of remote write operations  |
| remote_compare_ops             | remote.compare_ops      | Number of remote compare operations  |
| remote_caw_ops                 | remote.caw_ops          | Number of remote compare and write operations                              |
| remote_other_ops               | remote.other_ops        | Number of remote other operations  |
| read_data                      | read_data               | Read bytes   |
| write_data                     | write_data              | Write bytes  |
| compare_data                   | compare.data            | Compare bytes  |
| caw_data                       | caw_data                | Compare and Write bytes  |
| total_data                     | total_data              | Total bytes  |
| remote_read_data               | remote.read_data        | Remote read bytes  |
| remote_write_data              | remote.write_data       | Remote write bytes   |
| remote_compare_data            | remote.compare_data     | Remote compare bytes   |
| remote_caw_data                | remote.caw_data         | Remote compare and Write bytes   |
| remote_total_data              | remote.total_data       | Total remote bytes   |
| avg_latency                    | average_latency         | Average latency in microseconds for all operations on the Namespace        |
| total_ops                      | total_ops               | Total number of operations on the Namespace                                |
| remote_avg_latency             | remote.average_latency  | Average latency in microseconds for all remote operations on the Namespace |
| remote_handling_latency        | remote.handling_latency | Average total latency in microseconds for all remote operations            |

| <b>Classic Numeric Counter</b>        | <b>REST Counter</b>                   | <b>Description</b>   |
|---------------------------------------|---------------------------------------|--|
| remote_response_ion_io_poller_latency | remote_response_ion_io_poller_latency | Average time spent in client IO poller queue at ION by remote responses. |
| remote_wafl_latency                   | remote.average_wafl_latency           | Average WAFL latency for remote operations on NSON.                      |
| remote_total_ops                      | remote.total_ops                      | Total number of remote operations on the Namespace                       |
| read_size_hist                        | read_size_histogram                   | Histogram of read sizes  |
| write_size_hist                       | write_size_histogram                  | Histogram of write sizes   |
| remote_read_size_hist                 | remote.read_size_histogram            | Histogram of remote read sizes   |
| remote_write_size_hist                | remote.write_size_histogram           | Histogram of remote write sizes  |
| large_read_ops                        | large_io.read_ops                     | Number of read operations with a request size greater than 64k           |
| large_write_ops                       | large_io.write_ops                    | Number of write operations with a request size greater than 64k          |
| remote_large_read_ops                 | remote_large_io.read_ops              | Number of remote read operations with a request size greater than 64k    |
| remote_large_write_ops                | remote_large_io.write_ops             | Number of remote write operations with a request size greater than 64k   |
| large_read_data                       | large_io.read_data                    | Read bytes from large IO   |
| large_write_data                      | large_io.write_data                   | Write bytes from large IO  |
| remote_large_read_data                | remote_large_io.read_data             | Remote read bytes from large IO  |
| remote_large_write_data               | remote_large_io.write_data            | Remote write bytes from large IO   |
| remote_server_r_ops_pending           | remote_server.read_ops_pending        | Number of remote server read operations pending in WAFL                  |
| remote_server_w_ops_pending           | remote_server.write_ops_pending       | Number of remote server write operations pending in WAFL                 |

| <b>Classic Numeric Counter</b>        | <b>REST Counter</b>                   | <b>Description</b>   |
|---------------------------------------|---------------------------------------|--|
| remote_server_caw_ops_pending         | remote_server.caw_ops_pending         | Number of remote server caw operations pending in WAFL           |
| remote_server_r_ops_responded         | remote_server.read_ops_responded      | Number of remote server read operations responded to client      |
| remote_server_w_ops_responded         | remote_server.write_ops_responded     | Number of remote server write operations responded to client     |
| remote_server_caw_ops_responded       | remote_server_caw.ops_responded       | Number of remote server caw operations responded to client       |
| remote_server_cmds_aborted            | remote_server.cmds_aborted            | Number of remote server commands aborted by client               |
| remote_server_cmds_not_found_on_abort | remote_server.cmds_not_found_on_abort | Number of remote server commands not found while trying to abort |
| abort_request                         | abort_request                         | Number of Abort requests per second.                             |
| compare_errs                          | compare.errors                        | Number of Compare Errors returned per second.                    |
| other_errs                            | other_errors                          | Number of Other Errors returned per second.                      |
| remote_ana_transition_errs            | remote_ana.transition_errors          | Number of Remote ANA transition Errors returned per second.      |
| remote_ana_inaccess_errs              | remote_ana.inaccessible_errors        | Number of Remote ANA Inaccessible Errors returned per second.    |
| remote_abort_request                  | remote_abort_requests                 | Number of Remote Aborts requests per second.                     |
| remote_compare_errs                   | remote.compare_errors                 | Number of Remote Compare Errors returned per second.             |
| remote_other_errs                     | remote.other_errors                   | Number of Other Remote Errors returned per second.               |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>  |
|--------------------------------|--------------------------------|---|
| avg_read_latency               | average_read_latency           | Average read latency in microseconds for all operations on the Namespace                    |
| avg_write_latency              | average_write_latency          | Average write latency in microseconds for all operations on the Namespace                   |
| avg_other_latency              | average_other_latency          | Average other ops latency in microseconds for all operations on the Namespace               |
| remote_avg_read_latency        | remote.average_read_latency    | Average remote read latency in microseconds for all operations on the Namespace             |
| remote_avg_write_latency       | remote.average_write_latency   | Average remote write latency in microseconds for all operations on the Namespace            |
| remote_avg_other_latency       | remote.average_other_latency   | Average other remote operations latency in microseconds for all operations on the Namespace |
| read_latency_hist              | read_latency_histogram         | Latency histogram of read operations  |
| write_latency_hist             | write_latency_histogram        | Latency histogram of write operations   |
| remote_read_latency_hist       | remote.read_latency_histogram  | Latency histogram of remote read operations   |
| remote_write_latency_hist      | remote.write_latency_histogram | Latency histogram of remote write operations  |
| read_io_align_history          | read_io_align_histogram        | Histogram of WAFL read alignment for NVMoF  |
| write_io_align_history         | write_io_align_histogram       | Histogram of WAFL write alignment for NVMoF   |
| partial_write_blocks           | partial.write_blocks           | Percent of writes whose size is not a multiple of WAFL block size                           |

| Classic Numeric Counter | REST Counter        | Description  |
|-------------------------|---------------------|--|
| partial_read_blocks     | partial.read_blocks | Percent of reads whose size is not a multiple of WAFL block size |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter                | Description   |
|-----------------------------|---|
| read_size_histogram         | For each of the labels in this histogram, all white spaces have been removed. |
| write_size_histogram        | For each of the labels in this histogram, all white spaces have been removed. |
| remote.read_size_histogram  | For each of the labels in this histogram, all white spaces have been removed. |
| remote.write_size_histogram | For each of the labels in this histogram, all white spaces have been removed. |

## Table Aliases

This section describes aliases for aggregated tables.

# nfs\_v4\_diag

The NFSv4\_diag object reports low-level diagnostics for the Network File System protocol, version 4. This is the ISOC file-sharing protocol that is predominant on UNIX platforms, used to connect to Network Attached Storage (NAS).

**Classic Object:** nfsv4\_diag

## Table Row IDs

| ID Format                       | Aggregation Type | Comment  |
|---------------------------------|------------------|--|
| {instance_name}:{instance_uuid} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                              |
|------------------------|---------------|--|
| node_name              | node.name     | Name of the node.                        |
| instance_name          | name          | Name of the row.                         |
| instance_uuid          | uuid          | Unique identifier (UUID) for the object. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter               | Description                                 |
|-------------------------|----------------------------|---|
| storePool_ClientMax     | storepool.client_maximum   | Maximum number of client objects.           |
| storePool_ClientAlloc   | storepool.client_allocated | Current number of client objects allocated. |
| storePool_SessionMax    | storepool.session_maximum  | Maximum number of session objects.          |

| <b>Classic Numeric Counter</b>                  | <b>REST Counter</b>                                     | <b>Description</b>   |
|---|---|--|
| storePool_SessionAlloc                          | storepool.session_allocated                             | Current number of session objects allocated.                             |
| storePool_SessionHolderMax                      | storepool.session_holder_maximum                        | Maximum number of session holder objects.                                |
| storePool_SessionHolderAlloc                    | storepool.session_holder_allocated                      | Current number of session holder objects allocated.                      |
| storePool_SessionConnectionHolderAlloc          | storepool.session_connection_holder_allocated           | Current number of session connection holder objects allocated.           |
| storePool_SessionConnectionHolderMax            | storepool.session_connection_holder_maximum             | Maximum number of session connection holder objects.                     |
| storePool_ConnectionParentSessionReferenceAlloc | storepool.connection_parent_session_reference_allocated | Current number of connection parent session reference objects allocated. |
| storePool_ConnectionParentSessionReferenceMax   | storepool.connection_parent_session_reference_maximum   | Maximum number of connection parent session reference objects.           |
| storePool_StringMax                             | storepool.string_maximum                                | Maximum number of string objects.  |
| storePool_StringAlloc                           | storepool.string_allocated                              | Current number of string objects allocated.                              |
| storePool_OwnerMax                              | storepool.owner_maximum                                 | Maximum number of owner objects.   |
| storePool_OwnerAlloc                            | storepool.owner_allocated                               | Current number of owner objects allocated.                               |
| storePool_OpenStateMax                          | storepool.openstate_maximum                             | Maximum number of open state objects.                                    |
| storePool_OpenStateAlloc                        | storepool.openstate_allocated                           | Current number of open state objects allocated.                          |
| storePool_DelegStateMax                         | storepool.delegation_state_maximum                      | Maximum number of delegation state objects.                              |
| storePool_DelegStateAlloc                       | storepool.delegation_state_allocated                    | Current number of delegation state objects allocated.                    |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>              | <b>Description</b>                                   |
|--------------------------------|----------------------------------|--|
| storePool_LockStateMax         | storepool.lock_state_maximum     | Maximum number of lock state objects.                |
| storePool_LockStateAlloc       | storepool.lock_state_allocated   | Current number of lock state objects allocated.      |
| storePool_LayoutStateMax       | storepool.layout_state_maximum   | Maximum number of layout state objects.              |
| storePool_LayoutStateAlloc     | storepool.layout_state_allocated | Current number of layout state objects allocated.    |
| storePool_CopyStateMax         | storepool.copy_state_maximum     | Maximum number of copy state objects.                |
| storePool_CopyStateAlloc       | storepool.copy_state_allocated   | Current number of copy state objects allocated.      |
| storePool_OpenMax              | storepool.open_maximum           | Maximum number of share lock objects.                |
| storePool_OpenAlloc            | storepool.open_allocated         | Current number of share objects allocated.           |
| storePool_DelegMax             | storepool.delegation_maximum     | Maximum number delegation lock objects.              |
| storePool_DelegAlloc           | storepool.delegation_allocated   | Current number of delegation lock objects allocated. |
| storePool_LayoutMax            | storepool.layout_maximum         | Maximum number of layout objects.                    |
| storePool_LayoutAlloc          | storepool.layout_allocated       | Current number of layout objects allocated.          |
| storepool_lock_layout_max      | storepool.lock_layout_maximum    | Maximum number of layout lock objects.               |
| storepool_lock_layout_alloc    | storepool.lock_layout_allocated  | Current number of layout lock objects allocated.     |
| storePool_ByteLockMax          | storepool.byte_lock_maximum      | Maximum number of byte range lock objects.           |

| Classic Numeric Counter        | REST Counter                                | Description  |
|--------------------------------|---|--|
| storePool_ByteLockAlloc        | storepool.byte_lock_allocated               | Current number of byte range lock objects allocated.                   |
| storePool_StateRefHistoryMax   | storepool.state_reference_history_maximum   | Maximum number of state reference callstack history objects.           |
| storePool_StateRefHistoryAlloc | storepool.state_reference_history_allocated | Current number of state reference callstack history objects allocated. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# nic\_common

This table tracks hardware network traffic performance and errors for all supported network interface cards (NIC), such as Intel Niantic, Qlogic, and CNA.

**Classic Object:** nic\_common

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property      | Description  |
|------------------------|--------------------|--|
| node_name              | node.name          | Node name.   |
| node_uuid              | node.uuid          | Node UUID  |
| nic_type               | type               | NIC card type, such as ixgbe or qlge                                       |
| link_current_state     | link_current_state | Current link state   |
| link_media_state       | link_media_state   | Current media state is active or no carrier                                |
| link_duplex            | link_duplex        | Link duplex setting is half or full  |
| link_speed             | link_speed         | Link speed   |
| link_flowcontrol       | link_flowcontrol   | Current link flow control state (Transmit or Receive or Both)              |
| rss_enabled            | rss_enabled        | Indicates whether the interface is enabled for RSS. Value can be on or off |

# Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>         | <b>Description</b>                               |
|--------------------------------|-----------------------------|--|
| nic_ipspace                    | ipspace                     | IPSpace of an interface                          |
| rx_frames_per_sec              | receive_frames_per_sec      | Frames received per second                       |
| rx_frames                      | receive_frames              | Frames received                                  |
| rx_bytes_per_sec               | receive_bytes_per_sec       | Bytes received per second                        |
| rx_bytes                       | receive_bytes               | Bytes received                                   |
| rx_errors_per_min              | receive_errors_per_min      | Error received per minute                        |
| rx_errors                      | receive_errors              | Error received                                   |
| rx_discards_per_min            | receive_discards_per_min    | Discarded receiving frames per minute            |
| rx_discards                    | receive_discards            | Discarded receiving frames                       |
| rx_total_frames                | receive_total_frames        | Total frames received                            |
| rx_total_bytes                 | receive_total_bytes         | Total bytes received                             |
| rx_total_errors                | receive_total_errors        | Total errors received                            |
| rx_total_discards              | receive_total_discards      | Total queue drop or discard frame received       |
| rx_multi_broadcast             | receive_multicast_broadcast | Multicast or broadcast frames received           |
| rx_no_buffers                  | receive_no_buffers          | No receiving buffer available counter            |
| rx_non_primary_uc              | receive_non_primary_unicast | Frames with non-primary unicast address received |
| rx_lro_segments                | receive_lro_segments        | LRO (Large Receive Offload) segments received    |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>          | <b>Description</b>                                       |
|--------------------------------|------------------------------|--|
| rx_lro_bytes                   | receive_lro_bytes            | LRO (Large Receive Offload) bytes received               |
| rx_lro6_segments               | receive_lro6_segments        | IPv6 frame LRO (Large Receive Offload) segments received |
| rx_lro6_bytes                  | receive_lro6_bytes           | IPv6 frame LRO (Large Receive Offload) bytes received    |
| rx_mcast_v6_solicit            | receive_multicast_v6_solicit | IPv6 multicast received                                  |
| rx_tag_drop                    | receive_tag_drop             | Dropped frames with tag received                         |
| rx_vlan_tag_drop               | receive_vlan_tag_drop        | Dropped frames with tag on a VLAN received               |
| rx_vlan_untag_drop             | receive_vlan_untag_drop      | Dropped frames without tag received                      |
| rx_vlan_forwards               | receive_vlan_forwards        | Frames received which are forwarded on the VLAN          |
| rx_vlan_broadcasts             | receive_vlan_broadcasts      | Frames received which are broadcasted onto a VLAN        |
| rx_vlan_unicasts               | receive_vlan_unicasts        | Frames received which are unicasted on the VLAN          |
| tx_frames_per_sec              | transmit_frames_per_sec      | Frames sent per second                                   |
| tx_frames                      | transmit_frames              | Frames sent  |
| tx_bytes_per_sec               | transmit_bytes_per_sec       | Bytes sent per second                                    |
| tx_bytes                       | transmit_bytes               | Bytes sent   |
| tx_errors_per_min              | transmit_errors_per_min      | Error sent per minute                                    |
| tx_errors                      | transmit_errors              | Error sent   |
| tx_discards_per_min            | transmit_discards_per_min    | Discarded frames sent per minute                         |
| tx_discards                    | transmit_discards            | Discards frames sent                                     |

| <b>Classic Numeric Counter</b>   | <b>REST Counter</b>           | <b>Description</b>  |
|----------------------------------|-------------------------------|---|
| tx_total_frames                  | transmit_total_frames         | Total frames sent   |
| tx_total_bytes                   | transmit_total_bytes          | Total bytes sent  |
| tx_total_errors                  | transmit_total_errors         | Total errors sent   |
| tx_total_discards                | transmit_total_discards       | Total queue drops on sending                                  |
| tx_multi_broadcast               | transmit_multicast_broadcast  | Total multicast/broadcast frames sent                         |
| tx_queue_overflows               | transmit_queue_overflows      | Sending queue overflows counter                               |
| tx_no_buffers                    | transmit_no_buffers           | No sending buffers available count                            |
| tx_frames_queued                 | transmit_frames_queued        | Sent frames queued  |
| tx_buffer_coalesces (Deprecated) | transmit_buffer_coalesces     | Sending coalesces buffers                                     |
| tx_mtus_too_big                  | transmit_mtus_too_big         | Oversized packets sent (MTUs too big)                         |
| tx_tso_segments                  | transmit_tso_segments         | TSO (TCP Segmentation Offload) segments sent                  |
| tx_tso_bytes                     | transmit_tso_bytes            | TSO (TCP Segmentation Offload) bytes sent                     |
| tx_tso6_segments                 | transmit_tso6_segments        | IPv6 frame TSO (TCP Segmentation Offload) segments sent       |
| tx_tso6_bytes                    | transmit_tso6_bytes           | IPv6 frame TSO (TCP Segmentation Offload) bytes sent          |
| tx_mcast_v6_solicit              | transmit_multicast_v6_solicit | IPv6 multicast transmitted                                    |
| hwassist                         | hwassist                      | HW offload capability of the interface                        |
| tx_ltm_fastp_calls (Deprecated)  | transmit_ltm_fast_path_calls  | LTM (LAN Topology Module) sending fast path call counter      |
| tx_ltm_busy_calls (Deprecated)   | transmit_ltm_busy_calls       | LTM (LAN Topology Module) sending failed or busy call counter |

| <b>Classic Numeric Counter</b>   | <b>REST Counter</b>       | <b>Description</b>   |
|----------------------------------|---------------------------|--|
| tx_ltm_update_calls (Deprecated) | transmit_ltm_update_calls | LTM (LAN Topology Module) sending update call counter  |
| tx_ltm_yield_calls (Deprecated)  | transmit_ltm_yield_calls  | LTM (LAN Topology Module) sending yield call counter   |
| tx_ltm_total_calls (Deprecated)  | transmit_ltm_total_calls  | LTM (LAN Topology Module) sending call counter   |
| total_bytes                      | total_bytes               | Bytes received and sent  |
| total_errors                     | total_errors              | Error received and sent  |
| rx_crc_errors                    | receive_crc_errors        | CRC errors detected on received packets  |
| rx_unsupported_op                | receive_unsupported_op    | Total number of unsupported MAC Control Opcode received in Ethernet MAC control frame format field |
| rx_length_errors                 | receive_length_errors     | Length errors detected on received packets   |
| rx_runt_frames                   | receive_runt_frames       | Very short frames received   |
| rx_fragment                      | receive_fragment          | Fragmented frames received   |
| rx_long_frames                   | receive_long_frames       | Very long frames received  |
| rx_jabber                        | receive_jabber            | Jabber frames received   |
| rx_illegal_symbol                | receive_illegal_symbol    | Illegal symbol frames received   |
| rx_error_symbol                  | receive_error_symbol      | Error symbol frames received   |
| rx_alignment_errors              | receive_alignment_errors  | Alignment errors detected on received packets  |
| rx_bus_overruns                  | receive_bus_overruns      | Bus overruns received  |
| rx_xon                           | receive_xon               | Xon frames received  |
| rx_xoff                          | receive_xoff              | Xoff frames received   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>          | <b>Description</b>   |
|--------------------------------|------------------------------|--|
| rx_jumbo                       | receive_jumbo                | Jumbo frames received  |
| tx_hw_errors                   | transmit_hw_errors           | Transmit errors reported by hardware   |
| tx_link_down_drop              | transmit_link_down_drop      | Transmit packets dropped by hardware because the link was down   |
| tx_cfg_up_down                 | transmit_config_up_down      | Link down state changes  |
| tx_bus_overruns                | transmit_bus_overruns        | Bus overruns sent  |
| collisions                     | collisions                   | Collisions on CSMA interfaces  |
| tx_max_collisions              | transmit_max_collisions      | Maximum collisions sent  |
| tx_single_collisions           | transmit_single_collisions   | Single collisions sent   |
| tx_multi_collisions            | transmit_multiple_collisions | Multiple collisions sent   |
| tx_late_collisions             | transmit_late_collisions     | Late collisions sent   |
| tx_xon                         | transmit_xon                 | Transmitter on (XON) frames sent   |
| tx_xoff                        | transmit_xoff                | Transmitter Off (XOFF) frames sent   |
| tx_jumbo                       | transmit_jumbo               | Jumbo frames sent  |
| link_up_to_downs               | link_up_to_down              | Number of link state change from UP to DOWN.   |
| rss_itable_updates             | rss_itable_updates           | Number of hash table updates   |
| rss_hash_errors                | rss_hash_errors              | Number of hash errors  |
| rss_num_of_queues_used         | rss_num_of_queues_used       | Number of queues used in RSS   |
| rss_matrix                     | rss_matrix                   | Matrix of RSS queue stats. The distribution of packet processing on each network queue. Each queue will have Transmit/Receive frames/bytes counting and requeue counting |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b> | <b>Description</b>                    |
|--------------------------------|---------------------|---------------------------------------|
| rss_cg_stat                    | rss_cg_stat         | RSS receive queue to CG queue mapping |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# nvmf\_lif

An NVMe/FC LIF is a logical interface that connects an SVM to a NVMe/FC capable physical port. This table collects diagnostics and performance information for an NVMF LIF. The table counters can be used to debug and diagnose connectivity issues with initiators or fabric on a LIF. The table counters can be used to debug and diagnose connectivity issues with initiators or fabric at a LIF level granularity. The alias names for nvmf\_lif:node and nvmf\_lif:port are nvmf\_lif\_node and nvmf\_lif\_port respectively.

**Classic Object:** nvmf\_fc\_lif

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {port_id}:{port_wwpn}                       | port             | This represents the construction of the row ID field in port combo object, which is a single unique string that identifies a row. |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                          |
|------------------------|---------------|--------------------------------------|
| node_name              | node.name     | Node name                            |
| node_uuid              | node.uuid     | Node UUID                            |
| instance_name          | name          | Name of this logical interface (LIF) |
| instance_uuid          | uuid          | UUID of this logical interface (LIF) |

| Classic String Counter | REST Property       | Description  |
|------------------------|---------------------|--|
| aggregated_lif_uuid    | aggregated_lif_uuid | Aggregated NVMF logical interface row ID   |
| aggregated_lif_name    | aggregated_lif_name | Aggregated NVME LIF name   |
| vserver_name           | svm.name            | Name of the SVM (formerly referred to as a Vserver) that owns this logical interface (LIF) |
| vserver_uuid           | svm.uuid            | UUID of the SVM (formerly referred to as a Vserver) that owns this logical interface (LIF) |
| port_wwpn              | port_wwpn           | WWPN for the port  |
| port_id                | port_id             | ID for the port  |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter     | Description  |
|-------------------------|------------------|--|
| read_ops                | read_ops         | Number of read operations  |
| write_ops               | write_ops        | Number of write operations   |
| compare_ops             | compare_ops      | Number of compare operations   |
| caw_ops                 | caw_ops          | Number of compare and write operations                                       |
| other_ops               | other_ops        | Number of operations that are not read, write, compare or compare-and-write. |
| total_ops               | total_ops        | Total number of operations.  |
| remote_read_ops         | remote_read_ops  | Number of remote read operations   |
| remote_write_ops        | remote_write_ops | Number of remote write operations  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>  | <b>Description</b>  |
|--------------------------------|----------------------|---|
| remote_compare_ops             | remote_compare_ops   | Number of remote compare operations   |
| remote_caw_ops                 | remote_caw_ops       | Number of remote compare and write operations                                       |
| remote_other_ops               | remote_other_ops     | Number of remote operations that are not read, write, compare or compare-and-write. |
| remote_total_ops               | remote_total_ops     | Total number of remote operations.  |
| read_data                      | read_data            | Amount of data read from the storage system   |
| write_data                     | write_data           | Amount of data written to the storage system  |
| compare_data                   | compare_data         | Amount of data compared in the storage system                                       |
| caw_data                       | caw_data             | Amount of data compared and written to in the storage system                        |
| total_data                     | total_data           | Amount of NVMF traffic to and from the storage system                               |
| remote_read_data               | remote_read_data     | Amount of remote data read from the storage system                                  |
| remote_write_data              | remote_write_data    | Amount of remote data written to the storage system                                 |
| remote_compare_data            | remote_compare_data  | Amount of remote data compared in the storage system                                |
| remote_caw_data                | remote_caw_data      | Amount of remote data compared and written to in the storage system                 |
| remote_total_data              | remote_total_data    | Amount of NVMF remote traffic to and from the storage system                        |
| avg_read_latency               | average_read_latency | Average latency for read operations   |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>  |
|--------------------------------|--------------------------------|---|
| avg_write_latency              | average_write_latency          | Average latency for write operations  |
| avg_other_latency              | average_other_latency          | Average latency for operations other than read, write, compare or compare-and-write.                            |
| avg_latency                    | average_latency                | Average latency for NVMF operations   |
| remote_avg_read_latency        | remote_average_read_latency    | Average latency for remote read operations  |
| remote_avg_write_latency       | remote_average_write_latency   | Average latency for remote write operations   |
| remote_avg_other_latency       | remote_average_other_latency   | Average latency for operations other than read, write, compare or compare-and-write.                            |
| remote_avg_latency             | remote_average_latency         | Average latency for NVMF remote operations  |
| read_latency_hist              | read_latency_histogram         | Latency histogram for read operations   |
| remote_read_latency_hist       | remote_read_latency_histogram  | Latency histogram for remote read operations  |
| pend_q_hist                    | pending_queue_histogram        | Latency histogram when IOs are placed in pending queue  |
| driver_read_latency_hist       | driver_read_latency_histogram  | Driver latency histogram for read operations. Records the time spent in the protocol stack for read operations. |
| storage_read_latency_hist      | storage_read_latency_histogram | Storage latency histogram for read operations. Records the time spent in storage for read operations.           |
| write_latency_hist             | write_latency_histogram        | Latency histogram for write operations  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                       | <b>Description</b>  |
|--------------------------------|---|---|
| remote_write_latency_hist      | remote_write_latency_histogram            | Latency histogram for remote write operations   |
| driver_write_latency_hist      | driver_write_latency_histogram            | Driver latency histogram for write operations. Records the time spent in the protocol stack for write operations.                     |
| storage_write_latency_hist     | storage_write_latency_histogram           | Storage latency histogram for write operations. Records the time spent in storage for write operations.                               |
| fabric_latency_hist            | fabric_latency_histogram                  | Fabric latency histogram for write operations. Records the time spent in the fabric after sending a transfer-ready to get WRITE data. |
| read_size_hist                 | read_size_histogram                       | Histogram of read sizes   |
| write_size_hist                | write_size_histogram                      | Histogram of write sizes  |
| remote_read_size_hist          | remote_read_size_histogram                | Histogram of remote read sizes  |
| remote_write_size_hist         | remote_write_size_histogram               | Histogram of remote write sizes   |
| read_cmd_proc_latency_hist_8K  | read_cmd_processing_latency_histogram_8K  | Histogram for read command processing and data delivery time for I/O less than/equal to 8K.   |
| read_cmd_proc_latency_hist_16K | read_cmd_processing_latency_histogram_16K | Histogram for read command processing and data delivery time for I/O greater than 8K AND less than/equal to 16K.                      |
| read_cmd_proc_latency_hist_32K | read_cmd_processing_latency_histogram_32K | Histogram for read command processing and data delivery time for I/O greater than 16K AND less than/equal to 32K.                     |

| <b>Classic Numeric Counter</b>             | <b>REST Counter</b>                                   | <b>Description</b>   |
|--|---|--|
| read_cmd_proc_latency_hist_64K             | read_cmd_processing_latency_histogram_64K             | Histogram for read command processing and data delivery time for I/O greater than 32K AND less than/equal to 64K.        |
| read_cmd_proc_latency_hist_large_io        | read_cmd_processing_latency_histogram_large_io        | Histogram for read command processing and data delivery time for I/O greater than 64K.                                   |
| remote_read_cmd_proc_latency_hist_8K       | remote_read_cmd_processing_latency_histogram_8K       | Histogram for remote read command processing and data delivery time for I/O less than/equal to 8K.                       |
| remote_read_cmd_proc_latency_hist_16K      | remote_read_cmd_processing_latency_histogram_16K      | Histogram for remote read command processing and data delivery time for I/O greater than 8K AND less than/equal to 16K.  |
| remote_read_cmd_proc_latency_hist_32K      | remote_read_cmd_processing_latency_histogram_32K      | Histogram for remote read command processing and data delivery time for I/O greater than 16K AND less than/equal to 32K. |
| remote_read_cmd_proc_latency_hist_64K      | remote_read_cmd_processing_latency_histogram_64K      | Histogram for remote read command processing and data delivery time for I/O greater than 32K AND less than/equal to 64K. |
| remote_read_cmd_proc_latency_hist_large_io | remote_read_cmd_processing_latency_histogram_large_io | Histogram for remote remote read command processing and data delivery time for I/O greater than 64K.                     |
| write_cmd_proc_latency_hist_8K             | write_cmd_processing_latency_histogram_8K             | Histogram for write command processing and data delivery time for I/O less than/equal to 8K.                             |

| <b>Classic Numeric Counter</b>         | <b>REST Counter</b>                               | <b>Description</b>  |
|--|---|---|
| write_cmd_proc_latency_hist_16K        | write_cmd_processing_latency_histogram_16K        | Histogram for write command processing and data delivery time for I/O greater than 8K AND less than/equal to 16K.         |
| write_cmd_proc_latency_hist_32K        | write_cmd_processing_latency_histogram_32K        | Histogram for write command processing and data delivery time for I/O greater than 16K AND less than/equal to 32K.        |
| write_cmd_proc_latency_hist_64K        | write_cmd_processing_latency_histogram_64K        | Histogram for write command processing and data delivery time for I/O greater than 32K AND less than/equal to 64K.        |
| write_cmd_proc_latency_hist_large_io   | write_cmd_processing_latency_histogram_large_io   | Histogram for write command processing and data delivery time for I/O greater than 64K.                                   |
| remote_write_cmd_proc_latency_hist_8K  | remote_write_cmd_processing_latency_histogram_8K  | Histogram for remote write command processing and data delivery time for I/O less than/equal to 8K.                       |
| remote_write_cmd_proc_latency_hist_16K | remote_write_cmd_processing_latency_histogram_16K | Histogram for remote write command processing and data delivery time for I/O greater than 8K AND less than/equal to 16K.  |
| remote_write_cmd_proc_latency_hist_32K | remote_write_cmd_processing_latency_histogram_32K | Histogram for remote write command processing and data delivery time for I/O greater than 16K AND less than/equal to 32K. |
| remote_write_cmd_proc_latency_hist_64K | remote_write_cmd_processing_latency_histogram_64K | Histogram for remote write command processing and data delivery time for I/O greater than 32K AND less than/equal to 64K. |

| <b>Classic Numeric Counter</b>                   | <b>REST Counter</b>                                    | <b>Description</b>   |
|--|--|--|
| remote_write_cmd_proc_latency_histogram_large_io | remote_write_cmd_processing_latency_histogram_large_io | Histogram for remote write command processing and data delivery time for I/O greater than 64K.   |
| write_data_consume_latency_histogram_8K          | write_data_consume_latency_histogram_8K                | Histogram for write data consumption time for I/O less than/equal to 8K.   |
| write_data_consume_latency_histogram_16K         | write_data_consume_latency_histogram_16K               | Histogram for write data consumption time for I/O greater than 8K AND less than/equal to 16K.  |
| write_data_consume_latency_histogram_32K         | write_data_consume_latency_histogram_32K               | Histogram for write data consumption time for I/O greater than 16K AND less than/equal to 32K.   |
| write_data_consume_latency_histogram_64K         | write_data_consume_latency_histogram_64K               | Histogram for write data consumption time for I/O greater than 32K AND less than/equal to 64K.   |
| write_data_consume_latency_histogram_large_io    | write_data_consume_latency_histogram_large_io          | Histogram for write data consumption time for I/O greater than 64K.  |
| remote_write_data_consume_latency_histogram_8K   | remote_write_data_consume_latency_histogram_8K         | Histogram for remote write data consumption time for I/O less than/equal to 8K. It includes remote transport latency and remote WAFL latency.                      |
| remote_write_data_consume_latency_histogram_16K  | remote_write_data_consume_latency_histogram_16K        | Histogram for remote write data consumption time for I/O greater than 8K AND less than/equal to 16K. It includes remote transport latency and remote WAFL latency. |

| Classic Numeric Counter                         | REST Counter   | Description   |
|---|--|---|
| remote_write_data_consume_latency_hist_32K      | remote_write_data_consume_latency_histogram_32K      | Histogram for remote write data consumption time for I/O greater than 16K AND less than/equal to 32K. It includes remote transport latency and remote WAFL latency. |
| remote_write_data_consume_latency_hist_64K      | remote_write_data_consume_latency_histogram_64K      | Histogram for remote write data consumption time for I/O greater than 32K AND less than/equal to 64K. It includes remote transport latency and remote WAFL latency. |
| remote_write_data_consume_latency_hist_large_io | remote_write_data_consume_latency_histogram_large_io | Histogram for remote write data consumption time for I/O greater than 64K. It includes remote transport latency and remote WAFL latency.                            |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object | Alias         |
|----------------|---------------|
| nvmf_lif:port  | nvmf_lif_port |
| nvmf_lif:node  | nvmf_lif_node |

# nvmf\_rdma\_port

An NVMF RDMA port is a logical interface that connects a SVM to a physical RoCE-V2 capable device. This table collects diagnostics and performance information for an NVMe/RDMA port. The table counters can be used to debug and diagnose connectivity issues with initiators or fabric on a LIF. The table counters can be used to debug and diagnose connectivity issues with initiators or fabric at a LIF level granularity. The alias names for nvmf\_rdma\_port:port and nvmf\_rdma\_port:node are nvmf\_rdma\_port\_port and nvmf\_rdma\_port\_node.

**Classic Object:** nvmf\_rdma\_port

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {port.id}:{port_ip_addr}                    | port             | This represents the construction of the row ID field in port combo object, which is a single unique string that identifies a row. |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description  |
|------------------------|---------------|--|
| node_name              | node.name     | Name of the node hosting the RDMA logical interface. |
| node_uuid              | node.uuid     | UUID of the node hosting the RDMA logical interface  |

| Classic String Counter | REST Property | Description  |
|------------------------|---------------|--|
| instance_name          | name          | Name of the row.                                       |
| instance_uuid          | uuid          | Unique identifier (UUID) for the row.                  |
| vserver_name           | svm.name      | Name of the SVM that owns this logical interface (LIF) |
| vserver_uuid           | svm.uuid      | UUID of the SVM that owns this logical interface (LIF) |
| port_ip_addr           | port_ip_addr  | IP address for the port                                |
| port.id                | port.id       | ID for the port  |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter         | Description  |
|-------------------------|----------------------|--|
| read_ops                | read_ops             | Number of read operations  |
| write_ops               | write_ops            | Number of write operations   |
| other_ops               | other_ops            | Number of operations that are not read, write, compare or compare-and-right. |
| total_ops               | total_ops            | Total number of operations.  |
| read_data               | read_data            | Amount of data read from the storage system                                  |
| write_data              | write_data           | Amount of data written to the storage system                                 |
| total_data              | total_data           | Amount of NVMF traffic to and from the storage system                        |
| avg_read_latency        | average_read_latency | Average latency for read operations  |



| Classic Numeric Counter | REST Counter          | Description   |
|-------------------------|-----------------------|---|
| avg_write_latency       | average_write_latency | Average latency for write operations  |
| avg_other_latency       | average_other_latency | Average latency for operations other than read, write, compare or compare-and-write |
| avg_latency             | average_latency       | Average latency for NVMF operations   |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object      | Alias               |
|---------------------|---------------------|
| nvmf_rdma_port:port | nvmf_rdma_port_port |
| nvmf_rdma_port:node | nvmf_rdma_port_node |

# nvmf\_tcp\_port

An NVMe/TCP port is a logical interface that connects a SVM to a physical TCP capable device. This table collects diagnostics and performance information for an NVMe/TCP port. The table counters can be used to debug and diagnose connectivity issues with initiators or fabric on a LIF. The table counters can be used to debug and diagnose connectivity issues with initiators or fabric at a LIF level granularity. The alias names for nvmf\_tcp\_port:port and nvmf\_tcp\_port:node are nvmf\_tcp\_port\_port and nvmf\_tcp\_port\_node respectively.

**Classic Object:** nvmf\_tcp\_port

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {port.id}:{port_ip_addr}                    | port             | This represents the construction of the row ID field in port combo object, which is a single unique string that identifies a row. |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description  |
|------------------------|---------------|--|
| node_name              | node.name     | Name of the node hosting the TCP logical interface |
| node_uuid              | node.uuid     | UUID of the node hosting the TCP logical interface |

| Classic String Counter | REST Property | Description  |
|------------------------|---------------|--|
| instance_name          | name          | Name of the row.                                       |
| instance_uuid          | uuid          | Unique identifier (UUID) for the row.                  |
| vserver_name           | svm.name      | Name of the SVM that owns this logical interface (LIF) |
| vserver_uuid           | svm.uuid      | UUID of the SVM that owns this logical interface (LIF) |
| port_ip_addr           | port_ip_addr  | ID for the port  |
| port_id                | port.id       | ID for the port  |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter         | Description  |
|-------------------------|----------------------|--|
| read_ops                | read_ops             | Number of read operations  |
| write_ops               | write_ops            | Number of write operations   |
| other_ops               | other_ops            | Number of operations that are not read, write, compare or compare-and-write. |
| total_ops               | total_ops            | Total number of operations.  |
| read_data               | read_data            | Amount of data read from the storage system                                  |
| write_data              | write_data           | Amount of data written to the storage system                                 |
| total_data              | total_data           | Amount of NVMF traffic to and from the storage system                        |
| avg_read_latency        | average_read_latency | Average latency for read operations  |

| Classic Numeric Counter | REST Counter          | Description   |
|-------------------------|-----------------------|---|
| avg_write_latency       | average_write_latency | Average latency for write operations  |
| avg_other_latency       | average_other_latency | Average latency for operations other than read, write, compare or compare-and-write |
| avg_latency             | average_latency       | Average latency for NVMF operations   |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object     | Alias              |
|--------------------|--------------------|
| nvmf_tcp_port:port | nvmf_tcp_port_port |
| nvmf_tcp_port:node | nvmf_tcp_port_node |

# object\_store\_client\_op

CM table for exporting object store operation statistics. Object store operations are HTTP operations sent over TCP to manage objects on any object store.

**Classic Object:** object\_store\_client\_op

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description      |
|------------------------|---------------|------------------|
| node_name              | node.name     | System node name |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter | Description  |
|-------------------------|--------------|--|
| stats                   | stats        | Counter that indicates the number of object store operations sent, and their success and failure counts for particular handle and particular object store commands.                        |
| ops                     | ops          | Counter that indicates the number of object store operations in total for particular handle and particular object store commands. This counter acts as a base counter for average_latency. |

| Classic Numeric Counter | REST Counter     | Description  |
|-------------------------|------------------|--|
| average_latency         | average_latency  | Average latencies executed during various phases of command execution. The execution_start latency represents the average time taken to start executing an operation. The request_prepare latency represents the average time taken to prepare the complete request that needs to be sent to the server. The send latency represents the average time taken to send requests to the server. The execution_start_to_send_complete represents the average time taken to send a request since its execution started. The execution_start_to_first_byte_received latency represents the average time taken to receive the first byte of a response since the command's execution started. These counters can be used to identify performance bottlenecks within the object store client module. This counter contains average latency for a particular handle and a particular object store command. |
| throughput_bytes        | throughput_bytes | Counter that indicates the throughput for the corresponding object store command in bytes per second.  |
| throughput_ops          | throughput_ops   | Counter that indicates the throughput for commands in operations per second.   |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| <b>REST Counter</b> | <b>Description</b>  |
|---------------------|---|
| ops                 | The counter labels have changed. All of the dashes '-' in the labels were changed to underscores '_'. |
| average_latency     | The counter labels have changed. All of the dashes '-' in the labels were changed to underscores '_'. |

## Table Aliases

This section describes aliases for aggregated tables.

# object\_store\_server

The counter table reports activity for the Object Store Server. This counter table records statistics of different requests, their latency, throughput along with other nblade specific statistics. The alias name for object\_store\_server:node is object\_store\_server\_node.

**Classic Object:** object\_store\_server

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description           |
|------------------------|---------------|-----------------------|
| node_name              | node.name     | System node name      |
| instance_name          | name          | Constituent SVM Name. |
| vserver_name           | svm.name      | Name of the SVM       |
| vserver_id             | svm.id        | Unique ID of the SVM  |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>  |
|--------------------------------|--------------------------------|---|
| requests                       | requests                       | Total number of object store server requests  |
| connections                    | connections                    | Total number of object store server connections.  |
| requests_outstanding           | requests_outstanding           | Number of object store server requests in process   |
| connected_connections          | connected_connections          | Number of object store server connections currently established   |
| max_requests_outstanding       | maximum_requests_outstanding   | Maximum number of object store server requests in process at one time   |
| max_connected_connections      | maximum_connected_connections  | Maximum number of object store server connections established at one time   |
| signature_v2_reqs              | signature_v2_requests          | Total number of object store server signature V2 requests   |
| signature_v4_reqs              | signature_v4_requests          | Total number of object store server signature V4 requests   |
| chunked_upload_reqs            | chunked_upload_requests        | Total number of object store server chunked object upload requests  |
| multi_delete_reqs              | multiple_delete_requests       | Total number of object store server multiple object delete requests   |
| presigned_url_reqs             | presigned_url_requests         | Total number of presigned object store server URL requests.   |
| get_object_total               | get_object_total               | Number of GET object operations   |
| get_object_failed              | get_object_failed              | Number of failed GET object operations  |
| get_object_failed_client_close | get_object_failed_client_close | Number of times GET object operation failed due to the case where client closed the connection while the operation was still pending on server. |

| <b>Classic Numeric Counter</b>         | <b>REST Counter</b>                    | <b>Description</b>  |
|--|--|---|
| get_object_rate                        | get_object_rate                        | Number of GET object operations per second  |
| get_object_latency                     | get_object_latency                     | Average first-byte latency for GET object operations  |
| get_object_lastbyte_latency            | get_object_lastbyte_latency            | Average last-byte latency for GET object operations   |
| get_bucket_acl_total                   | get_bucket_acl_total                   | Number of GET Bucket ACL operations   |
| get_bucket_acl_failed                  | get_bucket_acl_failed                  | Number of failed GET Bucket ACL operations  |
| get_object_acl_total                   | get_object_acl_total                   | Number of GET Object ACL operations   |
| get_object_acl_failed                  | get_object_acl_failed                  | Number of failed GET Object ACL operations  |
| get_object_tagging_total               | get_object_tagging_total               | Number of GET object tagging operations   |
| get_object_tagging_failed              | get_object_tagging_failed              | Number of failed GET object tagging operations  |
| get_object_tagging_failed_client_close | get_object_tagging_failed_client_close | Number of times GET object tagging operation failed due to the case where client closed the connection while the operation was still pending on server. |
| get_object_tagging_rate                | get_object_tagging_rate                | Number of GET object tagging operations per second  |
| get_object_tagging_latency             | get_object_tagging_latency             | Average latency for GET object tagging operations   |
| delete_object_tagging_total            | delete_object_tagging_total            | Number of DELETE object tagging operations.   |
| delete_object_tagging_failed           | delete_object_tagging_failed           | Number of failed DELETE object tagging operations.  |

| <b>Classic Numeric Counter</b>            | <b>REST Counter</b>                       | <b>Description</b>  |
|---|---|---|
| delete_object_tagging_failed_client_close | delete_object_tagging_failed_client_close | Number of times DELETE object tagging operation failed because client terminated connection while the operation was still pending on server.    |
| delete_object_tagging_rate                | delete_object_tagging_rate                | Number of DELETE object tagging operations per second.  |
| delete_object_tagging_latency             | delete_object_tagging_latency             | Average latency for DELETE object tagging operations.   |
| put_object_total                          | put_object_total                          | Number of PUT object operations   |
| put_object_failed                         | put_object_failed                         | Number of failed PUT object operations  |
| put_object_failed_client_close            | put_object_failed_client_close            | Number of times PUT object operation failed due to the case where client closed the connection while the operation was still pending on server. |
| put_object_rate                           | put_object_rate                           | Number of PUT object operations per second  |
| put_object_latency                        | put_object_latency                        | Average latency for PUT object operations   |
| put_object_tagging_total                  | put_object_tagging_total                  | Number of PUT object tagging operations.  |
| put_object_tagging_failed                 | put_object_tagging_failed                 | Number of failed PUT object tagging operations.   |
| put_object_tagging_failed_client_close    | put_object_tagging_failed_client_close    | Number of times PUT object tagging operation failed because client terminated connection while the operation was still pending on server.       |
| put_object_tagging_rate                   | put_object_tagging_rate                   | Number of PUT object tagging operations per second.   |
| put_object_tagging_latency                | put_object_tagging_latency                | Average latency for PUT object tagging operations.  |

| <b>Classic Numeric Counter</b>  | <b>REST Counter</b>             | <b>Description</b>   |
|---------------------------------|---------------------------------|--|
| head_bucket_total               | head_bucket_total               | Number of HEAD bucket operations   |
| head_bucket_failed              | head_bucket_failed              | Number of failed HEAD bucket operations  |
| head_bucket_failed_client_close | head_bucket_failed_client_close | Number of times HEAD bucket operation failed due to the case where client closed the connection while the operation was still pending on server. |
| head_bucket_rate                | head_bucket_rate                | Number of HEAD bucket operations per second  |
| head_bucket_latency             | head_bucket_latency             | Average latency for HEAD bucket operations   |
| head_object_total               | head_object_total               | Number of HEAD Object operations   |
| head_object_failed              | head_object_failed              | Number of failed HEAD Object operations  |
| head_object_failed_client_close | head_object_failed_client_close | Number of times HEAD object operation failed due to the case where client closed the connection while the operation was still pending on server. |
| head_object_rate                | head_object_rate                | Number of HEAD Object operations per second  |
| head_object_latency             | head_object_latency             | Average latency for HEAD object operations   |
| delete_object_total             | delete_object_total             | Number of DELETE object operations   |
| delete_object_failed            | delete_object_failed            | Number of failed DELETE object operations  |

| <b>Classic Numeric Counter</b>           | <b>REST Counter</b>                      | <b>Description</b>  |
|--|--|---|
| delete_object_failed_client_close        | delete_object_failed_client_close        | Number of times DELETE object operation failed due to the case where client closed the connection while the operation was still pending on server.        |
| delete_object_rate                       | delete_object_rate                       | Number of DELETE object operations per second   |
| delete_object_latency                    | delete_object_latency                    | Average latency for DELETE object operations  |
| list_objects_total                       | list_objects_total                       | Number of LIST Objects operations   |
| list_objects_failed                      | list_objects_failed                      | Number of failed LIST objects operations  |
| list_objects_failed_client_close         | list_objects_failed_client_close         | Number of times LIST objects operation failed due to the case where client closed the connection while the operation was still pending on server.         |
| list_objects_rate                        | list_objects_rate                        | Number of LIST Objects operations per second  |
| list_objects_latency                     | list_objects_latency                     | Average latency for LIST Objects operations   |
| list_object_versions_total               | list_object_versions_total               | Number of LIST Object Versions operations   |
| list_object_versions_failed              | list_object_versions_failed              | Number of failed LIST object versions operations  |
| list_object_versions_failed_client_close | list_object_versions_failed_client_close | Number of times LIST object versions operation failed due to the case where client closed the connection while the operation was still pending on server. |
| list_object_versions_rate                | list_object_versions_rate                | Number of LIST Object Versions operations per second  |

| <b>Classic Numeric Counter</b>   | <b>REST Counter</b>              | <b>Description</b>   |
|----------------------------------|----------------------------------|--|
| list_object_versions_latency     | list_object_versions_latency     | Average latency for LIST Object versions operations  |
| put_bucket_versioning_total      | put_bucket_versioning_total      | Number of Put Bucket Versioning operations.  |
| put_bucket_versioning_failed     | put_bucket_versioning_failed     | Number of failed Put Bucket Versioning operations  |
| get_bucket_versioning_total      | get_bucket_versioning_total      | Number of Get Bucket Versioning operations.  |
| get_bucket_versioning_failed     | get_bucket_versioning_failed     | Number of failed Get Bucket Versioning operations  |
| list_buckets_total               | list_buckets_total               | Number of LIST Buckets operations  |
| list_buckets_failed              | list_buckets_failed              | Number of failed LIST Buckets operations   |
| list_buckets_rate                | list_buckets_rate                | Number of LIST Buckets operations per second   |
| list_buckets_latency             | list_buckets_latency             | Average latency for LIST Buckets operations  |
| list_buckets_failed_client_close | list_buckets_failed_client_close | Number of times LIST Bucket operation failed due to the case where client closed the connection while the operation was still pending on server. |
| list_uploads_total               | list_uploads_total               | Number of LIST Upload operations   |
| list_uploads_failed              | list_uploads_failed              | Number of failed LIST Upload operations  |
| list_uploads_failed_client_close | list_uploads_failed_client_close | Number of times LIST Upload operation failed due to the case where client closed the connection while the operation was still pending on server. |
| list_uploads_rate                | list_uploads_rate                | Number of LIST Upload operations per second  |

| <b>Classic Numeric Counter</b>                | <b>REST Counter</b>                           | <b>Description</b>   |
|---|---|--|
| list_uploads_latency                          | list_uploads_latency                          | Average latency for LIST Upload operations   |
| initiate_multipart_upload_total               | initiate_multipart_upload_total               | Number of Initiate Multipart Upload operations.  |
| initiate_multipart_upload_failed              | initiate_multipart_upload_failed              | Number of failed Initiate Multipart Upload operations.   |
| initiate_multipart_upload_failed_client_close | initiate_multipart_upload_failed_client_close | Number of times Initiate Multipart Upload operation failed because client terminated connection while the operation was still pending on server. |
| initiate_multipart_upload_rate                | initiate_multipart_upload_rate                | Number of Initiate Multipart Upload operations per second.   |
| initiate_multipart_upload_latency             | initiate_multipart_upload_latency             | Average latency for Initiate Multipart Upload operations.  |
| upload_part_total                             | upload_part_total                             | Number of Upload Part operations.  |
| upload_part_failed                            | upload_part_failed                            | Number of failed Upload Part operations.   |
| upload_part_failed_client_close               | upload_part_failed_client_close               | Number of times Upload Part operation failed because client terminated connection while the operation was still pending on server.               |
| upload_part_rate                              | upload_part_rate                              | Number of Upload Part operations per second.   |
| upload_part_latency                           | upload_part_latency                           | Average latency for Upload Part operations.  |
| complete_multipart_upload_total               | complete_multipart_upload_total               | Number of Complete Multipart Upload operations.  |
| complete_multipart_upload_failed              | complete_multipart_upload_failed              | Number of failed Complete Multipart Upload operations.   |

| <b>Classic Numeric Counter</b>                | <b>REST Counter</b>                           | <b>Description</b>   |
|---|---|--|
| complete_multipart_upload_failed_client_close | complete_multipart_upload_failed_client_close | Number of times Complete Multipart Upload operation failed because client terminated connection while the operation was still pending on server. |
| complete_multipart_upload_rate                | complete_multipart_upload_rate                | Number of Complete Multipart Upload operations per second.   |
| complete_multipart_upload_latency             | complete_multipart_upload_latency             | Average latency for Complete Multipart Upload operations.  |
| abort_multipart_upload_total                  | abort_multipart_upload_total                  | Number of Abort Multipart Upload operations.   |
| abort_multipart_upload_failed                 | abort_multipart_upload_failed                 | Number of failed Abort Multipart Upload operations.  |
| abort_multipart_upload_failed_client_close    | abort_multipart_upload_failed_client_close    | Number of times Abort Multipart Upload operation failed because client terminated connection while the operation was still pending on server.    |
| abort_multipart_upload_rate                   | abort_multipart_upload_rate                   | Number of Abort Multipart Upload operations per second.  |
| abort_multipart_upload_latency                | abort_multipart_upload_latency                | Average latency for Abort Multipart Upload operations.   |
| create_bucket_total                           | create_bucket_total                           | Number of Create Bucket operations.  |
| create_bucket_failed                          | create_bucket_failed                          | Number of failed Create Bucket operations.   |
| create_bucket_failed_client_close             | create_bucket_failed_client_close             | Number of times Create Bucket operation failed because client terminated connection while the operation was still pending on server.             |
| create_bucket_rate                            | create_bucket_rate                            | Number of Create Bucket operations per second.   |



| <b>Classic Numeric Counter</b>    | <b>REST Counter</b>               | <b>Description</b>   |
|-----------------------------------|-----------------------------------|--|
| create_bucket_latency             | create_bucket_latency             | Average latency for Create Bucket operations.  |
| delete_bucket_total               | delete_bucket_total               | Number of Delete Bucket operations.  |
| delete_bucket_failed              | delete_bucket_failed              | Number of failed Delete Bucket operations.   |
| delete_bucket_failed_client_close | delete_bucket_failed_client_close | Number of times Delete Bucket operation failed because client terminated connection while the operation was still pending on server. |
| delete_bucket_rate                | delete_bucket_rate                | Number of Delete Bucket operations per second.   |
| delete_bucket_latency             | delete_bucket_latency             | Average latency for Delete Bucket operations.  |
| get_data                          | get_data                          | Rate of GET object data transfers per second   |
| put_data                          | put_data                          | Rate of PUT object data transfers per second   |
| max_cmds_per_connection           | maximum_commands_per_connection   | Maximum commands pipelined at any instance on a connection.  |
| request_parse_errors              | request_parse_errors              | Number of request parser errors due to malformed requests.   |
| server_connection_close           | server_connection_close           | Number of connection closes triggered by server due to fatal errors.   |
| authentication_failures           | authentication_failures           | Number of authentication failures.   |
| input_flow_control_entry          | input_flow_control_entry          | Number of times input flow control was entered.  |
| input_flow_control_exit           | input_flow_control_exit           | Number of times input flow control was exited.   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>       | <b>Description</b>  |
|--------------------------------|---------------------------|---|
| output_flow_control_entry      | output_flow_control_entry | Number of output flow control was entered.  |
| output_flow_control_exit       | output_flow_control_exit  | Number of times output flow control was exited.                                       |
| root_user_access               | root_user_access          | Number of times access was done by root user.   |
| allow_access                   | allow_access              | Number of times access was allowed.   |
| explicit_deny_access           | explicit_deny_access      | Number of times access was denied explicitly by a policy statement.                   |
| default_deny_access            | default_deny_access       | Number of times access was denied by default and not through any policy statement.    |
| anonymous_access               | anonymous_access          | Number of times anonymous access was allowed.   |
| anonymous_deny_access          | anonymous_deny_access     | Number of times anonymous access was denied.  |
| group_policy_evaluated         | group_policy_evaluated    | Number of times group policies were evaluated.  |
| tagging                        | tagging                   | Number of requests with tagging specified.  |
| multipart_op_flexvol           | multipart_op_flexvol      | Total number of S3Adapter Multipart Ops attempted on a NAS bucket hosted on a FlexVol |
| s3a_multipart_get_part         | s3a_multipart_get_part    | Total number of S3Adapter Multipart Get Part Ops attempted on a NAS bucket            |
| css_multipart_get_part         | css_multipart_get_part    | Total number of CSS Multipart Get Part Ops attempted on a CSS bucket                  |

# Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object           | Alias                    |
|--------------------------|--------------------------|
| object_store_server:node | object_store_server_node |

# path

The Counter Manager table for exporting path performance counters.

**Classic Object:** path

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description      |
|------------------------|---------------|------------------|
| node_name              | node.name     | System node name |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter        | Description   |
|-------------------------|---------------------|---|
| debounce_count          | debounce_count      | Total number of times a device has entered into debounce period.                              |
| login_attempt_count     | login_attempt_count | Total number of PLOGI attempts to the device.   |
| read_iops               | read_iops           | The number of I/O read operations sent from the initiator port to the indicated target port.  |
| write_iops              | write_iops          | The number of I/O write operations sent from the initiator port to the indicated target port. |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>        | <b>Description</b>   |
|--------------------------------|----------------------------|--|
| total_iops                     | total_iops                 | The number of total read/write I/O operations sent from the initiator port to the indicated target port.   |
| read_latency                   | read_latency               | The average latency of I/O read operations sent from this controller to the indicated target port.   |
| write_latency                  | write_latency              | The average latency of I/O write operations sent from this controller to the indicated target port.  |
| total_latency                  | average_latency            | The average latency of I/O read and write operations sent from this controller to the indicated target port.   |
| average_read_service_time      | average_read_service_time  | The average time taken to process a read operation on the indicated target port by the indicated initiator port. This is the sum of time waiting to be transmitted and latency.          |
| average_write_service_time     | average_write_service_time | The average time taken to process a write operation on the indicated target port by the indicated initiator port. This is the sum of time waiting to be transmitted and latency.         |
| average_service_time           | average_service_time       | The average time taken to process a read or write operation on the indicated target port by the indicated initiator port. This is the sum of time waiting to be transmitted and latency. |
| read_data                      | read_data                  | The average read throughput in kilobytes per second read from the indicated target port by the controller.   |

| Classic Numeric Counter | REST Counter | Description  |
|-------------------------|--------------|--|
| write_data              | write_data   | The average write throughput in kilobytes per second written to the indicated target port by the controller.         |
| total_data              | total_data   | The average throughput in kilobytes per second read and written from/to the indicated target port by the controller. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# processor

The processor table exports performance counters for the central processing units of the system. The alias name for processor:node is processor\_node.

**Classic Object:** processor

## Table Row IDs

| ID Format       | Aggregation Type | Comment   |
|-----------------|------------------|---|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description       |
|------------------------|---------------|-------------------|
| node_name              | node.name     | Name of the node  |
| node_uuid              | node.uuid     | UUID for the node |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter | Description   |
|-------------------------|--------------|---|
| processor_busy          | busy_percent | Percentage of elapsed time that the processor is executing non-idle processes |
| processor_elapsed_time  | elapsed_time | Wall-clock time since boot used for calculating processor utilization         |
| sk_switches             | sk_switches  | Number of sk switches per second  |

| Classic Numeric Counter | REST Counter                  | Description  |
|-------------------------|-------------------------------|--|
| hard_switches           | hard_switches                 | Number of context switches per second                          |
| domain_switches         | domain_switches               | Number of domain switches per second                           |
| cp_rupts                | cp_interrupts                 | Number of CP interrupts per second                             |
| noncp_rupts             | non_cp_interrupts             | Number of non-CP interrupts per second                         |
| ipi_rupts               | ipi_interrupts                | Number of IPI interrupts per second                            |
| grab_kahuna             | grab_kahuna                   | Number of grab kahuna domain per second                        |
| grab_kahuna_time        | grab_kahuna_time_percent      | Grab kahuna domain time percentage                             |
| suspend_domain          | suspend_domain                | Number of context switches per second                          |
| suspend_domain_time     | suspend_domain_time_percent   | Suspend domain time percentage                                 |
| cp_rupt_time            | cp_interrupt_time_percent     | CP interrupt time percentage                                   |
| noncp_rupt_time         | non_cp_interrupt_time_percent | Non-CP interrupt time percentage                               |
| domain_busy             | domain_busy_percent           | Array of processor time in percentage spent in various domains |
| freebsd_cp_time         | freebsd_cp_time               | FreeBSD CP time array  |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.



| <b>REST Counter</b> | <b>Description</b>   |
|---------------------|--|
| domain_busy_percent | The counter labels have changed. The only label to change was hostOS.<br>It changed from hostOS to hostos.   |
| freebsd_cp_time     | The counter labels have changed. The old counter labels were USER, NICE, SYS, INTR, IDLE.<br>The new counter labels are user, nice, system, interrupt_request, idle. |

## Table Aliases

This section describes aliases for aggregated tables.

| <b>Classic Object</b> | <b>Alias</b>   |
|-----------------------|----------------|
| processor:node        | processor_node |

# qos

A workload represents work being done on behalf of an application or system process. The QoS Counter Manager table reports information such as operations per second, a breakdown of where read operations are going, the inter-arrival time of operation request messages, working set size information, operation latency per workload, and deferred workload information. These statistics illustrate system performance with given workloads. The alias name for qos:policy\_group is qos\_policy\_group.

**Classic Object:** workload

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                              |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {policy_group_name}:{policy_group_uuid}     | policy_group     | This represents the construction of the row ID field in policy_group combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property     | Description  |
|------------------------|-------------------|--|
| node_name              | node.name         | System node name   |
| instance_name          | name              | Name of the constituent workload.  |
| instance_uuid          | uuid              | Unique identifier (UUID) for the workload. While the workload name may change, the UUID will be invariant. |
| policy_group_name      | policy_group_name | This is the name of the policy group containing this workload.   |

| Classic String Counter | REST Property     | Description  |
|------------------------|-------------------|--|
| policy_group_uuid      | policy_group_uuid | This is the UUID of the policy group containing this workload. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter              | Description   |
|-------------------------|---------------------------|---|
| ops                     | ops                       | Workload operations executed per second.  |
| read_io_type            | read_io_type_percent      | This is the percentage of read requests served from various components (such as buffer cache, ext_cache, disk, etc.).   |
| read_io_type_base       | read_io_type_base         | This is the total number of WAFL read requests. It is used as a base counter (or denominator) for the read_io_type_percent counter, to compute the percentage of reads by various WAFL components.            |
| sequential_reads        | sequential_reads_percent  | This is the percentage of reads, performed on behalf of the workload, that were sequential.   |
| sequential_reads_base   | sequential_reads_base     | This is the total number of reads, performed on behalf of the workload. It is used as a base counter (or denominator) for the sequential_reads_percent counter to compute the percentage of sequential reads. |
| sequential_writes       | sequential_writes_percent | This is the percentage of writes, performed on behalf of the workload, that were sequential. This counter is only available on platforms with more than 4GB of NVRAM.   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>   |
|--------------------------------|------------------------|--|
| sequential_writes_base         | sequential_writes_base | This is the total number of writes, performed on behalf of the workload. This is used as a base counter (or denominator) for the sequential_writes_percent counter to compute the percentage of sequential writes. |
| latency                        | latency                | This is the average response time for requests that were initiated by the workload.  |
| total_data                     | total_data             | This is the total amount of data read/written per second from/to the filer by the workload.  |
| read_ops                       | read_ops               | This is the rate of this workload's read operations that completed during the measurement interval.  |
| write_ops                      | write_ops              | This is the workload's write operations that completed during the measurement interval; measured per second.   |
| other_ops                      | other_ops              | This is the rate of this workload's other operations that completed during the measurement interval.   |
| read_data                      | read_data              | This is the amount of data read per second from the filer by the workload.   |
| write_data                     | write_data             | This is the amount of data written per second to the filer by the workload.  |
| read_latency                   | read_latency           | This is the average response time for read requests that were initiated by the workload.   |

| Classic Numeric Counter | REST Counter  | Description   |
|-------------------------|---------------|---|
| write_latency           | write_latency | This is the average response time for write requests that were initiated by the workload. |
| concurrency             | concurrency   | This is the average number of concurrent requests for the workload.                       |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object   | Alias            |
|------------------|------------------|
| qos:policy_group | qos_policy_group |

# qos\_detail

The qos\_detail Counter Manager table that provides service center-based statistical information. Note: This table returns a very large number of instances. Querying by instance name and using wild cards may improve response times.

**Classic Object:** workload\_detail

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                      |
|------------------------|---------------|----------------------------------|
| node_name              | node.name     | System node name                 |
| resource_name          | resource.name | Name of the associated resource. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter | Description   |
|-------------------------|--------------|---|
| visits                  | visits       | The number of visits that the workload made to the service center; measured in visits per second. |
| service_time            | service_time | The workload's average service time per visit to the service center.                              |
| wait_time               | wait_time    | The workload's average wait time per visit to the service center.                                 |

| Classic Numeric Counter | REST Counter    | Description  |
|-------------------------|-----------------|--|
| in_latency_path         | in_latency_path | Determines whether or not service center-based statistics are in the latency path. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# qos\_detail\_volume

The qos\_detail\_volume Counter Manager table provides service center-based statistical information for all volumes. Service centers are resource elements that contribute to the latency of a request. This table provides information on the breakdown of a volume's response time such as the number of visits, service time and wait time across service centers. Note: This table returns a very large number of rows.

**Classic Object:** workload\_detail\_volume

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                      |
|------------------------|---------------|----------------------------------|
| node_name              | node.name     | System node name                 |
| resource_name          | resource_name | Name of the associated resource. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter | Description   |
|-------------------------|--------------|---|
| visits                  | visits       | The number of visits that the workload made to the service center; measured in visits per second. |
| service_time            | service_time | The workload's average service time per visit to the service center.                              |
| wait_time               | wait_time    | The workload's average wait time per visit to the service center.                                 |



| Classic Numeric Counter | REST Counter    | Description  |
|-------------------------|-----------------|--|
| in_latency_path         | in_latency_path | Determines whether or not service center-based statistics are in the latency path. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# qos\_resource

The resource CM table provides CPU and aggregate resource utilization information.

**Classic Object:** resource

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property   | Description                 |
|------------------------|-----------------|-----------------------------|
| node_name              | node.name       | System node name            |
| parent_resource        | parent_resource | The parent of the resource. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter  | Description   |
|-------------------------|---------------|---|
| busy_percent            | busy_percent  | The percent of the resource that is used.                               |
| num_instances           | num_instances | The number of instances (processors, disks, and so on) in the resource. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

# Table Aliases

This section describes aliases for aggregated tables.

# qos\_volume

The workload\_volume table provides workload statistics on a per volume basis. Workload information at a file or LUN level are not shown by this table (See the workload table). This table provides information such as operations per second, a breakdown of where read operations are going, the interarrival time of operation request messages, working set size information, operation latency per workload, and deferred workload information.

**Classic Object:** workload\_volume

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment  |
|---|------------------|--|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.                             |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description   |
|------------------------|---------------|---|
| node_name              | node.name     | System node name  |
| instance_name          | name          | This is the name of the constituent workload row.   |
| instance_uuid          | uuid          | This is the unique identifier for the qos_volume row. While the qos_volume name may change, the UUID will not change. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>  |
|--------------------------------|--------------------------|---|
| latency                        | latency                  | This is the average response time for requests that were initiated by the workload.   |
| ops                            | ops                      | This field is the workload's rate of operations that completed during the measurement interval; measured per second.  |
| read_data                      | read_data                | This is the amount of data read per second from the filer by the workload.  |
| read_io_type                   | read_io_type_percent     | This is the percentage of read requests served from various components (such as buffer cache, ext_cache, disk, etc.).   |
| read_io_type_base              | read_io_type_base        | This is the total number of WAFL read requests. It is used as a base counter (or denominator) for the read_io_types counter, to compute the percentage of reads by various WAFL components. |
| read_latency                   | read_latency             | This is the average response time for read requests that were initiated by the workload.  |
| read_ops                       | read_ops                 | This is the rate of this workload's read operations that completed during the measurement interval.   |
| sequential_reads               | sequential_reads_percent | This is the percentage of reads, performed on behalf of the workload, that were sequential.   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>       | <b>Description</b>   |
|--------------------------------|---------------------------|--|
| sequential_reads_base          | sequential_reads_base     | This is the total number of reads, performed on behalf of the workload. It is used as a base counter (or denominator) for the sequential_reads_percent counter to compute the percentage of sequential reads.      |
| sequential_writes_percent      | sequential_writes_percent | This is the percentage of writes, performed on behalf of the workload, that were sequential. This counter is only available on platforms with more than 4GB of NVRAM.  |
| sequential_writes_base         | sequential_writes_base    | This is the total number of writes, performed on behalf of the workload. This is used as a base counter (or denominator) for the sequential_writes_percent counter to compute the percentage of sequential writes. |
| total_data                     | total_data                | This is the total amount of data read/written per second from/to the filer by the workload.  |
| write_data                     | write_data                | This is the amount of data written per second to the filer by the workload.  |
| write_latency                  | write_latency             | This is the average response time for write requests that were initiated by the workload.  |
| write_ops                      | write_ops                 | This is the workload's write operations that completed during the measurement interval; measured per second.   |
| concurrency                    | concurrency               | This is the average number of concurrent requests for the workload.  |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# qtree

CM table for exporting qtree performance counters

Classic Object: qtree

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment  |
|---|------------------|--|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.                             |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property                  | Description   |
|------------------------|--------------------------------|---|
| node_name              | node.name                      | System node name  |
| vserver_name           | svm.name                       | SVM name  |
| parent_vol             | parent_volume.name             | Name of the parent volume   |
| constituent_parent_vol | parent_constituent_volume.name | Name of the FlexGroup constituent or FlexVol that handled the Ops |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter | Description                                       |
|-------------------------|--------------|---|
| nfs_ops                 | nfs_ops      | Number of NFS operations per second to the qtree  |
| cifs_ops                | cifs_ops     | Number of CIFS operations per second to the qtree |



| Classic Numeric Counter | REST Counter | Description   |
|-------------------------|--------------|---|
| css_ops                 | css_ops      | Number of CSS operations per second to the qtree  |
| internal_ops            | internal_ops | Number of internal operations generated by activities such as snapmirror and backup per second to the qtree |
| total_ops               | total_ops    | Summation of NFS ops, CIFS ops, CSS ops and internal ops  |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# raid

CM object for exporting raid performance counters

**Classic Object:** raid

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description          |
|------------------------|---------------|----------------------|
| node_name              | node.name     | The name of the node |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter               | Description   |
|-------------------------|----------------------------|---|
| tetris_written          | tetris_write_rate          | Tetris writes per second                                  |
| partial_stripes         | partial_stripes_write_rate | Partial stripes written per second                        |
| full_stripes            | full_stripes_write_rate    | Full stripes written per second                           |
| blocks_written          | block_write_rate           | Blocks written per second                                 |
| blocks_read             | block_read_rate            | Number of blocks read per second while processing writes  |
| raid_read_ops           | hdd_operations_read_rate   | Read operations per second issued by WAFL to RAID on HDD. |

| Classic Numeric Counter | REST Counter             | Description  |
|-------------------------|--------------------------|--|
| raid_read_latency       | hdd_read_latency_average | Average latency for all reads operations sent by WAFL to RAID on HDD in microseconds |
| raid_ssd_read_ops       | ssd_operations_read_rate | Read operations per second issued by WAFL to RAID on SSD                             |
| raid_ssd_read_latency   | ssd_read_latency_average | Average latency for all SSD reads operations sent by WAFL to RAID in microseconds    |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# rewind\_context

These counters are used to monitor the performance of the rewind context manager in the nblade.

**Classic Object:** rw\_ctx

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                 |
|------------------------|---------------|-----------------------------|
| node_name              | node.name     | Node Name.                  |
| node_uuid              | node.uuid     | Node UUID.                  |
| instance_name          | name          | Name of the rewind context. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter | Description  |
|-------------------------|--------------|--|
| nfs_rewinds             | nfs_rewinds  | Array of number of rewinds for NFS ops based on their reasons.   |
| nfs_giveups             | nfs_give_ups | Array of number of give-ups of NFS ops because they rewind more than a certain threshold, categorized by their rewind reasons. |
| cifs_rewinds            | cifs_rewinds | Array of number of rewinds for CIFS ops based on their reasons.  |

| Classic Numeric Counter | REST Counter  | Description   |
|-------------------------|---------------|---|
| cifs_giveups            | cifs_give_ups | Array of number of give-ups of CIFS ops because they rewind more than a certain threshold, categorized by their rewind reasons. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter  | Description   |
|---------------|---|
| nfs_rewinds   | Most array labels are kept the same. In general, labels containing abbreviations are expanded. Two specific labels that are changed more than the general change are<br>1. "FC Close DOC" is changed to "FlexCache Deletion On Last Close".<br>2. "Dpo msg failed" is changed to "Insufficient Memory for RPC-over-RDMA Buffer Allocation". |
| nfs_give_ups  | Most array labels are kept the same. In general, labels containing abbreviations are expanded. Two specific labels that are changed more than the general change are<br>1. "FC Close DOC" is changed to "FlexCache Deletion On Last Close".<br>2. "Dpo msg failed" is changed to "Insufficient Memory for RPC-over-RDMA Buffer Allocation". |
| cifs_rewinds  | Most array labels are kept the same. In general, labels containing abbreviations are expanded.  |
| cifs_give_ups | Most array labels are kept the same. In general, labels containing abbreviations are expanded.  |

## Table Aliases

This section describes aliases for aggregated tables.

# smb2

These counters report SVM specific activity from SMB2/SMB3 revision of the protocol. The alias names for smb2:node and smb2:svm are smb2\_node and smb2\_svm respectively.

**Classic Object:** smb2

## Table Row IDs

| ID Format                       | Aggregation Type | Comment   |
|---------------------------------|------------------|---|
| {instance_name}:{instance_uuid} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {svm.name}:{svm.id}             | svm              | This represents the construction of the row ID field in svm combo object, which is a single unique string that identifies a row.  |
| {node.name}:{node.uuid}         | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description           |
|------------------------|---------------|-----------------------|
| node_name              | node.name     | Node name             |
| vserver_name           | svm.name      | Name of the SVM       |
| vserver_id             | svm.id        | Unique ID for the SVM |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>                                     |
|--------------------------------|--------------------------------|--|
| close_latency                  | close_latency                  | Average latency for SMB2_COM_CLOSE operations          |
| close_latency_histogram        | close_latency_histogram        | Latency histogram for SMB2_COM_CLOSE operations        |
| close_ops                      | close_ops                      | Number of SMB2_COM_CLOSE operations                    |
| create_latency                 | create_latency                 | Average latency for SMB2_COM_CREATE operations         |
| create_latency_histogram       | create_latency_histogram       | Latency histogram for SMB2_COM_CREATE operations       |
| create_ops                     | create_ops                     | Number of SMB2_COM_CREATE operations                   |
| lock_latency                   | lock_latency                   | Average latency for SMB2_COM_LOCK operations           |
| lock_latency_histogram         | lock_latency_histogram         | Latency histogram for SMB2_COM_LOCK operations         |
| lock_ops                       | lock_ops                       | Number of SMB2_COM_LOCK operations                     |
| negotiate_ops                  | negotiate_ops                  | Number of SMB2_COM_NEGOTIATE operations                |
| negotiate_latency              | negotiate_latency              | Average latency for SMB2_COM_NEGOTIATE operations      |
| negotiate_latency_histogram    | negotiate_latency_histogram    | Latency histogram for SMB2_COM_NEGOTIATE operations    |
| oplock_break_latency           | oplock_break_latency           | Average latency for SMB2_COM_OPLOCK_BREAK operations   |
| oplock_break_latency_histogram | oplock_break_latency_histogram | Latency histogram for SMB2_COM_OPLOCK_BREAK operations |

| <b>Classic Numeric Counter</b>    | <b>REST Counter</b>               | <b>Description</b>  |
|-----------------------------------|-----------------------------------|---|
| oplock_break_ops                  | oplock_break_ops                  | Number of SMB2_COM_OPLOCK_BREAK operations                |
| query_directory_latency           | query_directory_latency           | Average latency for SMB2_COM_QUERY_DIRECTORY operations   |
| query_directory_latency_histogram | query_directory_latency_histogram | Latency histogram for SMB2_COM_QUERY_DIRECTORY operations |
| query_directory_ops               | query_directory_ops               | Number of SMB2_COM_QUERY_DIRECTORY operations             |
| query_info_latency                | query_info_latency                | Average latency for SMB2_COM_QUERY_INFO operations        |
| query_info_latency_histogram      | query_info_latency_histogram      | Latency histogram for SMB2_COM_QUERY_INFO operations      |
| query_info_ops                    | query_info_ops                    | Number of SMB2_COM_QUERY_INFO operations                  |
| read_ops                          | read_ops                          | Number of SMB2_COM_READ operations                        |
| read_latency                      | read_latency                      | Average latency for SMB2_COM_READ operations              |
| read_latency_histogram            | read_latency_histogram            | Latency histogram for SMB2_COM_READ operations            |
| session_setup_latency             | session_setup_latency             | Average latency for SMB2_COM_SESSION_SETUP operations     |
| session_setup_latency_histogram   | session_setup_latency_histogram   | Latency histogram for SMB2_COM_SESSION_SETUP operations   |



| Classic Numeric Counter        | REST Counter                   | Description  |
|--------------------------------|--------------------------------|--|
| session_setup_ops              | session_setup_ops              | Number of SMB2_COM_SESSION_SETUP operations            |
| set_info_latency               | set_info_latency               | Average latency for SMB2_COM_SET_INFO operations       |
| set_info_latency_histogram     | set_info_latency_histogram     | Latency histogram for SMB2_COM_SET_INFO operations     |
| set_info_ops                   | set_info_ops                   | Number of SMB2_COM_SET_INFO operations                 |
| tree_connect_ops               | tree_connect_ops               | Number of SMB2_COM_TREE_CONNECT operations             |
| tree_connect_latency           | tree_connect_latency           | Average latency for SMB2_COM_TREE_CONNECT operations   |
| tree_connect_latency_histogram | tree_connect_latency_histogram | Latency histogram for SMB2_COM_TREE_CONNECT operations |
| write_ops                      | write_ops                      | Number of SMB2_COM_WRITE operations                    |
| write_latency                  | write_latency                  | Average latency for SMB2_COM_WRITE operations          |
| write_latency_histogram        | write_latency_histogram        | Latency histogram for SMB2_COM_WRITE operations        |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| <b>Classic Object</b> | <b>Alias</b> |
|-----------------------|--------------|
| smb2:svm              | smb2_svm     |
| smb2:node             | smb2_node    |

# svm\_cifs

The svm\_cifs table reports activity of the Common Internet File System Protocol subsystem. This is the Microsoft file-sharing protocol that evolved from the Server Message Block(SMB) application layer network protocol to connect PCs to Network Attached Storage(NAS) devices. This table supports SVM specific activity for SMB, SMB2, and SMB3 revisions of the CIFS protocol. The alias name for svm\_cifs:node is svm\_cifs\_node.

**Classic Object:** cifs

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description               |
|------------------------|---------------|---------------------------|
| node_name              | node.name     | Node name                 |
| node_uuid              | node.uuid     | Node UUID                 |
| instance_name          | name          | Name of the CIFS instance |
| vserver_name           | svm.name      | SVM name                  |
| vserver_id             | svm.id        | SVM ID                    |

# Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter            | Description  |
|-------------------------|-------------------------|--|
| cifs_ops                | total_ops               | Total number of CIFS operations  |
| cifs_op_count           | op_count                | Array of select CIFS operation counts  |
| cifs_op_pct             | op_percent              | Array of select CIFS operation counts as a percentage of total CIFS operations                   |
| cifs_latency            | latency                 | Average latency for CIFS operations  |
| cifs_latency_base       | latency_base            | Total observed CIFS operations to be used as a base counter for CIFS average latency calculation |
| cifs_latency_hist       | cifs_latency_histogram  | Histogram of latency for CIFS operations   |
| cifs_read_ops           | total_read_ops          | Total number of CIFS read operations   |
| cifs_write_ops          | total_write_ops         | Total number of CIFS write operations  |
| cifs_read_latency       | average_read_latency    | Average latency for CIFS read operations   |
| cifs_write_latency      | average_write_latency   | Average latency for CIFS write operations  |
| cifs_read_latency_hist  | read_latency_histogram  | Histogram of latency for CIFS read operations  |
| cifs_write_latency_hist | write_latency_histogram | Histogram of latency for CIFS write operations   |
| cifs_read_size_histo    | read_size_histogram     | Histogram of CIFS read sizes   |
| cifs_write_size_histo   | write_size_histogram    | Histogram of CIFS write sizes  |

| <b>Classic Numeric Counter</b>       | <b>REST Counter</b>                  | <b>Description</b>  |
|--------------------------------------|--------------------------------------|---|
| read_data                            | read_data                            | Rate of CIFS read data transfers per second.                            |
| write_data                           | write_data                           | Rate of CIFS write data transfers per second.                           |
| total_data                           | total_data                           | Rate of total CIFS data transfers per second.                           |
| commands_outstanding                 | commands_outstanding                 | Number of SMB and SMB2 commands in process                              |
| change_notifications_outstanding     | change_notifications_outstanding     | Number of active change notifications over SMB and SMB2                 |
| connections                          | connections                          | Number of connections   |
| outstanding_auth_requests            | outstanding_auth_requests            | Number of outstanding authentication requests                           |
| established_sessions                 | established_sessions                 | Number of established SMB and SMB2 sessions                             |
| signed_sessions                      | signed_sessions                      | Number of signed SMB and SMB2 sessions.                                 |
| connected_shares                     | connected_shares                     | Number of SMB and SMB2 share connections                                |
| open_files                           | open_files                           | Number of open files over SMB and SMB2                                  |
| active_searches                      | active_searches                      | Number of active searches over SMB and SMB2                             |
| max_commands_outstanding             | max_commands_outstanding             | Maximum number of SMB and SMB2 commands in process at one time          |
| max_change_notifications_outstanding | max_change_notifications_outstanding | Maximum number of SMB and SMB2 change notifications pending at one time |
| max_connections                      | max_connections                      | Maximum number of connections achieved                                  |

| <b>Classic Numeric Counter</b>    | <b>REST Counter</b>                   | <b>Description</b>   |
|-----------------------------------|---------------------------------------|--|
| max_outstanding_auth_requests     | max_outstanding_auth_requests         | Maximum number of outstanding authentication requests  |
| max_established_sessions          | max_established_sessions              | Maximum number of established SMB and SMB2 sessions achieved   |
| max_connected_shares              | max_connected_shares                  | Maximum number of SMB and SMB2 share connections achieved  |
| max_open_files                    | max_open_files                        | Maximum number of open files over SMB and SMB2 achieved  |
| max_active_searches               | max_active_searches                   | Maximum number of active searches over SMB and SMB2 achieved   |
| max_sessions_per_connection       | max_sessions_per_connection           | Maximum number of SMB or SMB2 sessions multiplexed over a single connection  |
| max_shares_per_session            | max_shares_per_session                | Maximum number of share connections achieved within a single SMB or SMB2 session   |
| max_open_files_per_share          | max_open_files_per_share              | Maximum number of open files achieved over a single share connection within a single SMB or SMB2 session   |
| handle_lease_ignored              | handle_lease_ignored                  | Number of times a request for a handle lease was ignored because too many had been granted to a single session.  |
| max_searches_per_session          | max_searches_per_session              | Maximum number of directory searches achieved over a single SMB or SMB2 session  |
| max_same_tree_connect_per_session | max_same_tree_connections_per_session | Maximum number of CIFS tree connects achieved over the same share within a single CIFS session. This counter will be set only when shares within a single CIFS session exceeds the max_same_tree_connections_per_session configured value. |

| Classic Numeric Counter              | REST Counter                          | Description   |
|--------------------------------------|---------------------------------------|---|
| max_same_user_session_per_connection | max_same_user_sessions_per_connection | Maximum number of CIFS sessions achieved by the same user within a single TCP connection. This counter will be set only when total number of session on a CIFS connection exceeds the max_same_user_sessions_per_connection configured value. |
| max_opens_same_file_per_tree         | max_opens_same_file_per_tree          | Maximum number of existing opens achieved on the same file within a single CIFS tree. This counter will be set only when number of opens on a CIFS tree exceeds the max_opens_same_file_per_tree configured value.                            |
| max_watches_set_per_tree             | max_watches_set_per_tree              | Maximum number of watches, also known as 'change notifies,' achieved within a single CIFS tree. This counter will be set only when number of watches on a CIFS tree exceeds the max_watches_set_per_tree configured value.                    |
| auth_reject_too_many                 | auth_reject_too_many                  | Authentication refused after too many requests were made in rapid succession  |
| open_reject_too_many                 | open_reject_too_many                  | Open refused after too many requests were made by the same user to the same file  |
| session_reject_too_many              | session_reject_too_many               | Session refused after too many session requests for the same user for the same client   |
| watch_reject_too_many                | watch_reject_too_many                 | Watch request, also known as 'change notify,' refused after too many watch requests   |

| <b>Classic Numeric Counter</b>       | <b>REST Counter</b>                         | <b>Description</b>   |
|--------------------------------------|---|--|
| tree_connect_reject_too_many         | tree_connect_reject_too_many                | Tree connect request rejected after too many connects to same share on session   |
| duplicate_session_disconnected       | duplicate_session_disconnected              | Detected and terminated a duplicate session  |
| connection_idle_close                | connection_idle_close                       | Detected and terminated an idle connection   |
| session_idle_close                   | session_idle_close                          | Detected and terminated an idle session  |
| conn_could_not_close_without_waiting | connections_could_not_close_without_waiting | Number of connections could not be closed right away   |
| conn_closed_internally_unexpected    | connections_closed_internally_unexpected    | Number of connections closed unexpectedly because of internal error  |
| conn_hung_cnt                        | hung_connections_count                      | Number of connections hung while closing   |
| non_unicode_client_rejected          | non_unicode_client_rejected                 | Rejected a connection attempt from a client that does not support Unicode  |
| node_referral_local                  | node_referral.local                         | Number of clients that connected using a LIF hosted by the same node that hosts the share root. Normally local access provides optimal performance.                |
| node_referral_remote                 | node_referral.remote                        | Number of clients that connected using a LIF hosted by a node different from the node that hosts the share root. Remote access may result in degraded performance. |



| <b>Classic Numeric Counter</b>     | <b>REST Counter</b>                | <b>Description</b>  |
|------------------------------------|------------------------------------|---|
| node_referral_issued               | node_referral.issued               | Number of clients that were issued a referral to the share root's node after the client connected using a LIF hosted by a node different from the share root's node.  |
| node_referral_not_possible         | node_referral.not_possible         | Number of clients that were not issued a referral to the node hosting the share root after connecting using a LIF hosted by a node different from the share root's node. No active data LIF for the share root's node was found |
| reconnection_requests_total        | reconnection_requests.total        | Number of clients who issued a CIFS reconnect request   |
| reconnection_requests_failed       | reconnection_requests.failed       | Number of CIFS reconnect requests which failed  |
| persistent_opens                   | persistent_opens                   | Number of CIFS persistent open requests made by clients   |
| durable_opens                      | durable_opens                      | Number of CIFS durable open requests made by clients  |
| optimized_smb2_opens               | optimized_smb2_opens               | Number of SMB2 open requests which used SPINNP_LOOKUP instead of SPINNP_OPEN  |
| lock_reconstruction                | lock_reconstruction                | Number of times lock reconstruction status has been encountered while processing CIFS requests from clients   |
| server_side_close_conn             | server_side_close_connection       | Number of times a decision has been made to initiate server-side closure of a client connection   |
| continuously_available_connections | continuously_available_connections | Number of connections from CA-capable clients to continuously available shares  |

| <b>Classic Numeric Counter</b>          | <b>REST Counter</b>                      | <b>Description</b>  |
|---|--|---|
| privileged_lock_test_req                | privileged_lock.test_requests            | Number of Lock Test requests from the client                              |
| privileged_lock_req                     | privileged_lock.requests                 | Number of Lock Requests from the client                                   |
| nbt_session_requests                    | nbt_session.requests                     | Number of NetBIOS session requests  |
| nbt_session_keepalives                  | nbt_session.keep_alives                  | Number of NetBIOS session keep-alives                                     |
| nameserver_query_requests               | name_server.query_requests               | Number of incoming nameserver query requests                              |
| nameserver_query_request_matches        | name_server.query_request_matches        | Number of incoming nameserver query requests that matched                 |
| nameserver_registration_requests        | name_server.registration_requests        | Number of incoming nameserver registration requests                       |
| nameserver_registration_request_matches | name_server.registration_request_matches | Number of incoming nameserver registration requests that matched          |
| path_based_ops                          | path_based_ops                           | Number of SMB and SMB2 path-based commands                                |
| avg_directory_depth                     | average_directory_depth                  | Average number of directories crossed by SMB and SMB2 path-based commands |
| max_directory_depth                     | max_directory_depth                      | Maximum number of directories crossed by SMB and SMB2 path-based commands |
| avg_junction_depth                      | average_junction_depth                   | Average number of junctions crossed by SMB and SMB2 path-based commands   |
| max_junction_depth                      | max_junction_depth                       | Maximum number of junctions crossed by SMB and SMB2 path-based commands   |
| path_cache_entries                      | path_cache.entries                       | Number of entries in the path cache                                       |

| <b>Classic Numeric Counter</b>    | <b>REST Counter</b>                   | <b>Description</b>  |
|-----------------------------------|---------------------------------------|---|
| path_cache_max_entries            | path_cache.max_entries                | Maximum number of entries in the path cache                   |
| path_cache_requests               | path_cache.requests                   | Number of path cache requests                                 |
| path_cache_hits                   | path_cache.hits                       | Number of path cache hits                                     |
| path_cache_misses                 | path_cache.misses                     | Number of path cache misses                                   |
| path_cache_latency                | path_cache.average_latency            | Average latency for a path cache request (hit or miss)        |
| path_cache_hit_latency            | path_cache.average_hit_latency        | Average latency for a path cache hit                          |
| path_cache_hit_latency_histogram  | path_cache.hit_latency_histogram      | Latency histogram for path cache hits                         |
| path_cache_miss_latency           | path_cache.average_miss_latency       | Average latency for a path cache miss                         |
| path_cache_miss_latency_histogram | path_cache.miss_latency_histogram     | Latency histogram for path cache misses                       |
| file_handle_cache_entries         | file_handle_cache.entries             | Number of entries in the file handle cache                    |
| file_handle_cache_max_entries     | file_handle.cache_max_entries         | Maximum number of entries in the file handle cache            |
| file_handle_cache_requests        | file_handle_cache.requests            | Number of file handle cache requests                          |
| file_handle_cache_hits            | file_handle_cache.hits                | Number of file handle cache hits                              |
| file_handle_cache_misses          | file_handle_cache.misses              | Number of file handle cache misses                            |
| file_handle_cache_latency         | file_handle_cache.average_latency     | Average latency for a file handle cache request (hit or miss) |
| file_handle_cache_hit_latency     | file_handle_cache.average_hit_latency | Average latency for a file handle cache hit                   |

| <b>Classic Numeric Counter</b>           | <b>REST Counter</b>                      | <b>Description</b>  |
|--|--|---|
| file_handle_cache_hit_latency_histogram  | file_handle_cache.hit_latency_histogram  | Latency histogram for file handle cache hits  |
| file_handle_cache_miss_latency           | file_handle_cache.average_miss_latency   | Average latency for a file handle cache miss  |
| file_handle_cache_miss_latency_histogram | file_handle_cache.miss_latency_histogram | Latency histogram for file handle cache misses  |
| flow_control_connections                 | flow_control.connections                 | Total number of connections that have been placed into flow control   |
| flow_control_back_to_back                | flow_control.back_to_back                | Total number of connections that have been released from flow control only to be put back into flow control before they get a chance to run |
| flow_control_max_queue_depth             | flow_control.max_queue_depth             | Maximum queue depth for flow control connection queues  |
| flow_control_latency_hist                | flow_control.latency_histogram           | Histogram of flow control caused latency for CIFS connections   |
| flexgroup_lookup_redrive                 | flexgroup_lookup.redrive                 | Total number of times a CIFS lookup has been redirected by the FlexGroup layer  |
| flexgroup_open_redrive                   | flexgroup_open.redrive                   | Total number of times a CIFS file or directory open or creation has been redirected by the FlexGroup layer                                  |
| flexgroup_lookup_multiple_redrive        | flexgroup_lookup.multiple_redrive        | Total number of times a single CIFS lookup operation has been redirected multiple times in a row  |
| flexgroup_open_multiple_redrive          | flexgroup_open.multiple_redrive          | Total number of times a single CIFS file or directory open or creation operation has been redirected multiple times in a row                |
| flexgroup_msid_cache_hit                 | flexgroup_msid_cache.hits                | Total number of times we have found a FlexGroup target MSID in the CIFS MSID path cache   |

| Classic Numeric Counter                     | REST Counter                                | Description  |
|---|---|--|
| flexgroup_msid_cache_max_entries            | flexgroup_msid_cache.max_entries            | Maximum number of entries stored in any CIFS MSID path cache   |
| flexgroup_msid_cache_max_depth              | flexgroup_msid_cache.max_depth              | Maximum path depth stored in any CIFS MSID path cache  |
| component_cache                             | component_cache                             | Array of component cache stats   |
| branchcache_hash_fetch_fail                 | branch_cache.hash_fetch_fail                | Total number of times a request to fetch hash data failed. These are failures when attempting to read existing hash data. It does not include attempts to fetch hash data that has not yet been generated. |
| branchcache_missing_hash_bytes              | branch_cache.missing_hash_bytes             | Total number of bytes of data that had to be read by the client because the hash for that content was not available on the server.   |
| branchcache_hash_fetch_ok                   | branch_cache.hash_fetch_ok                  | Total number of times a request to fetch hash data succeeded.  |
| branchcache_hash_sent_bytes                 | branch_cache.hash_sent_bytes                | Total number of bytes sent to clients requesting hashes.   |
| copyoffload_directcopy_read_request         | copyoffload_directcopy_read_request         | Total number of Copy Offload Direct-Copy Read requests.  |
| ioctl_fsctl_set_zero_data_unaligned_request | ioctl_fsctl_set_zero_data_unaligned_request | Total number of FSCTL_SET_ZERO_DATA requests not aligned to block-boundaries.  |
| sd_max_ace_count                            | security_descriptor.max_ace_count           | Maximum number of ACEs received in a security descriptor.  |
| sd_max_ace_size                             | security_descriptor.max_ace_size            | Maximum size of ACEs received in a security descriptor.  |
| cred_max_user_claims                        | credential.max_user_claims                  | Maximum size of user claims in CIFS credentials  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>             | <b>Description</b>  |
|--------------------------------|---------------------------------|---|
| cred_max_device_claims         | credential.max_device_claims    | Maximum size of device claims in CIFS credentials                   |
| cred_max_device_groups         | credential.max_device_groups    | Maximum number of groups a device is a member of.                   |
| cred_build_req                 | credential.build_requests       | Number of SpinNp Credential Build Requests for CIFS.                |
| cred_avg_win_groups            | credential.average_win_groups   | Average number of windows groups any user is a member of.           |
| cred_max_win_groups            | credential.max_win_groups       | Maximum number of windows groups any user is a member of.           |
| cred_avg_unix_groups           | credential.average_unix_groups  | Average number of UNIX groups any user is a member of.              |
| cred_max_unix_groups           | credential.max_unix_groups      | Maximum number of UNIX groups any user is a member of.              |
| cred_avg_total_groups          | credential.average_total_groups | Average number of total groups any user is a member of.             |
| cred_max_total_groups          | credential.max_total_groups     | Maximum number of total groups any user is a member of.             |
| cred_max_size_bytes            | credential.max_size             | Maximum size (in bytes) of the SpinNp credential for any CIFS user. |
| cred_avg_size_bytes            | credential.average_size         | Average size (in bytes) of the SpinNp credential for any CIFS user. |
| smb1_connections_count         | smb1_connections_count          | Current number of SMB1 connections.                                 |
| total_smb1_connections_count   | total_smb1_connections_count    | Total number of SMB1 connections.                                   |
| smb2_connections_count         | smb2_connections_count          | Current number of SMB2.0 connections.                               |
| encrypted_sessions             | encrypted_sessions              | Number of encrypted SMB3 sessions.                                  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>  |
|--------------------------------|--------------------------------|---|
| encrypted_share_connections    | encrypted_share_connections    | Number of encrypted share connections. This is the number of encrypted shares on which a tree connect has happened. |
| rejected_unencrypted_sessions  | rejected_unencrypted_sessions  | Number of session setups rejected due to the lack of client encryption capability.                                  |
| rejected_unencrypted_shares    | rejected_unencrypted_shares    | Number of share mappings rejected due to the lack of client encryption capability.                                  |
| total_smb2_connections_count   | total_smb2_connections_count   | Total number of SMB2.0 connections.   |
| smb2_1_connections_count       | smb2_1_connections_count       | Current number of SMB2.1 connections.   |
| total_smb2_1_connections_count | total_smb2_1_connections_count | Total number of SMB2.1 connections.   |
| smb3_connections_count         | smb3_connections_count         | Current number of SMB3 connections.   |
| total_smb3_connections_count   | total_smb3_connections_count   | Total number of SMB3 connections.   |
| smb3_1_connections_count       | smb3_1_connections_count       | Current number of SMB3.1 connections.   |
| total_smb3_1_connections_count | total_smb3_1_connections_count | Total number of SMB3.1 connections.   |
| extended_dfs_referral_reqs     | extended_dfs_referral_reqs     | Total number of extended DFS referrals (FSCTL_DFS_GET_REFERRALS_EX requests).                                       |
| no_version_negotiated          | no_version_negotiated_requests | Total number of negotiate requests that were dropped because no mutually supported SMB version is found.            |

| Classic Numeric Counter  | REST Counter                      | Description   |
|--------------------------|-----------------------------------|---|
| widelink_request         | widelink_requests                 | Total number of widelink encountered while resolving symlinks.    |
| homedir_share_request    | homedir_share_requests            | Total number of Tree Connect requests for a home directory share. |
| export_policy_request    | export_policy_requests            | Total number of requests using an export policy.                  |
| access_based_enumeration | access_based_enumeration_requests | Total number of requests using Access Based Enumeration.          |
| optimized_smb2_open      | optimized_smb2_open_requests      | Total number of optimized SMB2 open requests.                     |
| null_user_session        | null_user_sessions                | Total number of NULL or Anonymous user sessions.                  |
| guest_user_session       | guest_user_sessions               | Total number of Guest user sessions.                              |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter    | Description  |
|-----------------|--|
| op_count        | For each of the new labels in this array, the values have been changed to be lower case and any other non-alphanumeric characters have been replaced by an underscore. |
| op_percent      | For each of the new labels in this array, the values have been changed to be lower case and any other non-alphanumeric characters have been replaced by an underscore. |
| component_cache | For each of the new labels in this array, the values have been changed to be lower case and white spaces have been replaced by underscores.                            |



| <b>REST Counter</b>               | <b>Description</b>  |
|-----------------------------------|---|
| security_descriptor.max_ace_count | For each of the new labels in this array, the values have been changed to be lower case and white spaces have been replaced by underscores. |
| security_descriptor.max_ace_size  | For each of the new labels in this array, the values have been changed to be lower case and white spaces have been replaced by underscores. |

## Table Aliases

This section describes aliases for aggregated tables.

| <b>Classic Object</b> | <b>Alias</b>  |
|-----------------------|---------------|
| svm_cifs:node         | svm_cifs_node |

# svm\_nfs\_v3

The svm\_nfs\_v3 table reports activity for the Network File System protocol, version 3. This is the Sun file-sharing protocol that is predominant on UNIX platforms, used to connect to Network Attached Storage (NAS). The alias name for svm\_nfs\_v3:node is svm\_nfs\_v3\_node.

**Classic Object:** nfsv3

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description          |
|------------------------|---------------|----------------------|
| node_name              | node.name     | Name of the node     |
| instance_name          | name          | Constituent SVM name |
| vserver_id             | svm.id        | Vserver ID           |
| vserver_name           | svm.name      | Name of SVM          |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>  | <b>Description</b>  |
|--------------------------------|----------------------|---|
| null_success                   | null.success         | Number of successful Null procedure requests. A null procedure does nothing. It is used for testing and timing.   |
| null_error                     | null.error           | Number of erroneous Null procedure requests. A null procedure does nothing. It is used for testing and timing.  |
| null_total                     | null.total           | Total number of Null procedure requests. It is the total of null success and null error requests.   |
| null_percent                   | null.percent         | Percentage of Null procedure requests.  |
| null_avg_latency               | null.average_latency | Average latency of Null procedure requests.   |
| getattr_success                | getattr.success      | Number of successful GetAttr procedure requests. The GetAttr procedure retrieves the attributes for a specified file system object. Successful returns have a NFS3_OK status. |
| getattr_error                  | getattr.error        | Number of erroneous GetAttr procedure requests. The GetAttr procedure retrieves the attributes for a specified file system object.  |
| getattr_total                  | getattr.total        | Total number of Getattr procedure requests. It is the total number of getattr success and getattr error requests.   |
| getattr_percent                | getattr.percent      | Percentage of GetAttr procedure requests. This counter shows how often GetAttr requests happen out of all NFSv3 requests.   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>  |
|--------------------------------|-------------------------|---|
| getattr_avg_latency            | getattr.average_latency | Average latency of GetAttr procedure requests. This counter keeps track of the average response time of GetAttr requests.   |
| setattr_success                | setattr.success         | Number of successful SetAttr procedure requests. The SetAttr procedure changes one or more of the attributes of a file system object on the server.                                 |
| setattr_error                  | setattr.error           | Number of erroneous SetAttr procedure requests. The SetAttr procedure changes one or more of the attributes of a file system object on the server.                                  |
| setattr_total                  | setattr.total           | Total number of Setattr procedure requests. It is the total number of Setattr success and setattr error requests.   |
| setattr_percent                | setattr.percent         | Percentage of Setattr procedure requests. The counter shows how often SetAttr requests happen out of all NFSv3 requests.  |
| setattr_avg_latency            | setattr.average_latency | Average latency of SetAttr procedure requests. The counter keeps track of the average response time of SetAttr requests.  |
| lookup_success                 | lookup.success          | Number of successful LookUp procedure requests. The LookUp procedure searches a directory for a specific name and returns the file handle for the corresponding file system object. |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>   |
|--------------------------------|------------------------|--|
| lookup_error                   | lookup.error           | Number of erroneous LookUp procedure requests. The LookUp procedure searches a directory for a specific name and returns the file handle for the corresponding file system object. |
| lookup_total                   | lookup.total           | Total number of Lookup procedure requests. It is the total number of lookup success and lookup error requests.   |
| lookup_percent                 | lookup.percent         | Percentage of LookUp procedure requests. This counter shows how often LookUp requests happen out of all NFSv3 requests.  |
| lookup_avg_latency             | lookup.average_latency | Average latency of LookUp procedure requests. This shows the average time it takes for the LookUp operation to reply to the request.   |
| access_success                 | access.success         | Number of successful Access procedure requests. The Access procedure determines the access rights that a user has with respect to a file system object.                            |
| access_error                   | access.error           | Number of erroneous Access procedure requests. The Access procedure determines the access rights that a user has with respect to a file system object.                             |
| access_total                   | access.total           | Total number of Access procedure requests. It is the total number of access success and access error requests.   |
| access_percent                 | access.percent         | Percentage of Access procedure requests. This counter shows how often Access requests happen out of all NFSv3 requests.  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>          | <b>Description</b>  |
|--------------------------------|------------------------------|---|
| access_avg_latency             | access.average_latency       | Average latency of Access procedure requests. The counter keeps track of the average response time of Access requests.  |
| read_symlink_success           | read_symlink.success         | Number of successful ReadSymlink procedure requests. The ReadSymLink procedure reads the data associated with a symbolic link. A symbolic link is a file system object that points to another file system object. |
| read_symlink_error             | read_symlink.error           | Number of erroneous ReadSymlink procedure requests. The ReadSymLink procedure reads the data associated with a symbolic link. A symbolic link is a file system object that points to another file system object.  |
| read_symlink_total             | read_symlink.total           | Total number of ReadSymLink procedure requests. It is the total number of read symlink success and read symlink error requests.   |
| read_symlink_percent           | read_symlink.percent         | Percentage of ReadSymLink procedure requests. This counter shows how often ReadSymLink requests happen out of all NFSv3 requests.   |
| read_symlink_avg_latency       | read_symlink.average_latency | Average latency of ReadSymLink procedure requests. The counter keeps track of the average response time of ReadSymLink requests.  |
| read_success                   | read.success                 | Number of successful Read procedure requests. The Read procedure reads data from a file.  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>   | <b>Description</b>  |
|--------------------------------|-----------------------|---|
| read_error                     | read.error            | Number of erroneous Read procedure requests. The Read procedure reads data from a file.                               |
| read_total                     | read.total            | Total number Read of procedure requests. It is the total number of read success and read error requests.              |
| read_percent                   | read.percent          | Percentage of Read procedure requests. This counter shows how often Read requests happen out of all NFSv3 requests.   |
| read_avg_latency               | read.average_latency  | Average latency of Read procedure requests. The counter keeps track of the average response time of Read requests.    |
| write_success                  | write.success         | Number of successful Write procedure requests. The Write procedure writes data to a file.                             |
| write_error                    | write.error           | Number of erroneous Write procedure requests. The Write procedure writes data to a file.                              |
| write_total                    | write.total           | Total number of Write procedure requests. It is the total number of write success and write error requests.           |
| write_percent                  | write.percent         | Percentage of Write procedure requests. This counter shows how often Write requests happen out of all NFSv3 requests. |
| write_avg_latency              | write.average_latency | Average latency of Write procedure requests. The counter keeps track of the average response time of Write requests.  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>  |
|--------------------------------|------------------------|---|
| create_success                 | create.success         | Number of successful Create procedure requests. The Create procedure creates a regular file.                            |
| create_error                   | create.error           | Number of erroneous Create procedure requests. The Create procedure creates a regular file.                             |
| create_total                   | create.total           | Total number Create of procedure requests. It is the total number of create success and create error requests.          |
| create_percent                 | create.percent         | Percentage of Create procedure requests. This counter shows how often Create requests happen out of all NFSv3 requests. |
| create_avg_latency             | create.average_latency | Average latency of Create procedure requests. The counter keeps track of the average response time of Create requests.  |
| mkdir_success                  | mkdir.success          | Number of successful Mkdir procedure requests. The Mkdir procedure creates a new subdirectory.                          |
| mkdir_error                    | mkdir.error            | Number of erroneous Mkdir procedure requests. The Mkdir procedure creates a new subdirectory.                           |
| mkdir_total                    | mkdir.total            | Total number Mkdir of procedure requests. It is the total number of Mkdir success and Mkdir error requests.             |
| mkdir_percent                  | mkdir.percent          | Percentage of Mkdir procedure requests. This counter shows how often Mkdir requests happen out of all NFSv3 requests.   |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>   |
|--------------------------------|-------------------------|--|
| mkdir_avg_latency              | mkdir.average_latency   | Average latency of Mkdir procedure requests. The counter keeps track of the average response time of Mkdir requests.   |
| symlink_success                | symlink.success         | Number of successful SymLink procedure requests. The SymLink procedure creates a symbolic link. A symbolic link is a file system object that points to another file system object. |
| symlink_error                  | symlink.error           | Number of erroneous SymLink procedure requests. The SymLink procedure creates a symbolic link. A symbolic link is a file system object that points to another file system object.  |
| symlink_total                  | symlink.total           | Total number SymLink of procedure requests. It is the total number of SymLink success and create SymLink requests.   |
| symlink_percent                | symlink.percent         | Percentage of Symlink procedure requests. This counter shows how often Symlink requests happen out of all NFSv3 requests.  |
| symlink_avg_latency            | symlink.average_latency | Average latency of SymLink procedure requests. The counter keeps track of the average response time of SymLink requests.   |
| mknod_success                  | mknod.success           | Number of successful Mknod procedure requests. The Mknod procedure creates a special file that can be device files or named pipes.   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>   | <b>Description</b>  |
|--------------------------------|-----------------------|---|
| mknod_error                    | mknod.error           | Number of erroneous Mknod procedure requests. The Mknod procedure creates a special file that can be device files or named pipes. |
| mknod_total                    | mknod.total           | Total number Mknod of procedure requests. It is the total number of Mknod success and Mknod error requests.                       |
| mknod_percent                  | mknod.percent         | Percentage of Mknod procedure requests. This counter shows how often Mknod requests happen out of all NFSv3 requests.             |
| mknod_avg_latency              | mknod.average_latency | Average latency of Mknod procedure requests. The counter keeps track of the average response time of Mknod requests.              |
| remove_success                 | remove.success        | Number of successful Remove procedure requests. The Remove procedure deletes an entry from a directory.                           |
| remove_error                   | remove.error          | Number of erroneous Remove procedure requests. The Remove procedure deletes an entry from a directory.                            |
| remove_total                   | remove.total          | Total number Remove of procedure requests. It is the total number of Remove success and Remove error requests.                    |
| remove_percent                 | remove.percent        | Percentage of Remove procedure requests. This counter shows how often Remove requests happen out of all NFSv3 requests.           |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>   |
|--------------------------------|------------------------|--|
| remove_avg_latency             | remove.average_latency | Average latency of Remove procedure requests. The counter keeps track of the average response time of Remove requests. |
| rmdir_success                  | rmdir.success          | Number of successful Rmdir procedure requests. The Rmdir procedure deletes a subdirectory from a directory.            |
| rmdir_error                    | rmdir.error            | Number of erroneous Rmdir procedure requests. The Rmdir procedure deletes a subdirectory from a directory.             |
| rmdir_total                    | rmdir.total            | Total number Rmdir of procedure requests. It is the total number of Rmdir success and Rmdir error requests.            |
| rmdir_percent                  | rmdir.percent          | Percentage of Rmdir procedure requests. This counter shows how often Rmdir requests happen out of all NFSv3 requests.  |
| rmdir_avg_latency              | rmdir.average_latency  | Average latency of Rmdir procedure requests. The counter keeps track of the average response time of Rmdir requests.   |
| rename_success                 | rename.success         | Number of successful Rename procedure requests. The Rename procedure renames the file.                                 |
| rename_error                   | rename.error           | Number of erroneous Rename procedure requests. The Rename procedure renames the file.                                  |
| rename_total                   | rename.total           | Total number Rename of procedure requests. It is the total number of Rename success and Rename error requests.         |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>  |
|--------------------------------|------------------------|---|
| rename_percent                 | rename.percent         | Percentage of Rename procedure requests. This counter shows how often Rename requests happen out of all NFSv3 requests.   |
| rename_avg_latency             | rename.average_latency | Average latency of Rename procedure requests. The counter keeps track of the average response time of Rename requests.  |
| link_success                   | link.success           | Number of successful Link procedure requests. The Link procedure creates a hard link. A hard link is a directory entry that associates a name with a file on a file system. |
| link_error                     | link.error             | Number of erroneous Link procedure requests. The Link procedure creates a hard link. A hard link is a directory entry that associates a name with a file on a file system.  |
| link_total                     | link.total             | Total number Link of procedure requests. It is the total number of Link success and Link error requests.  |
| link_percent                   | link.percent           | Percentage of Link procedure requests. This counter shows how often Link requests happen out of all NFSv3 requests.   |
| link_avg_latency               | link.average_latency   | Average latency of Link procedure requests. The counter keeps track of the average response time of Link requests.  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>  |
|--------------------------------|-------------------------|---|
| readdir_success                | readdir.success         | Number of successful ReadDir procedure requests. The ReadDir retrieves a variable number of entries from a directory and returns the name of file identifier for each.                |
| readdir_error                  | readdir.error           | Number of erroneous ReadDir procedure requests. The ReadDir retrieves a variable number of entries from a directory and returns the name of file identifier for each.                 |
| readdir_total                  | readdir.total           | Total number ReadDir of procedure requests. It is the total number of ReadDir success and ReadDir error requests.   |
| readdir_percent                | readdir.percent         | Percentage of ReadDir procedure requests. This counter shows how often ReadDir requests happen out of all NFSv3 requests.   |
| readdir_avg_latency            | readdir.average_latency | Average latency of ReadDir procedure requests. The counter keeps track of the average response time of ReadDir requests.  |
| readdirplus_success            | readdirplus.success     | Number of successful ReadDirPlus procedure requests. The ReadDirPlus retrieves a variable number of entries from a file system directory and returns complete information about each. |
| readdirplus_error              | readdirplus.error       | Number of erroneous ReadDirPlus procedure requests. The ReadDirPlus retrieves a variable number of entries from a file system directory and returns complete information about each.  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>         | <b>Description</b>  |
|--------------------------------|-----------------------------|---|
| readdirplus_total              | readdirplus.total           | Total number ReadDirPlus of procedure requests. It is the total number of ReadDirPlus success and ReadDirPlus error requests.     |
| readdirplus_percent            | readdirplus.percent         | Percentage of ReadDirPlus procedure requests. This counter shows how often ReadDirPlus requests happen out of all NFSv3 requests. |
| readdirplus_avg_latency        | readdirplus.average_latency | Average latency of ReadDirPlus procedure requests. The counter keeps track of the average response time of ReadDirPlus requests.  |
| fsstat_success                 | fsstat.success              | Number of successful FsStat procedure requests. The FsStat procedure retrieves volatile file system state information.            |
| fsstat_error                   | fsstat.error                | Number of erroneous FsStat procedure requests. The FsStat procedure retrieves volatile file system state information.             |
| fsstat_total                   | fsstat.total                | Total number FSStat of procedure requests. It is the total number of FSStat success and FSStat error requests.                    |
| fsstat_percent                 | fsstat.percent              | Percentage of FSStat procedure requests. This counter shows how often FSStat requests happen out of all NFSv3 requests.           |
| fsstat_avg_latency             | fsstat.average_latency      | Average latency of FSStat procedure requests. The counter keeps track of the average response time of FSStat requests.            |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>   |
|--------------------------------|------------------------|--|
| fsinfo_success                 | fsinfo.success         | Number of successful FsInfo procedure requests. The FsInfo procedure retrieves nonvolatile file system state information and provides general information about the NFS version 3 protocol server implementation |
| fsinfo_error                   | fsinfo.error           | Number of erroneous FsInfo procedure requests. The FsInfo procedure retrieves nonvolatile file system state information and provides general information about the NFS version 3 protocol server implementation  |
| fsinfo_total                   | fsinfo.total           | Total number FSInfo of procedure requests. It is the total number of FSInfo success and FSInfo error requests.   |
| fsinfo_percent                 | fsinfo.percent         | Percentage of FsInfo procedure requests. This counter shows how often FsInfo requests happen out of all NFSv3 requests.  |
| fsinfo_avg_latency             | fsinfo.average_latency | Average latency of FSInfo procedure requests. The counter keeps track of the average response time of FSInfo requests.   |
| pathconf_success               | pathconf.success       | Number of successful PathConf procedure requests. The PathConf retrieves the pathconf information for a file or directory.   |
| pathconf_error                 | pathconf.error         | Number of erroneous PathConf procedure requests. The PathConf retrieves the pathconf information for a file or directory.  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>   |
|--------------------------------|--------------------------|--|
| pathconf_total                 | pathconf.total           | Total number PathConf of procedure requests. It is the total number of PathConf success and PathConf error requests.   |
| pathconf_percent               | pathconf.percent         | Percentage of PathConf procedure requests. This counter shows how often PathConf requests happen out of all NFSv3 requests.  |
| pathconf_avg_latency           | pathconf.average_latency | Average latency of PathConf procedure requests. The counter keeps track of the average response time of PathConf requests.   |
| commit_success                 | commit.success           | Number of successful Commit procedure requests. The Commit procedure forces or flushes data to stable storage that was previously written with a Write procedure call with stable field set to UNSTABLE. |
| commit_error                   | commit.error             | Number of erroneous Commit procedure requests. The Commit procedure forces or flushes data to stable storage that was previously written with a Write procedure call with stable field set to UNSTABLE.  |
| commit_total                   | commit.total             | Total number of Commit procedure requests. It is the total number of Commit success and Commit error requests.   |
| commit_percent                 | commit.percent           | Percentage of Commit procedure requests. This counter shows how often Commit requests happen out of all NFSv3 requests.  |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>        | <b>Description</b>   |
|--------------------------------|----------------------------|--|
| commit_avg_latency             | commit.average_latency     | Average latency of Commit procedure requests. The counter keeps track of the average response time of Commit requests. |
| nfsv3_ops                      | ops                        | Total number of NFSv3 procedure requests per second.   |
| null_latency_hist              | null_latency_histogram     | Histogram of latency for Null operations.  |
| getattr_latency_hist           | getattr_latency_histogram  | Histogram of latency for Getattr operations.   |
| setattr_latency_hist           | setattr_latency_histogram  | Histogram of latency for Setattr operations.   |
| lookup_latency_hist            | lookup_latency_histogram   | Histogram of latency for LookUp operations.  |
| access_latency_hist            | access_latency_histogram   | Histogram of latency for Access operations.  |
| readlink_latency_hist          | readlink_latency_histogram | Histogram of latency for ReadLink operations.  |
| read_latency_hist              | read_latency_histogram     | Histogram of latency for Read operations.  |
| write_latency_hist             | write_latency_histogram    | Histogram of latency for Write operations.   |
| create_latency_hist            | create_latency_histogram   | Histogram of latency for Create operations.  |
| mkdir_latency_hist             | mkdir_latency_histogram    | Histogram of latency for Mkdir operations.   |
| symlink_latency_hist           | symlink_latency_histogram  | Histogram of latency for SymLink operations.   |
| mknod_latency_hist             | mknod_latency_histogram    | Histogram of latency for Mknod operations.   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>                               |
|--------------------------------|--------------------------------|--|
| remove_latency_hist            | remove_latency_histogram       | Histogram of latency for Remove operations.      |
| rmdir_latency_hist             | rmdir_latency_histogram        | Histogram of latency for Rmdir operations.       |
| rename_latency_hist            | rename_latency_histogram       | Histogram of latency for Rename operations.      |
| link_latency_hist              | link_latency_histogram         | Histogram of latency for Link operations.        |
| readdir_latency_hist           | readdir_latency_histogram      | Histogram of latency for Readdir operations.     |
| readdirplus_latency_hist       | readdir_plus_latency_histogram | Histogram of latency for Readdirplus operations. |
| fsstat_latency_hist            | fsstat_latency_histogram       | Histogram of latency for Fsstat operations.      |
| fsinfo_latency_hist            | fsinfo_latency_histogram       | Histogram of latency for Fsinfo operations.      |
| pathconf_latency_hist          | pathconf_latency_histogram     | Histogram of latency for Pathconf operations.    |
| comit_latency_hist             | commit_latency_histogram       | Histogram of latency for Commit operations.      |
| nfsv3_latency_hist             | latency_histogram              | Histogram of latency for NFSv3 operations.       |
| nfsv3_read_size_histo          | read_size_histogram            | Histogram of NFSv3 Read size.                    |
| nfsv3_write_size_histo         | write_size_histogram           | Histogram of NFSv3 Write size.                   |
| nfsv3_read_throughput          | read_throughput                | Rate of NFSv3 read data transfers per second.    |
| nfsv3_write_throughput         | write_throughput               | Rate of NFSv3 write data transfers per second.   |
| nfsv3_throughput               | throughput                     | Rate of NFSv3 data transfers per second.         |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>   |
|--------------------------------|--------------------------|--|
| nfsv3_dnfs_ops                 | oracle_direct_nfs_ops    | Rate of NFSv3 operations per second served to clients utilizing Oracle Direct NFS  |
| nfsv3_hadoop_ops               | hadoop_ops               | Rate of NFSv3 operations per second served to clients utilizing Hadoop NFS   |
| readdir_postop_error           | readdir.postop_error     | Number of failed Postop ReadDir procedures. Postop ReadDir errors are failures in the response of the request.                   |
| readdirplus_postop_error       | readdirplus.postop_error | Number of failed Postop ReadDirPlus procedures. Postop ReadDirPlus errors are failures in the response of the request.           |
| raidprop_error                 | raid_error               | Number of times RAID errors encountered in NFSv3 path.   |
| nfsv3_read_ops                 | read_ops                 | Total observed NFSv3 read operations per second.   |
| nfsv3_write_ops                | write_ops                | Total observed NFSv3 write operations per second.  |
| latency                        | latency                  | Average latency of NFSv3 requests. This counter keeps track of the average response time of NFSv3 requests.                      |
| nfsv3_fileop_max_latency       | fileop_max_latency       | Maximum latency of NFSv3 requests. This counter keeps track of the maximum response time of NFSv3 requests at any point in time. |
| total_ops                      | total_ops                | Total number of NFSv3 requests per sec   |
| free_pass_ops                  | free_pass_ops            | This counter keeps track of the number of NFSv3 requests which have asked to bypass COP HWM                                      |

# Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object  | Alias           |
|-----------------|-----------------|
| svm_nfs_v3:node | svm_nfs_v3_node |

# svm\_nfs\_v4

The svm\_nfs\_v4 table reports activity for the Network File System protocol, version 4. This is the ISOC file-sharing protocol that is predominant on UNIX platforms, used to connect to Network Attached Storage (NAS). The alias name for svm\_nfs\_v4:node is svm\_nfs\_v4\_node.

**Classic Object:** nfsv4

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description          |
|------------------------|---------------|----------------------|
| node_name              | node.name     | Node name            |
| instance_name          | name          | Constituent SVM Name |
| vserver_id             | svm.id        | SVM ID               |
| vserver_name           | svm.name      | SVM Name             |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>                       |
|--------------------------------|--------------------------|--|
| null_success                   | null.success             | Number of successful NULL procedures     |
| null_error                     | null.error               | Number of failed NULL procedures         |
| null_total                     | null.total               | Total number of NULL procedures          |
| null_percent                   | null.percent             | Percentage of NULL procedures            |
| null_avg_latency               | null.average_latency     | Average Latency of NULL procedures       |
| compound_success               | compound.success         | Number of successful COMPOUND procedures |
| compound_error                 | compound.error           | Number of failed COMPOUND procedures     |
| compound_total                 | compound.total           | Total number of COMPOUND procedures      |
| compound_percent               | compound.percent         | Percentage of COMPOUND procedures        |
| compound_avg_latency           | compound.average_latency | Average Latency of COMPOUND procedures   |
| access_success                 | access_success           | Number of successful ACCESS procedures   |
| access_error                   | access.error             | Number of failed ACCESS procedures       |
| access_total                   | access.total             | Total number of ACCESS procedures        |
| access_percent                 | access.percent           | Percentage of ACCESS procedures          |
| access_avg_latency             | access.average_latency   | Average latency of ACCESS procedures     |
| close_success                  | close.success            | Number of successful CLOSE procedures    |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>                         |
|--------------------------------|------------------------|--|
| close_error                    | close.error            | Number of failed CLOSE procedures          |
| close_total                    | close.total            | Total number of CLOSE procedures           |
| close_percent                  | close.percent          | Percentage of CLOSE procedures             |
| close_avg_latency              | close.average_latency  | Average latency of CLOSE procedures        |
| commit_success                 | commit.success         | Number of successful COMMIT procedures     |
| commit_error                   | commit.error           | Number of failed COMMIT procedures         |
| commit_total                   | commit.total           | Total number of COMMIT procedures          |
| commit_percent                 | commit.percent         | Percentage of COMMIT procedures            |
| commit_avg_latency             | commit.average_latency | Average latency of COMMIT procedures       |
| create_success                 | create.success         | Number of successful CREATE procedures     |
| create_error                   | create.error           | Number of failed CREATE procedures         |
| create_total                   | create.total           | Total number of CREATE procedures          |
| create_percent                 | create.percent         | Percentage of CREATE procedures            |
| create_avg_latency             | create.average_latency | Average latency of CREATE procedures       |
| delempurge_success             | delempurge.success     | Number of successful DELEMPURGE procedures |
| delempurge_error               | delempurge.error       | Number of failed DELEMPURGE procedures     |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>          | <b>Description</b>                           |
|--------------------------------|------------------------------|--|
| delempurge_total               | delempurge.total             | Total number of DELEMPURGE procedures        |
| delempurge_percent             | delempurge.percent           | Percentage of DELEMPURGE procedures          |
| delempurge_avg_latency         | delempurge.average_latency   | Average latency of DELEMPURGE procedures     |
| delempreturn_success           | delempreturn.success         | Number of successful DELEMPRETURN procedures |
| delempreturn_error             | delempreturn.error           | Number of failed DELEMPRETURN procedures     |
| delempreturn_total             | delempreturn.total           | Total number of DELEMPRETURN procedures      |
| delempreturn_percent           | delempreturn.percent         | Percentage of DELEMPRETURN procedures        |
| delempreturn_avg_latency       | delempreturn.average_latency | Average latency of DELEMPRETURN procedures   |
| getattr_success                | getattr.success              | Number of successful GETATTR procedures      |
| getattr_error                  | getattr.error                | Number of failed GETATTR procedures          |
| getattr_total                  | getattr.total                | Total number of GETATTR procedures           |
| getattr_percent                | getattr.percent              | Percentage of GETATTR procedures             |
| getattr_avg_latency            | getattr.average_latency      | Average latency of GETATTR procedures        |
| getfh_success                  | getfh.success                | Number of successful GETFH procedures        |
| getfh_error                    | getfh.error                  | Number of failed GETFH procedures            |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>   | <b>Description</b>                    |
|--------------------------------|-----------------------|---------------------------------------|
| getfh_total                    | getfh.total           | Total number of GETFH procedures      |
| getfh_percent                  | getfh.percent         | Percentage of GETFH procedures        |
| getfh_avg_latency              | getfh.average_latency | Average latency of GETFH procedures   |
| link_success                   | link.success          | Number of successful LINK procedures  |
| link_error                     | link.error            | Number of failed LINK procedures      |
| link_total                     | link.total            | Total number of LINK procedures       |
| link_percent                   | link.percent          | Percentage of LINK procedures         |
| link_avg_latency               | link.average_latency  | Average latency of LINK procedures    |
| lock_success                   | lock.success          | Number of successful LOCK procedures  |
| lock_error                     | lock.error            | Number of failed LOCK procedures      |
| lock_total                     | lock.total            | Total number of LOCK procedures       |
| lock_percent                   | lock.percent          | Percentage of LOCK procedures         |
| lock_avg_latency               | lock.average_latency  | Average latency of LOCK procedures    |
| lockt_success                  | lockt.success         | Number of successful LOCKT procedures |
| lockt_error                    | lockt.error           | Number of failed LOCKT procedures     |
| lockt_total                    | lockt.total           | Total number of LOCKT procedures      |
| lockt_percent                  | lockt.percent         | Percentage of LOCKT procedures        |
| lockt_avg_latency              | lockt.average_latency | Average latency of LOCKT procedures   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>                      |
|--------------------------------|-------------------------|---|
| locku_success                  | locku.success           | Number of successful LOCKU procedures   |
| locku_error                    | locku.error             | Number of failed LOCKU procedures       |
| locku_total                    | locku.total             | Total number of LOCKU procedures        |
| locku_percent                  | locku.percent           | Percentage of LOCKU procedures          |
| locku_avg_latency              | locku.average_latency   | Average latency of LOCKU procedures     |
| lookup_success                 | lookup.success          | Number of successful LOOKUP procedures  |
| lookup_error                   | lookup.error            | Number of failed LOOKUP procedures      |
| lookup_total                   | lookup.total            | Total number of LOOKUP procedures       |
| lookup_percent                 | lookup.percent          | Percentage of LOOKUP procedures         |
| lookup_avg_latency             | lookup.average_latency  | Average latency of LOOKUP procedures    |
| lookupp_success                | lookupp.success         | Number of successful LOOKUPP procedures |
| lookupp_error                  | lookupp.error           | Number of failed LOOKUPP procedures     |
| lookupp_total                  | lookupp.total           | Total number of LOOKUPP procedures      |
| lookupp_percent                | lookupp.percent         | Percentage of LOOKUPP procedures        |
| lookupp_avg_latency            | lookupp.average_latency | Average latency of LOOKUPP procedures   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>                           |
|--------------------------------|--------------------------|--|
| nverify_success                | nverify.success          | Number of successful NVERIFY procedures      |
| nverify_error                  | nverify.error            | Number of failed NVERIFY procedures          |
| nverify_total                  | nverify.total            | Total number of NVERIFY procedures           |
| nverify_percent                | nverify.percent          | Percentage of NVERIFY procedures             |
| nverify_avg_latency            | nverify.average_latency  | Average latency of NVERIFY procedures        |
| open_success                   | open.success             | Number of successful OPEN procedures         |
| open_error                     | open.error               | Number of failed OPEN procedures             |
| open_total                     | open.total               | Total number of OPEN procedures              |
| open_percent                   | open.percent             | Percentage of OPEN procedures                |
| open_avg_latency               | open.average_latency     | Average latency of OPEN procedures           |
| openattr_success               | openattr.success         | Number of successful OPENATTR procedures     |
| openattr_error                 | openattr.error           | Number of failed OPENATTR procedures         |
| openattr_total                 | openattr.total           | Total number of OPENATTR procedures          |
| openattr_percent               | openattr.percent         | Percentage of OPENATTR procedures            |
| openattr_avg_latency           | openattr.average_latency | Average latency of OPENATTR procedures       |
| open_confirm_success           | open_confirm.success     | Number of successful OPEN_CONFIRM procedures |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>                             |
|--------------------------------|--------------------------------|--|
| open_confirm_error             | open_confirm.error             | Number of failed OPEN_CONFIRM procedures       |
| open_confirm_total             | open_confirm.total             | Total number of OPEN_CONFIRM procedures        |
| open_confirm_percent           | open_confirm.percent           | Percentage of OPEN_CONFIRM procedures          |
| open_confirm_avg_latency       | open_confirm.average_latency   | Average latency of OPEN_CONFIRM procedures     |
| open_downgrade_success         | open_downgrade.success         | Number of successful OPEN_DOWNGRADE procedures |
| open_downgrade_error           | open_downgrade.error           | Number of failed OPEN_DOWNGRADE procedures     |
| open_downgrade_total           | open_downgrade.total           | Total number of OPEN_DOWNGRADE procedures      |
| open_downgrade_percent         | open_downgrade.percent         | Percentage of OPEN_DOWNGRADE procedures        |
| open_downgrade_avg_latency     | open_downgrade.average_latency | Average latency of OPEN_DOWNGRADE procedures   |
| putfh_success                  | putfh.success                  | Number of successful PUTFH procedures          |
| putfh_error                    | putfh.error                    | Number of failed PUTFH procedures              |
| putfh_total                    | putfh.total                    | Total number of PUTFH procedures               |
| putfh_percent                  | putfh.percent                  | Percentage of PUTFH procedures                 |
| putfh_avg_latency              | putfh.average_latency          | Average latency of PUTFH procedures            |
| putpubfh_success               | putpubfh.success               | Number of successful PUTPUBFH procedures       |
| putpubfh_error                 | putpubfh.error                 | Number of failed PUTPUBFH procedures           |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>       | <b>Description</b>                        |
|--------------------------------|---------------------------|---|
| putpubfh_total                 | putpubfh.total            | Total number of PUTPUBFH procedures       |
| putpubfh_percent               | putpubfh.percent          | Percentage of PUTPUBFH procedures         |
| putpubfh_avg_latency           | putpubfh.average_latency  | Average latency of PUTPUBFH procedures    |
| putrootfh_success              | putrootfh.success         | Number of successful PUTROOTFH procedures |
| putrootfh_error                | putrootfh.error           | Number of failed PUTROOTFH procedures     |
| putrootfh_total                | putrootfh.total           | Total number of PUTROOTFH procedures      |
| putrootfh_percent              | putrootfh.percent         | Percentage of PUTROOTFH procedures        |
| putrootfh_avg_latency          | putrootfh.average_latency | Average latency of PUTROOTFH procedures   |
| read_success                   | read.success              | Number of successful READ procedures      |
| read_error                     | read.error                | Number of failed READ procedures          |
| read_total                     | read.total                | Total number of READ procedures           |
| read_percent                   | read.percent              | Percentage of READ procedures             |
| read_avg_latency               | read.average_latency      | Average latency of READ procedures        |
| readdir_success                | readdir.success           | Number of successful REaddir procedures   |
| readdir_error                  | readdir.error             | Number of failed REaddir procedures       |
| readdir_total                  | readdir.total             | Total number of REaddir procedures        |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>                       |
|--------------------------------|--------------------------|--|
| readdir_percent                | readdir.percent          | Percentage of REaddir procedures         |
| readdir_avg_latency            | readdir.average_latency  | Average latency of REaddir procedures    |
| readlink_success               | readlink.success         | Number of successful READLINK procedures |
| readlink_error                 | readlink.error           | Number of failed READLINK procedures     |
| readlink_total                 | readlink.total           | Total number of READLINK procedures      |
| readlink_percent               | readlink.percent         | Percentage of READLINK procedures        |
| readlink_avg_latency           | readlink.average_latency | Average latency of READLINK procedures   |
| remove_success                 | remove.success           | Number of successful REMOVE procedures   |
| remove_error                   | remove.error             | Number of failed REMOVE procedures       |
| remove_total                   | remove.total             | Total number of REMOVE procedures        |
| remove_percent                 | remove.percent           | Percentage of REMOVE procedures          |
| remove_avg_latency             | remove.average_latency   | Average latency of REMOVE procedures     |
| rename_success                 | rename.success           | Number of successful RENAME procedures   |
| rename_error                   | rename.error             | Number of failed RENAME procedures       |
| rename_total                   | rename.total             | Total number of RENAME procedures        |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>       | <b>Description</b>                        |
|--------------------------------|---------------------------|---|
| rename_percent                 | rename.percent            | Percentage of RENAME procedures           |
| rename_avg_latency             | rename.average_latency    | Average latency of RENAME procedures      |
| renew_success                  | renew.success             | Number of successful RENEW procedures     |
| renew_error                    | renew.error               | Number of failed RENEW procedures         |
| renew_total                    | renew.total               | Total number of RENEW procedures          |
| renew_percent                  | renew.percent             | Percentage of RENEW procedures            |
| renew_avg_latency              | renew.average_latency     | Average latency of RENEW procedures       |
| restorefh_success              | restorefh.success         | Number of successful RESTOREFH procedures |
| restorefh_error                | restorefh.error           | Number of failed RESTOREFH procedures     |
| restorefh_total                | restorefh.total           | Total number of RESTOREFH procedures      |
| restorefh_percent              | restorefh.percent         | Percentage of RESTOREFH procedures        |
| restorefh_avg_latency          | restorefh.average_latency | Average latency of RESTOREFH procedures   |
| savefh_success                 | savefh.success            | Number of successful SAVEFH procedures    |
| savefh_error                   | savefh.error              | Number of failed SAVEFH procedures        |
| savefh_total                   | savefh.total              | Total number of SAVEFH procedures         |
| savefh_percent                 | savefh.percent            | Percentage of SAVEFH procedures           |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>                          |
|--------------------------------|-------------------------|---|
| savefh_avg_latency             | savefh.average_latency  | Average latency of SAVEFH procedures        |
| secinfo_success                | secinfo.success         | Number of successful SECINFO procedures     |
| secinfo_error                  | secinfo.error           | Number of failed SECINFO procedures         |
| secinfo_total                  | secinfo.total           | Total number of SECINFO procedures          |
| secinfo_percent                | secinfo.percent         | Percentage of SECINFO procedures            |
| secinfo_avg_latency            | secinfo.average_latency | Average latency of SECINFO procedures       |
| setattr_success                | setattr.success         | Number of successful SETATTR procedures     |
| setattr_error                  | setattr.error           | Number of failed SETATTR procedures         |
| setattr_total                  | setattr.total           | Total number of SETATTR procedures          |
| setattr_percent                | setattr.percent         | Percentage of SETATTR procedures            |
| setattr_avg_latency            | setattr.average_latency | Average latency of SETATTR procedures       |
| setclientid_success            | setclientid.success     | Number of successful SETCLIENTID procedures |
| setclientid_error              | setclientid.error       | Number of failed SETCLIENTID procedures     |
| setclientid_total              | setclientid.total       | Total number of SETCLIENTID procedures      |
| setclientid_percent            | setclientid.percent     | Percentage of SETCLIENTID procedures        |



| <b>Classic Numeric Counter</b>  | <b>REST Counter</b>                 | <b>Description</b>                                  |
|---------------------------------|-------------------------------------|---|
| setclientid_avg_latency         | setclientid.average_latency         | Average latency of SETCLIENTID procedures           |
| setclientid_confirm_success     | setclientid_confirm.success         | Number of successful SETCLIENTID_CONFIRM procedures |
| setclientid_confirm_error       | setclientid_confirm.error           | Number of failed SETCLIENTID_CONFIRM procedures     |
| setclientid_confirm_total       | setclientid_confirm.total           | Total number of SETCLIENTID_CONFIRM procedures      |
| setclientid_confirm_percent     | setclientid_confirm.percent         | Percentage of SETCLIENTID_CONFIRM procedures        |
| setclientid_confirm_avg_latency | setclientid_confirm.average_latency | Average latency of SETCLIENTID_CONFIRM procedures   |
| verify_success                  | verify.success                      | Number of successful VERIFY procedures              |
| verify_error                    | verify.error                        | Number of failed VERIFY procedures                  |
| verify_total                    | verify.total                        | Total number of VERIFY procedures                   |
| verify_percent                  | verify.percent                      | Percentage of VERIFY procedures                     |
| verify_avg_latency              | verify.average_latency              | Average latency of VERIFY procedures                |
| write_success                   | write.success                       | Number of successful WRITE procedures               |
| write_error                     | write.error                         | Number of failed WRITE procedures                   |
| write_total                     | write.total                         | Total number of WRITE procedures                    |
| write_percent                   | write.percent                       | Percentage of WRITE procedures                      |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                | <b>Description</b>                                |
|--------------------------------|------------------------------------|---|
| write_avg_latency              | write.average_latency              | Average Latency of WRITE procedures               |
| release_lock_owner_success     | release_lock_owner.success         | Number of successful RELEASE_LOCKOWNER procedures |
| release_lock_owner_error       | release_lock_owner.error           | Number of failed RELEASE_LOCKOWNER procedures     |
| release_lock_owner_total       | release_lock_owner.total           | Total number of RELEASE_LOCKOWNER procedures      |
| release_lock_owner_percent     | release_lock_owner.percent         | Percentage of RELEASE_LOCKOWNER procedures        |
| release_lock_owner_avg_latency | release_lock_owner.average_latency | Average Latency of RELEASE_LOCKOWNER procedures   |
| nfsv4_ops                      | ops                                | Total number of NFSv4 procedures per second.      |
| getattr_latency_hist           | getattr_latency_histogram          | Histogram of latency for GETATTR procedures       |
| lookup_latency_hist            | lookup_latency_histogram           | Histogram of latency for LOOKUP procedures        |
| access_latency_hist            | access_latency_histogram           | Histogram of latency for ACCESS procedures        |
| read_latency_hist              | read_latency_histogram             | Histogram of latency for READ procedures          |
| write_latency_hist             | write_latency_histogram            | Histogram of latency for WRITE procedures         |
| remove_latency_hist            | remove_latency_histogram           | Histogram of latency for REMOVE procedures        |
| nfsv4_latency_hist             | total.latency_histogram            | Histogram of latency for NFSv4 procedures         |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                 | <b>Description</b>   |
|--------------------------------|-------------------------------------|--|
| nfs4_read_size_histo           | total.read_size_histogram           | Histogram of NFSv4 Read size   |
| nfs4_write_size_histo          | total.write_size_histogram          | Histogram of NFSv4 Write size  |
| nfs4_read_throughput           | total.read_throughput               | NFSv4 read data transfers  |
| nfs4_write_throughput          | total.write_throughput              | NFSv4 write data transfers   |
| nfs4_throughput                | total.throughput                    | NFSv4 data transfers   |
| latency                        | latency                             | Average latency of NFSv4 requests. This counter keeps track of the average response time of NFSv4 requests.                      |
| total_ops                      | total_ops                           | Total number of NFSv4 requests per second.   |
| nfsv4_fileop_max_latency       | fileop_max_latency                  | Maximum latency of NFSv4 requests. This counter keeps track of the maximum response time of NFSv4 requests at any point in time. |
| nfs4_cbmgr_user_req_canceled   | callback_manager.user_req_cancelled | Number of times NFS4 Callback Manager user canceled after scheduling to send.  |
| nfs4_cbmgr_user_rsrc_cleaned   | callback_manager.user_rsrc_cleaned  | Number of times NFS4 Callback Manager user cleaned before sending.   |
| nfs4_cbmgr_user_rsrc_denied    | callback_manager.user_rsrc_denied   | Number of times NFS4 Callback Manager denied contexts.   |
| nfs4_cbmgr_user_req_ok         | callback_manager.user_req_ok        | Number of times NFS4 Callback Manager gave contexts.   |
| nfs4_cbmgr_user_req            | callback_manager.user_req           | Number of times NFS4 Callback Manager users asked for context.   |
| nfs4_cbmgr_rep_timeout         | callback_manager.reply_timeout      | Number of times NFS4 Callback Manager reply timed out.   |
| callback_manager.recieved      | callback_manager.recieved           | Number of times NFS4 Callback Manager recieved.  |

| Classic Numeric Counter | REST Counter                       | Description  |
|-------------------------|------------------------------------|--|
| nfscbmgr_send           | callback_manager.send              | Number of times NFS4 Callback Manager send request.  |
| nfscbmgr_inuse_ctx      | callback_manager.inuse_context     | Number of times NFS4 Callback Manager inuse context. |
| nfscbmgr_free_ctx       | callback_manager.free_context      | Number of times NFS4 Callback Manager free context.  |
| nfscbmgr_latency_hist   | callback_manager.latency_histogram | NFS Callback Manager latency bin.                    |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object  | Alias           |
|-----------------|-----------------|
| svm_nfs_v4:node | svm_nfs_v4_node |

# svm\_nfs\_v41

The svm\_nfs\_v41 table reports activity for the Network File System protocol, version 4.1 or later. This is the file-sharing protocol that implements Parallel NFS (pNFS), used to connect to Network Attached Storage (NAS). The alias name for svm\_nfs\_v41:node is svm\_nfs\_v41\_node.

**Classic Object:** nfsv4\_1

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description          |
|------------------------|---------------|----------------------|
| node_name              | node.name     | Node name            |
| instance_name          | name          | Constituent SVM name |
| svm.id                 | svm.id        | SVM ID               |
| svm.name               | svm.name      | SVM Name             |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>                                    |
|--------------------------------|--------------------------|---|
| null_success                   | null.success             | Number of successful NFSv4.1 NULL procedures.         |
| null_error                     | null.error               | Number of failed NFSv4.1 NULL procedures.             |
| null_total                     | null.total               | Total number of NFSv4.1 NULL procedures.              |
| null_percent                   | null.percent             | Percentage of NFSv4.1 NULL procedures.                |
| null_avg_latency               | null.average_latency     | Average latency of NFSv4.1 NULL procedures.           |
| compound_success               | compound.success         | The number of successful NFSv4.1 COMPOUND procedures. |
| compound_error                 | compound.error           | The number of failed NFSv4.1 COMPOUND procedures.     |
| compound_total                 | compound.total           | Total number of NFSv4.1 COMPOUND procedures.          |
| compound_percent               | compound.percent         | Percentage of NFSv4.1 COMPOUND procedures.            |
| compound_avg_latency           | compound.average_latency | Average latency of NFSv4.1 COMPOUND procedures.       |
| access_success                 | access.success           | Number of successful NFSv4.1 ACCESS operations.       |
| access_error                   | access.error             | Number of failed NFSv4.1 ACCESS operations.           |
| access_total                   | access.total             | Total number of NFSv4.1 ACCESS operations.            |
| access_percent                 | access.percent           | Percentage of NFSv4.1 ACCESS operations.              |
| access_avg_latency             | access.average_latency   | Average latency of NFSv4.1 ACCESS operations.         |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>                                  |
|--------------------------------|------------------------|---|
| close_success                  | close.success          | The number of successful NFSv4.1 CLOSE operations.  |
| close_error                    | close.error            | The number of failed NFSv4.1 CLOSE operations.      |
| close_total                    | close.total            | Total number of NFSv4.1 CLOSE operations.           |
| close_percent                  | close.percent          | Percentage of NFSv4.1 CLOSE operations.             |
| close_avg_latency              | close.average_latency  | Average latency of NFSv4.1 CLOSE operations.        |
| commit_success                 | commit.success         | The number of successful NFSv4.1 COMMIT operations. |
| commit_error                   | commit.error           | The number of failed NFSv4.1 COMMIT operations.     |
| commit_total                   | commit.total           | Total number of NFSv4.1 COMMIT operations.          |
| commit_percent                 | commit.percent         | Percentage of NFSv4.1 COMMIT operations.            |
| commit_avg_latency             | commit.average_latency | Average latency of NFSv4.1 COMMIT operations.       |
| create_success                 | create.success         | The number of successful NFSv4.1 CREATE operations. |
| create_error                   | create.error           | The number of failed NFSv4.1 CREATE operations.     |
| create_total                   | create.total           | Total number of NFSv4.1 CREATE operations.          |
| create_percent                 | create.percent         | Percentage of NFSv4.1 CREATE operations.            |
| create_avg_latency             | create.average_latency | Average latency of NFSv4.1 CREATE operations.       |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>         | <b>Description</b>                                       |
|--------------------------------|-----------------------------|--|
| delempurge_success             | delempurge.success          | The number of successful NFSv4.1 DELEMPURGE operations.  |
| delempurge_error               | delempurge.error            | The number of failed NFSv4.1 DELEMPURGE operations.      |
| delempurge_total               | delempurge.total            | Total number of NFSv4.1 DELEMPURGE operations.           |
| delempurge_percent             | delempurge.percent          | Percentage of NFSv4.1 DELEMPURGE operations.             |
| delempurge_avg_latency         | delempurge.average_latency  | Average latency of NFSv4.1 DELEMPURGE operations.        |
| delegreturn_success            | delegreturn.success         | The number of successful NFSv4.1 DELEGRETURN operations. |
| delegreturn_error              | delegreturn.error           | The number of failed NFSv4.1 DELEGRETURN operations.     |
| delegreturn_total              | delegreturn.total           | Total number of NFSv4.1 DELEGRETURN operations.          |
| delegreturn_percent            | delegreturn.percent         | Percentage of NFSv4.1 DELEGRETURN operations.            |
| delegreturn_avg_latency        | delegreturn.average_latency | Average latency of NFSv4.1 DELEGRETURN operations.       |
| getattr_success                | getattr.success             | The number of successful NFSv4.1 GETATTR operations.     |
| getattr_error                  | getattr.error               | The number of failed NFSv4.1 GETATTR operations.         |
| getattr_total                  | getattr.total               | Total number of NFSv4.1 GETATTR operations.              |
| getattr_percent                | getattr.percent             | Percentage of NFSv4.1 GETATTR operations.                |
| getattr_avg_latency            | getattr.average_latency     | Average latency of NFSv4.1 GETATTR operations.           |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>   | <b>Description</b>                                 |
|--------------------------------|-----------------------|--|
| getfh_success                  | getfh.success         | The number of successful NFSv4.1 GETFH operations. |
| getfh_error                    | getfh.error           | The number of failed NFSv4.1 GETFH operations.     |
| getfh_total                    | getfh.total           | Total number of NFSv4.1 GETFH operations.          |
| getfh_percent                  | getfh.percent         | Percentage of NFSv4.1 GETFH operations.            |
| getfh_avg_latency              | getfh.average_latency | Average latency of NFSv4.1 GETFH operations.       |
| link_success                   | link.success          | The number of successful NFSv4.1 LINK operations.  |
| link_error                     | link.error            | The number of failed NFSv4.1 LINK operations.      |
| link_total                     | link.total            | Total number of NFSv4.1 LINK operations.           |
| link_percent                   | link.percent          | Percentage of NFSv4.1 LINK operations.             |
| link_avg_latency               | link.average_latency  | Average latency of NFSv4.1 LINK operations.        |
| lock_success                   | lock.success          | The number of successful NFSv4.1 LOCK operations.  |
| lock_error                     | lock.error            | The number of failed NFSv4.1 LOCK operations.      |
| lock_total                     | lock.total            | Total number of NFSv4.1 LOCK operations.           |
| lock_percent                   | lock.percent          | Percentage of NFSv4.1 LOCK operations.             |
| lock_avg_latency               | lock.average_latency  | Average latency of NFSv4.1 LOCK operations.        |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>    | <b>Description</b>                                  |
|--------------------------------|------------------------|---|
| lockt_success                  | lockt.success          | The number of successful NFSv4.1 LOCKT operations.  |
| lockt_error                    | lockt.error            | The number of failed NFSv4.1 LOCKT operations.      |
| lockt_total                    | lockt.total            | Total number of NFSv4.1 LOCKT operations.           |
| lockt_percent                  | lockt.percent          | Percentage of NFSv4.1 LOCKT operations.             |
| lockt_avg_latency              | lockt.average_latency  | Average latency of NFSv4.1 LOCKT operations.        |
| locku_success                  | locku.success          | The number of successful NFSv4.1 LOCKU operations.  |
| locku_error                    | locku.error            | The number of failed NFSv4.1 LOCKU operations.      |
| locku_total                    | locku.total            | Total number of NFSv4.1 LOCKU operations.           |
| locku_percent                  | locku.percent          | Percentage of NFSv4.1 LOCKU operations.             |
| locku_avg_latency              | locku.average_latency  | Average latency of NFSv4.1 LOCKU operations.        |
| lookup_success                 | lookup.success         | The number of successful NFSv4.1 LOOKUP operations. |
| lookup_error                   | lookup.error           | The number of failed NFSv4.1 LOOKUP operations.     |
| lookup_total                   | lookup.total           | Total number of NFSv4.1 LOOKUP operations.          |
| lookup_percent                 | lookup.percent         | Percentage of NFSv4.1 LOOKUP operations.            |
| lookup_avg_latency             | lookup.average_latency | Average latency of NFSv4.1 LOOKUP operations.       |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>                                   |
|--------------------------------|-------------------------|--|
| lookupp_success                | lookupp.success         | The number of successful NFSv4.1 LOOKUPP operations. |
| lookupp_error                  | lookupp.error           | The number of failed NFSv4.1 LOOKUPP operations.     |
| lookupp_total                  | lookupp.total           | Total number of NFSv4.1 LOOKUPP operations.          |
| lookupp_percent                | lookupp.percent         | Percentage of NFSv4.1 LOOKUPP operations.            |
| lookupp_avg_latency            | lookupp.average_latency | Average latency of NFSv4.1 LOOKUPP operations.       |
| nverify_success                | nverify.success         | The number of successful NFSv4.1 NVERIFY operations. |
| nverify_error                  | nverify.error           | The number of failed NFSv4.1 NVERIFY operations.     |
| nverify_total                  | nverify.total           | Total number of NFSv4.1 NVERIFY operations.          |
| nverify_percent                | nverify.percent         | Percentage of NFSv4.1 NVERIFY operations.            |
| nverify_avg_latency            | nverify.average_latency | Average latency of NFSv4.1 NVERIFY operations.       |
| open_success                   | open.success            | The number of successful NFSv4.1 OPEN operations.    |
| open_error                     | open.error              | The number of failed NFSv4.1 OPEN operations.        |
| open_total                     | open.total              | Total number of NFSv4.1 OPEN operations.             |
| open_percent                   | open.percent            | Percentage of NFSv4.1 OPEN operations.               |
| open_avg_latency               | open.average_latency    | Average latency of NFSv4.1 OPEN operations.          |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>                                      |
|--------------------------------|--------------------------------|---|
| openattr_success               | openattr.success               | The number of successful NFSv4.1 OPENATTR operations.   |
| openattr_error                 | openattr.error                 | The number of failed NFSv4.1 OPENATTR operations.       |
| openattr_total                 | openattr.total                 | Total number of NFSv4.1 OPENATTR operations.            |
| openattr_percent               | openattr.percent               | Percentage of NFSv4.1 OPENATTR operations.              |
| openattr_avg_latency           | openattr.average_latency       | Average latency of NFSv4.1 OPENATTR operations.         |
| open_downgrade_success         | open_downgrade.success         | Number of successful NFSv4.1 OPEN_DOWNGRADE operations. |
| open_downgrade_error           | open_downgrade.error           | The number of failed NFSv4.1 OPEN_DOWNGRADE operations. |
| open_downgrade_total           | open_downgrade.total           | Total number of NFSv4.1 OPEN_DOWNGRADE operations.      |
| open_downgrade_percent         | open_downgrade.percent         | Percentage of NFSv4.1 OPEN_DOWNGRADE operations.        |
| open_downgrade_avg_latency     | open_downgrade.average_latency | Average latency of NFSv4.1 OPEN_DOWNGRADE operations.   |
| putfh_success                  | putfh.success                  | The number of successful NFSv4.1 PUTFH operations.      |
| putfh_error                    | putfh.error                    | The number of failed NFSv4.1 PUTFH operations.          |
| putfh_total                    | putfh.total                    | Total number of NFSv4.1 PUTFH operations.               |
| putfh_percent                  | putfh.percent                  | Percentage of NFSv4.1 PUTFH operations.                 |
| putfh_avg_latency              | putfh.average_latency          | The number of successful NFSv4.1 PUTPUBFH operations.   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>       | <b>Description</b>                                     |
|--------------------------------|---------------------------|--|
| putpubfh_success               | putpubfh.success          | The number of successful NFSv4.1 PUTPUBFH operations.  |
| putpubfh_error                 | putpubfh.error            | The number of failed NFSv4.1 PUTPUBFH operations.      |
| putpubfh_total                 | putpubfh.total            | Total number of NFSv4.1 PUTPUBFH operations.           |
| putpubfh_percent               | putpubfh.percent          | Percentage of NFSv4.1 PUTPUBFH operations.             |
| putpubfh_avg_latency           | putpubfh.average_latency  | Average latency of NFSv4.1 PUTPUBFH operations.        |
| putrootfh_success              | putrootfh.success         | The number of successful NFSv4.1 PUTROOTFH operations. |
| putrootfh_error                | putrootfh.error           | The number of failed NFSv4.1 PUTROOTFH operations.     |
| putrootfh_total                | putrootfh.total           | Total number of NFSv4.1 PUTROOTFH operations.          |
| putrootfh_percent              | putrootfh.percent         | Percentage of NFSv4.1 PUTROOTFH operations.            |
| putrootfh_avg_latency          | putrootfh.average_latency | Average latency of NFSv4.1 PUTROOTFH operations.       |
| read_success                   | read.success              | The number of successful NFSv4.1 READ operations.      |
| read_error                     | read.error                | The number of failed NFSv4.1 READ operations.          |
| read_total                     | read.total                | Total number of NFSv4.1 READ operations.               |
| read_percent                   | read.percent              | Percentage of NFSv4.1 READ operations.                 |
| read_avg_latency               | read.average_latency      | Average latency of NFSv4.1 READ operations.            |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>                                    |
|--------------------------------|--------------------------|---|
| readdir_success                | readdir.success          | The number of successful NFSv4.1 REaddir operations.  |
| readdir_error                  | readdir.error            | The number of failed NFSv4.1 REaddir operations.      |
| readdir_total                  | readdir.total            | Total number of NFSv4.1 REaddir operations.           |
| readdir_percent                | readdir.percent          | Percentage of NFSv4.1 REaddir operations.             |
| readdir_avg_latency            | readdir.average_latency  | Average latency of NFSv4.1 REaddir operations.        |
| readlink_success               | readlink.success         | The number of successful NFSv4.1 READLINK operations. |
| readlink_error                 | readlink.error           | The number of failed NFSv4.1 READLINK operations.     |
| readlink_total                 | readlink.total           | Total number of NFSv4.1 READLINK operations.          |
| readlink_percent               | readlink.percent         | Percentage of NFSv4.1 READLINK operations.            |
| readlink_avg_latency           | readlink.average_latency | Average latency of NFSv4.1 READLINK operations.       |
| remove_success                 | remove.success           | The number of successful NFSv4.1 REMOVE operations.   |
| remove_error                   | remove.error             | Number of failed NFSv4.1 REMOVE operations.           |
| remove_total                   | remove.total             | Total number of NFSv4.1 REMOVE operations.            |
| remove_percent                 | remove.percent           | Percentage of NFSv4.1 REMOVE operations.              |
| remove_avg_latency             | remove.average_latency   | Average latency of NFSv4.1 REMOVE operations.         |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>       | <b>Description</b>                                     |
|--------------------------------|---------------------------|--|
| rename_success                 | rename.success            | The number of successful NFSv4.1 RENAME operations.    |
| rename_error                   | rename.error              | The number of failed NFSv4.1 RENAME operations.        |
| rename_total                   | rename.total              | Total number of NFSv4.1 RENAME operations.             |
| rename_percent                 | rename.percent            | Percentage of NFSv4.1 RENAME operations.               |
| rename_avg_latency             | rename.average_latency    | Average latency of NFSv4.1 RENAME operations.          |
| restorefh_success              | restorefh.success         | The number of successful NFSv4.1 RESTOREFH operations. |
| restorefh_error                | restorefh.error           | The number of failed NFSv4.1 RESTOREFH operations.     |
| restorefh_total                | restorefh.total           | Total number of NFSv4.1 RESTOREFH operations.          |
| restorefh_percent              | restorefh.percent         | Percentage of NFSv4.1 RESTOREFH operations.            |
| restorefh_avg_latency          | restorefh.average_latency | Average latency of NFSv4.1 RESTOREFH operations.       |
| savefh_success                 | savefh.success            | The number of successful NFSv4.1 SAVEFH operations.    |
| savefh_error                   | savefh.error              | The number of failed NFSv4.1 SAVEFH operations.        |
| savefh_total                   | savefh.total              | Total number of NFSv4.1 SAVEFH operations.             |
| savefh_percent                 | savefh.percent            | Percentage of NFSv4.1 SAVEFH operations.               |
| savefh_avg_latency             | savefh.average_latency    | Average latency of NFSv4.1 SAVEFH operations.          |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>                                   |
|--------------------------------|-------------------------|--|
| secinfo_success                | secinfo.success         | The number of successful NFSv4.1 SECINFO operations. |
| secinfo_error                  | secinfo.error           | The number of failed NFSv4.1 SECINFO operations.     |
| secinfo_total                  | secinfo.total           | Total number of NFSv4.1 SECINFO operations.          |
| secinfo_percent                | secinfo.percent         | Percentage of NFSv4.1 SECINFO operations.            |
| secinfo_avg_latency            | secinfo.average_latency | Average latency of NFSv4.1 SECINFO operations.       |
| setattr_success                | setattr.success         | The number of successful NFSv4.1 SETATTR operations. |
| setattr_error                  | setattr.error           | The number of failed NFSv4.1 SETATTR operations.     |
| setattr_total                  | setattr.total           | Total number of NFSv4.1 SETATTR operations.          |
| setattr_percent                | setattr.percent         | Percentage of NFSv4.1 SETATTR operations.            |
| setattr_avg_latency            | setattr.average_latency | Average latency of NFSv4.1 SETATTR operations.       |
| verify_success                 | verify.success          | The number of successful NFSv4.1 VERIFY operations.  |
| verify_error                   | verify.error            | The number of failed NFSv4.1 VERIFY operations.      |
| verify_total                   | verify.total            | Total number of NFSv4.1 VERIFY operations.           |
| verify_percent                 | verify.percent          | Percentage of NFSv4.1 VERIFY operations.             |
| verify_avg_latency             | verify.average_latency  | Average latency of NFSv4.1 VERIFY operations.        |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                 | <b>Description</b>  |
|--------------------------------|-------------------------------------|---|
| write_success                  | write.success                       | The number of successful NFSv4.1 WRITE operations.                |
| write_error                    | write.error                         | The number of failed NFSv4.1 WRITE operations.                    |
| write_total                    | write.total                         | Total number of NFSv4.1 WRITE operations.                         |
| write_percent                  | write.percent                       | Percentage of NFSv4.1 WRITE operations.                           |
| write_avg_latency              | write.average_latency               | Average latency of NFSv4.1 WRITE operations.                      |
| backchannel_ctl_success        | backchannel_ctl.success             | The number of successful NFSv4.1 BACKCHANNEL_CTL operations.      |
| backchannel_ctl_error          | backchannel_ctl.error               | The number of failed NFSv4.1 BACKCHANNEL_CTL operations.          |
| backchannel_ctl_total          | backchannel_ctl.total               | Total number of NFSv4.1 BACKCHANNEL_CTL operations.               |
| backchannel_ctl_percent        | backchannel_ctl.percent             | Percentage of NFSv4.1 BACKCHANNEL_CTL operations.                 |
| backchannel_ctl_avg_latency    | backchannel_ctl.average_latency     | Average latency of NFSv4.1 BACKCHANNEL_CTL operations.            |
| bind_conn_to_session_success   | bind_connections_to_session.success | The number of successful NFSv4.1 BIND_CONN_TO_SESSION operations. |
| bind_conn_to_session_error     | bind_connections_to_session.error   | The number of failed NFSv4.1 BIND_CONN_TO_SESSION operations.     |
| bind_conn_to_session_total     | bind_connections_to_session.total   | Total number of NFSv4.1 BIND_CONN_TO_SESSION operations.          |
| bind_conn_to_session_percent   | bind_connections_to_session.percent | Percentage of NFSv4.1 BIND_CONN_TO_SESSION operations.            |

| <b>Classic Numeric Counter</b>   | <b>REST Counter</b>                         | <b>Description</b>   |
|----------------------------------|---|--|
| bind_conn_to_session_avg_latency | bind_connections_to_session.average_latency | Average latency of NFSv4.1 BIND_CONN_TO_SESSION operations.  |
| exchange_id_success              | exchange_id.success                         | The number of successful NFSv4.1 EXCHANGE_ID operations.     |
| exchange_id_error                | exchange_id.error                           | The number of failed NFSv4.1 EXCHANGE_ID operations.         |
| exchange_id_total                | exchange_id.total                           | Total number of NFSv4.1 EXCHANGE_ID operations.              |
| exchange_id_percent              | exchange_id.percent                         | Percentage of NFSv4.1 EXCHANGE_ID operations.                |
| exchange_id_avg_latency          | exchange_id.average_latency                 | Average latency of NFSv4.1 EXCHANGE_ID operations.           |
| create_session_success           | create_session.success                      | The number of successful NFSv4.1 CREATE_SESSION operations.  |
| create_session_error             | create_session.error                        | The number of failed NFSv4.1 CREATE_SESSION operations.      |
| create_session_total             | create_session.total                        | Total number of NFSv4.1 CREATE_SESSION operations.           |
| create_session_percent           | create_session.percent                      | Percentage of NFSv4.1 CREATE_SESSION operations.             |
| create_session_avg_latency       | create_session.average_latency              | Average latency of NFSv4.1 CREATE_SESSION operations.        |
| destroy_session_success          | destroy_session.success                     | The number of successful NFSv4.1 DESTROY_SESSION operations. |
| destroy_session_error            | destroy_session.error                       | The number of failed NFSv4.1 DESTROY_SESSION operations.     |
| destroy_session_total            | destroy_session.total                       | Total number of NFSv4.1 DESTROY_SESSION operations.          |
| destroy_session_percent          | destroy_session.percent                     | Percentage of NFSv4.1 DESTROY_SESSION operations.            |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                | <b>Description</b>  |
|--------------------------------|------------------------------------|---|
| destroy_session_avg_latency    | destroy_session.average_latency    | Average latency of NFSv4.1 DESTROY_SESSION operations.          |
| free_stateid_success           | free_stateid.success               | The number of successful NFSv4.1 FREE_STATEID operations.       |
| free_stateid_error             | free_stateid.error                 | The number of failed NFSv4.1 FREE_STATEID operations.           |
| free_stateid_total             | free_stateid.total                 | Total number of NFSv4.1 FREE_STATEID operations.                |
| free_stateid_percent           | free_stateid.percent               | Percentage of NFSv4.1 FREE_STATEID operations.                  |
| free_stateid_avg_latency       | free_stateid.average_latency       | Average latency of NFSv4.1 FREE_STATEID operations.             |
| get_dir_delegation_success     | get_dir_delegation.success         | The number of successful NFSv4.1 GET_DIR_DELEGATION operations. |
| get_dir_delegation_error       | get_dir_delegation.error           | The number of failed NFSv4.1 GET_DIR_DELEGATION operations.     |
| get_dir_delegation_total       | get_dir_delegation.total           | Total number of NFSv4.1 GET_DIR_DELEGATION operations.          |
| get_dir_delegation_percent     | get_dir_delegation_percent         | Percentage of NFSv4.1 GET_DIR_DELEGATION operations.            |
| get_dir_delegation_avg_latency | get_dir_delegation.average_latency | Average latency of NFSv4.1 GET_DIR_DELEGATION operations.       |
| getdeviceinfo_success          | getdeviceinfo.success              | The number of successful NFSv4.1 GETDEVICEINFO operations.      |
| getdeviceinfo_error            | getdeviceinfo.error                | The number of failed NFSv4.1 GETDEVICEINFO operations.          |
| getdeviceinfo_total            | getdeviceinfo.total                | Total number of NFSv4.1 GETDEVICEINFO operations.               |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>           | <b>Description</b>   |
|--------------------------------|-------------------------------|--|
| getdeviceinfo_percent          | getdeviceinfo.percent         | Percentage of NFSv4.1 GETDEVICEINFO operations.            |
| getdeviceinfo_avg_latency      | getdeviceinfo.average_latency | Average latency of NFSv4.1 GETDEVICEINFO operations.       |
| getdevicelist_success          | getdevicelist.success         | The number of successful NFSv4.1 GETDEVICELIST operations. |
| getdevicelist_error            | getdevicelist.error           | The number of failed NFSv4.1 GETDEVICELIST operations.     |
| getdevicelist_total            | getdevicelist.total           | Total number of NFSv4.1 GETDEVICELIST operations.          |
| getdevicelist_percent          | getdevicelist.percent         | Percentage of NFSv4.1 GETDEVICELIST operations.            |
| getdevicelist_avg_latency      | getdevicelist.average_latency | Average latency of NFSv4.1 GETDEVICELIST operations.       |
| layoutcommit_success           | layoutcommit.success          | The number of successful NFSv4.1 LAYOUTCOMMIT operations.  |
| layoutcommit_error             | layoutcommit.error            | The number of failed NFSv4.1 LAYOUTCOMMIT operations.      |
| layoutcommit_total             | layoutcommit.total            | Total number of NFSv4.1 LAYOUTCOMMIT operations.           |
| layoutcommit_percent           | layoutcommit.percent          | Percentage of NFSv4.1 LAYOUTCOMMIT operations.             |
| layoutcommit_avg_latency       | layoutcommit.average_latency  | Average latency of NFSv4.1 LAYOUTCOMMIT operations.        |
| layoutget_success              | layoutget.success             | The number of successful NFSv4.1 LAYOUTGET operations.     |
| layoutget_error                | layoutget.error               | The number of failed NFSv4.1 LAYOUTGET operations.         |
| layoutget_total                | layoutget.total               | Total number of NFSv4.1 LAYOUTGET operations.              |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>             | <b>Description</b>   |
|--------------------------------|---------------------------------|--|
| layoutget_percent              | layoutget.percent               | Percentage of NFSv4.1 LAYOUTGET operations.                  |
| layoutget_avg_latency          | layoutget.average_latency       | Average latency of NFSv4.1 LAYOUTGET operations.             |
| layoutreturn_success           | layoutreturn.success            | The number of successful NFSv4.1 LAYOUTRETURN operations.    |
| layoutreturn_error             | layoutreturn.error              | The number of failed NFSv4.1 LAYOUTRETURN operations.        |
| layoutreturn_total             | layoutreturn.total              | Total number of NFSv4.1 LAYOUTRETURN operations.             |
| layoutreturn_percent           | layoutreturn.percent            | Percentage of NFSv4.1 LAYOUTRETURN operations.               |
| layoutreturn_avg_latency       | layoutreturn.average_latency    | Average latency of NFSv4.1 LAYOUTRETURN operations.          |
| secinfo_no_name_success        | secinfo_no_name.success         | The number of successful NFSv4.1 SECINFO_NO_NAME operations. |
| secinfo_no_name_error          | secinfo_no_name.error           | The number of failed NFSv4.1 SECINFO_NO_NAME operations.     |
| secinfo_no_name_total          | secinfo_no_name.total           | Total number of NFSv4.1 SECINFO_NO_NAME operations.          |
| secinfo_no_name_percent        | secinfo_no_name.percent         | Percentage of NFSv4.1 SECINFO_NO_NAME operations.            |
| secinfo_no_name_avg_latency    | secinfo_no_name.average_latency | Average latency of NFSv4.1 SECINFO_NO_NAME operations.       |
| sequence_success               | sequence.success                | The number of successful NFSv4.1 SEQUENCE operations.        |
| sequence_error                 | sequence.error                  | The number of failed NFSv4.1 SEQUENCE operations.            |
| sequence_total                 | sequence.total                  | Total number of NFSv4.1 SEQUENCE operations.                 |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>          | <b>Description</b>   |
|--------------------------------|------------------------------|--|
| sequence_percent               | sequence.percent             | Percentage of NFSv4.1 SEQUENCE operations.                   |
| sequence_avg_latency           | sequence.average_latency     | Average latency of NFSv4.1 SEQUENCE operations.              |
| set_ssv_success                | set_ssv.success              | The number of successful NFSv4.1 SET_SSV operations.         |
| set_ssv_error                  | set_ssv.error                | The number of failed NFSv4.1 SET_SSV operations.             |
| set_ssv_total                  | set_ssv.total                | Total number of NFSv4.1 SET_SSV operations.                  |
| set_ssv_percent                | set_ssv.percent              | Percentage of NFSv4.1 SET_SSV operations.                    |
| set_ssv_avg_latency            | set_ssv.average_latency      | Average latency of NFSv4.1 SET_SSV operations.               |
| test_stateid_success           | test_stateid.success         | The number of successful NFSv4.1 TEST_STATEID operations.    |
| test_stateid_error             | test_stateid.error           | The number of failed NFSv4.1 TEST_STATEID operations.        |
| test_stateid_total             | test_stateid.total           | Total number of NFSv4.1 TEST_STATEID operations.             |
| test_stateid_percent           | test_stateid.percent         | Percentage of NFSv4.1 TEST_STATEID operations.               |
| test_stateid_avg_latency       | test_stateid.average_latency | Average latency of NFSv4.1 TEST_STATEID operations.          |
| want_delegation_success        | want_delegation.success      | The number of successful NFSv4.1 WANT_DELEGATION operations. |
| want_delegation_error          | want_delegation.error        | The number of failed NFSv4.1 WANT_DELEGATION operations.     |
| want_delegation_total          | want_delegation.total        | Total number of NFSv4.1 WANT_DELEGATION operations.          |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>              | <b>Description</b>  |
|--------------------------------|----------------------------------|---|
| want_delegation_percent        | want_delegation.percent          | Percentage of NFSv4.1 WANT_DELEGATION operations.             |
| want_delegation_avg_latency    | want_delegation.average_latency  | Average latency of NFSv4.1 WANT_DELEGATION operations.        |
| destroy_clientid_success       | destroy_clientid.success         | The number of successful NFSv4.1 DESTROY_CLIENTID operations. |
| destroy_clientid_error         | destroy_clientid.error           | The number of failed NFSv4.1 DESTROY_CLIENTID operations.     |
| destroy_clientid_total         | destroy_clientid.total           | Total number of NFSv4.1 DESTROY_CLIENTID operations.          |
| destroy_clientid_percent       | destroy_clientid.percent         | Percentage of NFSv4.1 DESTROY_CLIENTID operations.            |
| destroy_clientid_avg_latency   | destroy_clientid.average_latency | Average latency of NFSv4.1 DESTROY_CLIENTID operations.       |
| reclaim_complete_success       | reclaim_complete.success         | The number of successful NFSv4.1 RECLAIM_COMPLETE operations. |
| reclaim_complete_error         | reclaim_complete.error           | The number of failed NFSv4.1 RECLAIM_COMPLETE operations.     |
| reclaim_complete_total         | reclaim_complete.total           | Total number of NFSv4.1 RECLAIM_COMPLETE operations.          |
| reclaim_complete_percent       | reclaim_complete.percent         | Percentage of NFSv4.1 RECLAIM_COMPLETE operations.            |
| reclaim_complete_avg_latency   | reclaim_complete.average_latency | Average latency of NFSv4.1 RECLAIM_COMPLETE operations.       |
| nfsv4_1_ops                    | ops                              | Total number of NFSv4.1 operations per second.                |
| getattr_latency_hist           | getattr_latency_histogram        | Histogram of latency for GETATTR operations.                  |
| lookup_latency_hist            | lookup_latency_histogram         | Histogram of latency for LOOKUP operations.                   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>        | <b>Description</b>   |
|--------------------------------|----------------------------|--|
| access_latency_hist            | access_latency_histogram   | Histogram of latency for ACCESS operations.  |
| read_latency_hist              | read_latency_histogram     | Histogram of latency for READ operations.  |
| write_latency_hist             | write_latency_histogram    | Histogram of latency for WRITE operations.   |
| remove_latency_hist            | remove_latency_histogram   | Histogram of latency for REMOVE operations.  |
| nfs41_latency_hist             | total.latency_histogram    | Histogram of latency for NFSv4.1 operations.   |
| nfs41_read_size_histo          | total.read_size_histogram  | Histogram of NFSv4.1 read sizes.   |
| nfs41_write_size_histo         | total.write_size_histogram | Histogram of NFSv4.1 write sizes.  |
| nfs41_read_throughput          | total.read_throughput      | NFSv4.1 read data transfers.   |
| nfs41_write_throughput         | total.write_throughput     | NFSv4.1 write data transfers.  |
| nfs41_throughput               | total.throughput           | NFSv4.1 data transfers.  |
| latency                        | latency                    | Average latency of NFSv4.1 requests. This counter keeps track of the average response time of NFSv4.1 requests.                      |
| total_ops                      | total_ops                  | Total number of NFSv4.1 requests per second.   |
| v42_create_session             | v42.create_session         | Number of successful NFSv4.2 CREATE_SESSION requests per second  |
| v42_total_ops                  | v42.total_ops              | Number of NFSv4.2 requests per second  |
| nfsv41_fileop_max_latency      | fileop_max_latency         | Maximum latency of NFSv4.1 requests. This counter keeps track of the maximum response time of NFSv4.1 requests at any point in time. |



# Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object   | Alias            |
|------------------|------------------|
| svm_nfs_v41:node | svm_nfs_v41_node |

# svm\_nfs\_v42

The svm\_nfs\_v42 table reports activity for the Network File System protocol, version 4.2. This is the file-sharing protocol that implements Parallel NFS (pNFS), used to connect to Network Attached Storage (NAS). The alias name for svm\_nfs\_v42:node is svm\_nfs\_v42\_node.

**Classic Object:** nfsv4\_2

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description          |
|------------------------|---------------|----------------------|
| node_name              | node.name     | Node name            |
| instance_name          | name          | Constituent SVM name |
| svm.id                 | svm.id        | SVM ID               |
| svm.name               | svm.name      | SVM Name             |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>          | <b>Description</b>                                  |
|--------------------------------|------------------------------|---|
| null_total                     | null.total                   | Total number of NFSv4.2 NULL procedures.            |
| null_avg_latency               | null.average_latency         | Average latency of NFSv4.2 NULL procedures.         |
| access_total                   | access.total                 | Total number of NFSv4.2 ACCESS operations.          |
| access_avg_latency             | access.average_latency       | Average latency of NFSv4.2 ACCESS operations.       |
| close_total                    | close.total                  | Total number of NFSv4.2 CLOSE operations.           |
| close_avg_latency              | close.average_latency        | Average latency of NFSv4.2 CLOSE operations.        |
| commit_total                   | commit.total                 | Total number of NFSv4.2 COMMIT operations.          |
| commit_avg_latency             | commit.average_latency       | Average latency of NFSv4.2 COMMIT operations.       |
| create_total                   | create.total                 | Total number of NFSv4.2 CREATE operations.          |
| create_avg_latency             | create.average_latency       | Average latency of NFSv4.2 CREATE operations.       |
| delempurge_total               | delempurge.total             | Total number of NFSv4.2 DELEMPURGE operations.      |
| delempurge_avg_latency         | delempurge.average_latency   | Average latency of NFSv4.2 DELEMPURGE operations.   |
| delempreturn_total             | delempreturn.total           | Total number of NFSv4.2 DELEMPRETURN operations.    |
| delempreturn_avg_latency       | delempreturn.average_latency | Average latency of NFSv4.2 DELEMPRETURN operations. |
| getattr_total                  | getattr.total                | Total number of NFSv4.2 GETATTR operations.         |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>                             |
|--------------------------------|-------------------------|--|
| getattr_avg_latency            | getattr.average_latency | Average latency of NFSv4.2 GETATTR operations. |
| getfh_total                    | getfh.total             | Total number of NFSv4.2 GETFH operations.      |
| getfh_avg_latency              | getfh.average_latency   | Average latency of NFSv4.2 GETFH operations.   |
| link_total                     | link.total              | Total number of NFSv4.2 LINK operations.       |
| link_avg_latency               | link.average_latency    | Average latency of NFSv4.2 LINK operations.    |
| lock_total                     | lock.total              | Total number of NFSv4.2 LOCK operations.       |
| lock_avg_latency               | lock.average_latency    | Average latency of NFSv4.2 LOCK operations.    |
| lockt_total                    | lockt.total             | Total number of NFSv4.2 LOCKT operations.      |
| lockt_avg_latency              | lockt.average_latency   | Average latency of NFSv4.2 LOCKT operations.   |
| locku_total                    | locku.total             | Total number of NFSv4.2 LOCKU operations.      |
| locku_avg_latency              | locku.average_latency   | Average latency of NFSv4.2 LOCKU operations.   |
| lookup_total                   | lookup.total            | Total number of NFSv4.2 LOOKUP operations.     |
| lookup_avg_latency             | lookup.average_latency  | Average latency of NFSv4.2 LOOKUP operations.  |
| lookupp_total                  | lookupp.total           | Total number of NFSv4.2 LOOKUPP operations.    |
| lookupp_avg_latency            | lookupp.average_latency | Average latency of NFSv4.2 LOOKUPP operations. |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>                                    |
|--------------------------------|--------------------------------|---|
| nverify_total                  | nverify.total                  | Total number of NFSv4.2 NVERIFY operations.           |
| nverify_avg_latency            | nverify.average_latency        | Average latency of NFSv4.2 NVERIFY operations.        |
| open_total                     | open.total                     | Total number of NFSv4.2 OPEN operations.              |
| open_avg_latency               | open.average_latency           | Average latency of NFSv4.2 OPEN operations.           |
| openattr_total                 | openattr.total                 | Total number of NFSv4.2 OPENATTR operations.          |
| openattr_avg_latency           | openattr.average_latency       | Average latency of NFSv4.2 OPENATTR operations.       |
| open_downgrade_total           | open_downgrade.total           | Total number of NFSv4.2 OPEN_DOWNGRADE operations.    |
| open_downgrade_avg_latency     | open_downgrade.average_latency | Average latency of NFSv4.2 OPEN_DOWNGRADE operations. |
| putfh_total                    | putfh.total                    | Total number of NFSv4.2 PUTFH operations.             |
| putfh_avg_latency              | putfh.average_latency          | Average latency of NFSv4.2 PUTFH operations.          |
| putpubfh_total                 | putpubfh.total                 | Total number of NFSv4.2 PUTPUBFH operations.          |
| putpubfh_avg_latency           | putpubfh.average_latency       | Average latency of NFSv4.2 PUTPUBFH operations.       |
| putrootfh_total                | putrootfh.total                | Total number of NFSv4.2 PUTROOTFH operations.         |
| putrootfh_avg_latency          | putrootfh.average_latency      | Average latency of NFSv4.2 PUTROOTFH operations.      |
| read_total                     | read.total                     | Total number of NFSv4.2 READ operations.              |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>       | <b>Description</b>                               |
|--------------------------------|---------------------------|--|
| read_avg_latency               | read.average_latency      | Average latency of NFSv4.2 READ operations.      |
| readdir_total                  | readdir.total             | Total number of NFSv4.2 REaddir operations.      |
| readdir_avg_latency            | readdir.average_latency   | Average latency of NFSv4.2 REaddir operations.   |
| readlink_total                 | readlink.total            | Total number of NFSv4.2 READLINK operations.     |
| readlink_avg_latency           | readlink.average_latency  | Average latency of NFSv4.2 READLINK operations.  |
| remove_total                   | remove.total              | Total number of NFSv4.2 REMOVE operations.       |
| remove_avg_latency             | remove.average_latency    | Average latency of NFSv4.2 REMOVE operations.    |
| rename_total                   | rename.total              | Total number of NFSv4.2 RENAME operations.       |
| rename_avg_latency             | rename.average_latency    | Average latency of NFSv4.2 RENAME operations.    |
| restorefh_total                | restorefh.total           | Total number of NFSv4.2 RESTOREFH operations.    |
| restorefh_avg_latency          | restorefh.average_latency | Average latency of NFSv4.2 RESTOREFH operations. |
| savefh_total                   | savefh.total              | Total number of NFSv4.2 SAVEFH operations.       |
| savefh_avg_latency             | savefh.average_latency    | Average latency of NFSv4.2 SAVEFH operations.    |
| secinfo_total                  | secinfo.total             | Total number of NFSv4.2 SECINFO operations.      |
| secinfo_avg_latency            | secinfo.average_latency   | Average latency of NFSv4.2 SECINFO operations.   |

| <b>Classic Numeric Counter</b>   | <b>REST Counter</b>                  | <b>Description</b>  |
|----------------------------------|--------------------------------------|---|
| setattr_total                    | setattr.total                        | Total number of NFSv4.2 SETATTR operations.                 |
| setattr_avg_latency              | setattr.average_latency              | Average latency of NFSv4.2 SETATTR operations.              |
| verify_total                     | verify.total                         | Total number of NFSv4.2 VERIFY operations.                  |
| verify_avg_latency               | verify.average_latency               | Average latency of NFSv4.2 VERIFY operations.               |
| write_total                      | write.total                          | Total number of NFSv4.2 WRITE operations.                   |
| write_avg_latency                | write.average_latency                | Average latency of NFSv4.2 WRITE operations.                |
| backchannel_ctl_total            | backchannel_ctl.total                | Total number of NFSv4.2 BACKCHANNEL_CTL operations.         |
| backchannel_ctl_avg_latency      | backchannel_ctl.average_latency      | Average latency of NFSv4.2 BACKCHANNEL_CTL operations.      |
| bind_conn_to_session_total       | bind_conn_to_session.total           | Total number of NFSv4.2 BIND_CONN_TO_SESSION operations.    |
| bind_conn_to_session_avg_latency | bind_conn_to_session.average_latency | Average latency of NFSv4.2 BIND_CONN_TO_SESSION operations. |
| exchange_id_total                | exchange_id.total                    | Total number of NFSv4.2 EXCHANGE_ID operations.             |
| exchange_id_avg_latency          | exchange_id.average_latency          | Average latency of NFSv4.2 EXCHANGE_ID operations.          |
| create_session_total             | create_session.total                 | Total number of NFSv4.2 CREATE_SESSION operations.          |
| create_session_avg_latency       | create_session.average_latency       | Average latency of NFSv4.2 CREATE_SESSION operations.       |
| destroy_session_total            | destroy_session.total                | Total number of NFSv4.2 DESTROY_SESSION operations.         |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                | <b>Description</b>  |
|--------------------------------|------------------------------------|---|
| destroy_session_avg_latency    | destroy_session.average_latency    | Average latency of NFSv4.2 DESTROY_SESSION operations.    |
| free_stateid_total             | free_stateid.total                 | Total number of NFSv4.2 FREE_STATEID operations.          |
| free_stateid_avg_latency       | free_stateid.average_latency       | Average latency of NFSv4.2 FREE_STATEID operations.       |
| get_dir_delegation_total       | get_dir_delegation.total           | Total number of NFSv4.2 GET_DIR_DELEGATION operations.    |
| get_dir_delegation_avg_latency | get_dir_delegation.average_latency | Average latency of NFSv4.2 GET_DIR_DELEGATION operations. |
| getdeviceinfo_total            | getdeviceinfo.total                | Total number of NFSv4.2 GETDEVICEINFO operations.         |
| getdeviceinfo_avg_latency      | getdeviceinfo.average_latency      | Average latency of NFSv4.2 GETDEVICEINFO operations.      |
| getdevicelist_total            | getdevicelist.total                | Total number of NFSv4.2 GETDEVICELIST operations.         |
| getdevicelist_avg_latency      | getdevicelist.average_latency      | Average latency of NFSv4.2 GETDEVICELIST operations.      |
| layoutcommit_total             | layoutcommit.total                 | Total number of NFSv4.2 LAYOUTCOMMIT operations.          |
| layoutcommit_avg_latency       | layoutcommit.average_latency       | Average latency of NFSv4.2 LAYOUTCOMMIT operations.       |
| layoutget_total                | layoutget.total                    | Total number of NFSv4.2 LAYOUTGET operations.             |
| layoutget_avg_latency          | layoutget.average_latency          | Average latency of NFSv4.2 LAYOUTGET operations.          |
| layoutreturn_total             | layoutreturn.total                 | Total number of NFSv4.2 LAYOUTRETURN operations.          |
| layoutreturn_avg_latency       | layoutreturn.average_latency       | Average latency of NFSv4.2 LAYOUTRETURN operations.       |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>              | <b>Description</b>                                      |
|--------------------------------|----------------------------------|---|
| secinfo_no_name_total          | secinfo_no_name.total            | Total number of NFSv4.2 SECINFO_NO_NAME operations.     |
| secinfo_no_name_avg_latency    | secinfo_no_name.average_latency  | Average latency of NFSv4.2 SECINFO_NO_NAME operations.  |
| sequence_total                 | sequence.total                   | Total number of NFSv4.2 SEQUENCE operations.            |
| sequence_avg_latency           | sequence.average_latency         | Average latency of NFSv4.2 SEQUENCE operations.         |
| set_ssv_total                  | set_ssv.total                    | Total number of NFSv4.2 SET_SSV operations.             |
| set_ssv_avg_latency            | set_ssv.average_latency          | Average latency of NFSv4.2 SET_SSV operations.          |
| test_stateid_total             | test_stateid.total               | Total number of NFSv4.2 TEST_STATEID operations.        |
| test_stateid_avg_latency       | test_stateid.average_latency     | Average latency of NFSv4.2 TEST_STATEID operations.     |
| want_delegation_total          | want_delegation.total            | Total number of NFSv4.2 WANT_DELEGATION operations.     |
| want_delegation_avg_latency    | want_delegation.average_latency  | Average latency of NFSv4.2 WANT_DELEGATION operations.  |
| destroy_clientid_total         | destroy_clientid.total           | Total number of NFSv4.2 DESTROY_CLIENTID operations.    |
| destroy_clientid_avg_latency   | destroy_clientid.average_latency | Average latency of NFSv4.2 DESTROY_CLIENTID operations. |
| reclaim_complete_total         | reclaim_complete.total           | Total number of NFSv4.2 RECLAIM_complete operations.    |
| reclaim_complete_avg_latency   | reclaim_complete.average_latency | Average latency of NFSv4.2 RECLAIM_complete operations. |
| nfs42_read_throughput          | total.read_throughput            | NFSv4.2 read data transfers.                            |
| nfs42_write_throughput         | total.write_throughput           | NFSv4.2 write data transfers.                           |

| Classic Numeric Counter | REST Counter     | Description   |
|-------------------------|------------------|---|
| nfs42_throughput        | total.throughput | NFSv4.2 data transfers.   |
| latency                 | latency          | Average latency of nfsv42 requests. This counter keeps track of the average response time of nfsv42 requests. |
| total_ops               | total_ops        | Total number of nfsv42 requests per sec.  |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object   | Alias            |
|------------------|------------------|
| svm_nfs_v42:node | svm_nfs_v42_node |

# svm\_vscan

Counter Manager (CM) table for exporting per Vserver Offbox Vscan counters. These counters provide the overall health of the Vscan feature.

**Classic Object:** offbox\_vscan

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment  |
|---|------------------|--|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.                             |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description          |
|------------------------|---------------|----------------------|
| node_name              | node.name     | System node name     |
| instance_name          | name          | Constituent SVM name |
| vserver_id             | svm.id        | Vserver ID           |
| svm.name               | svm.name      | Vserver Name         |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter      | Description  |
|-------------------------|-------------------|--|
| dispatch_latency_base   | dispatch.requests | Total number of requests to be used as a base counter for average dispatch latency calculation |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>             | <b>Description</b>   |
|--------------------------------|---------------------------------|--|
| dispatch_latency               | dispatch.latency                | Average dispatch latency   |
| scan_latency_base              | scan.requests                   | Total number of requests to be used as a base counter for average scan latency calculation |
| scan_latency                   | scan.latency                    | Average scan latency   |
| scan_noti_received_rate        | scan.notification_received_rate | Total number of scan notifications received by the dispatcher per second                   |
| scan_request_dispatched_rate   | scan.request_dispatched_rate    | Total number of scan requests sent to the Vscanner per second                              |
| connections_active             | connections_active              | Total number of current active connections   |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# system

The System table reports general system activity. This includes global throughput for the main services, I/O latency, and CPU activity. The alias name for system:node is system\_node.

**Classic Object:** system

## Table Row IDs

| ID Format                                   | Aggregation Type | Comment   |
|---|------------------|---|
| {instance_name}:{instance_uuid}             | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{instance_name}:{instance_uuid} | constituent      | This represents the construction of a row ID field for a row from a single node.  |
| {node.name}:{node.uuid}                     | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                              |
|------------------------|---------------|--|
| node_name              | node.name     | Node name                                |
| instance_name          | name          | This is the name of the system instance. |
| instance_uuid          | uuid          | This is the uuid of the system instance. |
| system_model           | system_model  | Name of the system model                 |
| ontap_version          | ontap_version | ONTAP version                            |
| compile_flags          | compile_flags | ONTAP compile flags                      |
| serial_no              | serial_no     | System serial number                     |

| Classic String Counter | REST Property | Description |
|------------------------|---------------|-------------|
| system_id              | system_id     | System ID   |
| hostname               | hostname      | Hostname    |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter          | Description  |
|-------------------------|-----------------------|--|
| memory                  | memory                | Total memory in megabytes (MB)                       |
| nfs_ops                 | nfs_ops               | Number of NFS operations per second                  |
| cifs_ops                | cifs_ops              | Number of CIFS operations per second                 |
| fcp_ops                 | fcp_ops               | Number of FCP operations per second                  |
| iscsi_ops               | iscsi_ops             | Number of iSCSI operations per second                |
| nvme_fc_ops             | nvme_fc_ops           | NVMe/FC operations per second                        |
| nvme_tcp_ops            | nvme_tcp_ops          | NVMe/TCP operations per second                       |
| nvme_roce_ops           | nvme_roce_ops         | NVMe/RoCE operations per second                      |
| net_data_recv           | network_data_received | Number of network kilobytes (KB) received per second |
| net_data_sent           | network_data_sent     | Number of network kilobytes (KB) sent per second     |
| fcp_data_recv           | fcp_data_received     | Number of FCP kilobytes (KB) received per second     |
| fcp_data_sent           | fcp_data_sent         | Number of FCP kilobytes (KB) sent per second         |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>     | <b>Description</b>  |
|--------------------------------|-------------------------|---|
| iscsi_data_received            | iscsi_data_received     | iSCSI kilobytes (KB) received per second                    |
| iscsi_data_sent                | iscsi_data_sent         | iSCSI kilobytes (KB) sent per second                        |
| nvme_fc_data_received          | nvme_fc_data_received   | NVMe/FC kilobytes (KB) received per second                  |
| nvme_fc_data_sent              | nvme_fc_data_sent       | NVMe/FC kilobytes (KB) sent per second                      |
| nvme_tcp_data_received         | nvme_tcp_data_received  | NVMe/TCP kilobytes (KB) received per second                 |
| nvme_tcp_data_sent             | nvme_tcp_data_sent      | NVMe/TCP kilobytes (KB) sent per second                     |
| nvme_roce_data_received        | nvme_roce_data_received | NVMe/RoCE kilobytes (KB) received per second                |
| nvme_roce_data_sent            | nvme_roce_data_sent     | NVMe/RoCE kilobytes (KB) sent per second                    |
| partner_data_received          | partner_data_received   | SCSI Partner kilobytes (KB) received per second             |
| partner_data_sent              | partner_data_sent       | SCSI Partner kilobytes (KB) sent per second                 |
| sys_read_data                  | sys_read_data           | Network and FCP kilobytes (KB) received per second          |
| sys_write_data                 | sys_write_data          | Network and FCP kilobytes (KB) sent per second              |
| sys_total_data                 | sys_total_data          | Network and FCP kilobytes (KB) received and sent per second |
| disk_data_read                 | disk_data_read          | Number of disk kilobytes (KB) read per second               |
| disk_data_written              | disk_data_written       | Number of disk kilobytes (KB) written per second            |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b> | <b>Description</b>   |
|--------------------------------|---------------------|--|
| hdd_data_read                  | hdd_data_read       | Number of HDD Disk kilobytes (KB) read per second                      |
| hdd_data_written               | hdd_data_written    | Number of HDD kilobytes (KB) written per second                        |
| ssd_data_read                  | ssd_data_read       | Number of SSD Disk kilobytes (KB) read per second                      |
| ssd_data_written               | ssd_data_written    | Number of SSD Disk kilobytes (KB) written per second                   |
| tape_data_read                 | tape_data_read      | Tape bytes read per millisecond  |
| tape_data_written              | tape_data_written   | Tape bytes written per millisecond                                     |
| read_ops                       | read_ops            | Read operations per second   |
| write_ops                      | write_ops           | Write operations per second  |
| other_ops                      | other_ops           | All other operations per second  |
| total_ops                      | total_ops           | Total number of operations per second                                  |
| read_latency                   | read_latency        | Average latency for all read operations in the system in microseconds  |
| write_latency                  | write_latency       | Average latency for all write operations in the system in microseconds |
| other_latency                  | other_latency       | Average latency for all other operations in the system in microseconds |
| total_latency                  | total_latency       | Average latency for all operations in the system in microseconds       |
| read_data                      | read_data           | Read throughput  |
| write_data                     | write_data          | Write throughput   |
| other_data                     | other_data          | Other throughput   |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>   |
|--------------------------------|--------------------------------|--|
| total_data                     | total_data                     | Total throughput in bytes  |
| cpu_busy                       | cpu_busy                       | System CPU resource utilization. Returns a computed percentage for the default CPU field. Basically computes a 'cpu usage summary' value which indicates how 'busy' the system is based upon the most heavily utilized domain. The idea is to determine the amount of available CPU until we're limited by either a domain maxing out OR we exhaust all available idle CPU cycles, whichever occurs first. |
| cpu_elapsed_time               | cpu_elapsed_time               | Elapsed time since boot  |
| avg_processor_busy             | average_processor_busy_percent | Average processor utilization across active processors in the system   |
| total_processor_busy           | total_processor_busy           | Total processor utilization of active processors in the system   |
| total_processor_busy_time      | total_processor_busy_time      | Total processor time of active processors in the system  |
| num_processors                 | num_processors                 | Total number of processors in the system   |
| interrupt_time                 | interrupt_time                 | Processor interrupt time   |
| interrupt                      | interrupt                      | Processor interrupt rate percentage  |
| interrupt_num                  | interrupt_num                  | Processor interrupt number   |
| time_per_interrupt             | time_per_interrupt             | Processor time per interrupt   |
| non_interrupt_time             | non_interrupt_time             | Processor non-interrupt time   |
| non_interrupt                  | non_interrupt                  | Processor non-interrupt rate percentage  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>  |
|--------------------------------|--------------------------|---|
| idle_time                      | idle_time                | Processor idle time   |
| idle                           | idle                     | Processor idle rate percentage  |
| cp_time                        | cp_time                  | Processor time in CP  |
| cp                             | cp                       | CP time rate  |
| interrupt_in_cp_time           | interrupt_in_cp_time     | Processor interrupt in CP time  |
| interrupt_in_cp                | interrupt_in_cp          | Processor interrupt rate percentage                                   |
| interrupt_num_in_cp            | interrupt_num_in_cp      | Number of processor interrupts in CP                                  |
| time_per_interrupt_in_cp       | time_per_interrupt_in_cp | Processor time per interrupt in CP                                    |
| processor_plevel_time          | processor_plevel_time    | Processor plevel rate percentage                                      |
| processor_plevel               | processor_plevel         | Processor plevel rate percentage                                      |
| domain_busy                    | domain_busy              | Array of processor time in percentage spent in various domains        |
| sk_switches                    | sk_switches              | Number of sk switches per second                                      |
| hard_switches                  | hard_switches            | Number of context switches per second                                 |
| domain_shared                  | domain_shared            | Array of processor time in percentage spent in various shared domains |
| dswichto_cnt                   | dswichto_cnt             | Array of processor time in percentage spent in domain switch          |
| intr_cnt                       | intr_cnt                 | Array of interrupt count per second                                   |
| intr_cnt_msec                  | intr_cnt_msec            | Millisecond interrupt count per second                                |
| intr_cnt_ipi                   | intr_cnt_ipi             | IPI interrupt count per second  |

| Classic Numeric Counter | REST Counter   | Description   |
|-------------------------|----------------|---|
| intr_cnt_total          | intr_cnt_total | Total interrupt count per second                              |
| time                    | time           | Time in seconds since the Epoch (00:00:00 UTC January 1 1970) |
| uptime                  | uptime         | Time in seconds that the system has been up                   |
| wafiron                 | wafiron        | Wafiron counters  |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object | Alias       |
|----------------|-------------|
| system:node    | system_node |

# token\_manager

A token represents the abstraction of data and is associated with a point-in-time (PIT) copy. A token is created by the SAN or SMB protocols by using protocol commands. Token Manager creates and copies tokens as requested by hosts for token-based copy-offload requests. In token-based copy-offload requests, the host sends an OFFLOAD READ command with the offsets to the source file or LUN. Token Manager creates a PIT copy of the offsets using sis-clone from source to PIT file and generates a token. This token is given to the host. The host then sends this token with the offsets to the destination file or LUN using an OFFLOAD WRITE cmd. Token Manager decodes the token, finds the data represented by the token and copies from the PIT file to the destination file or LUN. This completes a single token-based copy-offload request. This counter manager table tracks the number of token-based copy-offload requests received (both OFFLOAD READ and OFFLOAD WRITE), their sizes and their latencies. It also tracks various kind of errors that could have happened during token-based copy-offload requests.

**Classic Object:** token\_manager

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description      |
|------------------------|---------------|------------------|
| node_name              | node.name     | System node name |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter       | Description   |
|-------------------------|--------------------|---|
| token_create_bytes      | token_create.bytes | Total number of bytes for which tokens are created. |

| Classic Numeric Counter | REST Counter           | Description                                 |
|-------------------------|------------------------|---|
| token_copy_bytes        | token_copy.bytes       | Total number of bytes copied.               |
| token_zero_bytes        | token_zero.bytes       | Total number of bytes zeroed.               |
| token_create_success    | token_create.successes | Number of successful token create requests. |
| token_create_failure    | token_create.failures  | Number of failed token create requests.     |
| token_copy_success      | token_copy.successes   | Number of successful token copy requests.   |
| token_copy_failure      | token_copy.failures    | Number of failed token copy requests.       |
| token_zero_success      | token_zero.successes   | Number of successful token zero requests.   |
| token_zero_failure      | token_zero.failures    | Number of failed token zero requests.       |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# volume

Counter Manager table for exporting volume performance counters. The alias names for volume:node and volume:svm are volume\_node and volume\_svm respectively.

**Classic Object:** volume

## Table Row IDs

| ID Format                                  | Aggregation Type | Comment   |
|--|------------------|---|
| {svm.name}:{instance_name}:{instance_uuid} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row.                      |
| {node.name}:{node.uuid}                    | node             | This represents the construction of the row ID field in node combo object, which is a single unique string that identifies a row. |
| {svm.name}:{svm.uuid}                      | svm              | This represents the construction of the row ID field in svm combo object, which is a single unique string that identifies a row.  |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description                                |
|------------------------|---------------|--|
| node_name              | node.name     | Name of the Node                           |
| node_uuid              | node.uuid     | UUID for the node                          |
| instance_name          | name          | This is the name of the volume counter row |
| instance_uuid          | uuid          | DSID/MSID of this volume row               |
| vserver_name           | svm.name      | Name of the Copy Manager SVM row           |
| vserver_uuid           | svm.uuid      | UUID for the Copy Manager SVM row          |

| Classic String Counter | REST Property    | Description                                   |
|------------------------|------------------|---|
| parent_aggr            | parent_aggregate | The name of the aggregate hosting this volume |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter    | Description   |
|-------------------------|-----------------|---|
| read_data               | bytes_read      | Bytes read per second   |
| write_data              | bytes_written   | Bytes written per second  |
| read_ops                | total_read_ops  | Number of read operations per second from the volume  |
| write_ops               | total_write_ops | Number of write operations per second to the volume   |
| other_ops               | total_other_ops | Number of other operations per second to the volume   |
| total_ops               | total_ops       | Number of operations per second serviced by the volume  |
| read_latency            | read_latency    | Average latency in microseconds for the WAFL filesystem to process read request to the volume; not including request processing or network communication time     |
| write_latency           | write_latency   | Average latency in microseconds for the WAFL filesystem to process write request to the volume; not including request processing or network communication time    |
| other_latency           | other_latency   | Average latency in microseconds for the WAFL filesystem to process other operations to the volume; not including request processing or network communication time |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b> | <b>Description</b>  |
|--------------------------------|---------------------|---|
| avg_latency                    | average_latency     | Average latency in microseconds for the WAFL filesystem to process all the operations on the volume; not including request processing or network communication time   |
| cifs_read_data                 | cifs.read_data      | Bytes read per second via CIFS  |
| cifs_read_latency              | cifs.read_latency   | Average time for the WAFL filesystem to process CIFS read requests to the volume; not including CIFS protocol request processing or network communication time which will also be included in client observed CIFS request latency    |
| cifs_read_ops                  | cifs.read_ops       | Number of CIFS read operations per second from the volume   |
| cifs_write_data                | cifs.write_data     | Bytes written per second via CIFS   |
| cifs_write_latency             | cifs.write_latency  | Average time for the WAFL filesystem to process CIFS write requests to the volume; not including CIFS protocol request processing or network communication time which will also be included in client observed CIFS request latency   |
| cifs_write_ops                 | cifs.write_ops      | Number of CIFS write operations per second to the volume  |
| cifs_other_latency             | cifs.other_latency  | Average time for the WAFL filesystem to process other CIFS operations to the volume; not including CIFS protocol request processing or network communication time which will also be included in client observed CIFS request latency |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b> | <b>Description</b>   |
|--------------------------------|---------------------|--|
| cifs_other_ops                 | cifs.other_ops      | Number of other CIFS operations per second to the volume   |
| nfs_read_data                  | nfs.read_data       | Bytes read per second via NFS  |
| nfs_read_latency               | nfs.read_latency    | Average time for the WAFL filesystem to process NFS protocol read requests to the volume; not including NFS protocol request processing or network communication time which will also be included in client observed NFS request latency   |
| nfs_read_ops                   | nfs.read_ops        | Number of NFS read operations per second from the volume   |
| nfs_write_data                 | nfs.write_data      | Bytes written per second via NFS   |
| nfs_write_latency              | nfs.write_latency   | Average time for the WAFL filesystem to process NFS protocol write requests to the volume; not including NFS protocol request processing or network communication time, which will also be included in client observed NFS request latency |
| nfs_write_ops                  | nfs.write_ops       | Number of NFS write operations per second to the volume  |
| nfs_other_latency              | nfs.other_latency   | Average time for the WAFL filesystem to process other NFS operations to the volume; not including NFS protocol request processing or network communication time which will also be included in client observed NFS request latency         |
| nfs_other_ops                  | nfs.other_ops       | Number of other NFS operations per second to the volume  |

| Classic Numeric Counter | REST Counter        | Description  |
|-------------------------|---------------------|--|
| iscsi_read_data         | iscsi.read_data     | Bytes read per second via block protocol   |
| iscsi_read_latency      | iscsi.read_latency  | Average time for the WAFL filesystem to process iSCSI protocol read operations to the volume; not including iSCSI protocol request processing or network communication time which will also be included in client observed iSCSI protocol request latency  |
| iscsi_read_ops          | iscsi.read_ops      | Number of block protocol read operations per second from the volume  |
| iscsi_write_data        | iscsi.write_data    | Bytes written per second via block protocol  |
| iscsi_write_latency     | iscsi.write_latency | Average time for the WAFL filesystem to process iSCSI protocol write operations to the volume; not including iSCSI protocol request processing or network communication time which will also be included in client observed iSCSI request latency          |
| iscsi_write_ops         | iscsi.write_ops     | Number of block protocol write operations per second to the volume   |
| iscsi_other_latency     | iscsi.other_latency | Average time for the WAFL filesystem to process other iSCSI protocol operations to the volume; not including iSCSI protocol request processing or network communication time which will also be included in client observed iSCSI protocol request latency |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b> | <b>Description</b>   |
|--------------------------------|---------------------|--|
| iscsi_other_ops                | iscsi.other_ops     | Number of other block protocol operations per second to the volume   |
| fcp_read_data                  | fcp.read_data       | Bytes read per second via block protocol   |
| fcp_read_latency               | fcp.read_latency    | Average time for the WAFL filesystem to process FCP protocol read operations to the volume; not including FCP protocol request processing or network communication time which will also be included in client observed FCP protocol request latency  |
| fcp_read_ops                   | fcp.read_ops        | Number of block protocol read operations per second from the volume  |
| fcp_write_data                 | fcp.write_data      | Bytes written per second via block protocol  |
| fcp_write_latency              | fcp.write_latency   | Average time for the WAFL filesystem to process FCP protocol write operations to the volume; not including FCP protocol request processing or network communication time which will also be included in client observed FCP protocol request latency |
| fcp_write_ops                  | fcp.write_ops       | Number of block protocol write operations per second to the volume   |

| Classic Numeric Counter        | REST Counter                    | Description  |
|--------------------------------|---------------------------------|--|
| fcg_other_latency              | fcg.other_latency               | Average time for the WAFL filesystem to process other FCG protocol operations to the volume; not including FCG protocol request processing or network communication time which will also be included in client observed FCG protocol request latency |
| fcg_other_ops                  | fcg.other_ops                   | Number of other block protocol operations per second to the volume   |
| nfs_protocol_read_latency      | nfs.read_latency_histogram      | Latency histograms for read requests from the NFS protocol to the volume in the WAFL filesystem.   |
| nfs_protocol_write_histo       | nfs.write_histogram             | Write granularity histogram for write requests from NFS to the volume in the WAFL filesystem.  |
| nfs_protocol_misaligned_writes | nfs.misaligned_writes_histogram | Mis-aligned writes for specific sized write requests from NFS to the volume in the WAFL filesystem. The mis-alignment is reported only for sizes which are a power of 2.   |
| nfs_protocol_write_latency     | nfs.write_latency_histogram     | Latency histograms for write requests from the NFS protocol to the volume in the WAFL filesystem.  |
| nfs_getattr_latency            | nfs.getattr_latency             | Average time for the WAFL filesystem to process NFS protocol getattr requests to the volume; not including NFS protocol request processing or network communication time which will also be included in client observed NFS request latency.         |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>           | <b>Description</b>  |
|--------------------------------|-------------------------------|---|
| nfs_protocol_getattr_latency   | nfs.getattr_latency_histogram | Latency histograms for getattr requests from the NFS protocol to the volume in the WAFL filesystem.   |
| nfs_getattr_ops                | nfs.getattr_ops               | Number of NFS getattr per second to the volume.   |
| nfs_setattr_ops                | nfs.setattr_ops               | Number of NFS setattr requests per second to the volume.  |
| nfs_setattr_latency            | nfs.setattr_latency           | Average time for the WAFL filesystem to process NFS protocol setattr requests to the volume.  |
| nfs_protocol_setattr_latency   | nfs.setattr_latency_histogram | Latency histograms for setattr requests from the NFS protocol to the volume in the WAFL filesystem.   |
| nfs_lookup_latency             | nfs.lookup_latency            | Average time for the WAFL filesystem to process NFS protocol lookup requests to the volume; not including NFS protocol request processing or network communication time which will also be included in client observed NFS request latency. |
| nfs_protocol_lookup_latency    | nfs.lookup_latency_histogram  | Latency histograms for lookup requests from the NFS protocol to the volume in the WAFL filesystem.  |
| nfs_lookup_ops                 | nfs.lookup_ops                | Number of NFS lookups per second to the volume.   |

| <b>Classic Numeric Counter</b>  | <b>REST Counter</b>              | <b>Description</b>  |
|---------------------------------|----------------------------------|---|
| nfs_access_latency              | nfs.access_latency               | Average time for the WAFL filesystem to process NFS protocol access requests to the volume; not including NFS protocol request processing or network communication time which will also be included in client observed NFS request latency. |
| nfs_protocol_access_latency     | nfs.access_latency_histogram     | Latency histograms for access requests from the NFS protocol to the volume in the WAFL filesystem.  |
| nfs_access_ops                  | nfs.access_ops                   | Number of NFS accesses per second to the volume.  |
| nfs_punch_hole_data             | nfs.punch_hole_data              | Bytes hole-punched per second via NFS to the volume.  |
| nfs_punch_hole_ops              | nfs.punch_hole_ops               | Number of NFS hole-punch requests per second to the volume.   |
| nfs_punch_hole_latency          | nfs.punch_hole_latency           | Average time for the WAFL filesystem to process NFS protocol hole-punch requests to the volume.   |
| nfs_protocol_punch_hole_histo   | nfs.punch_hole_histogram         | Granularity histogram for hole-punch requests from NFS to the volume in the WAFL filesystem.  |
| nfs_protocol_punch_hole_latency | nfs.punch_hole_latency_histogram | Latency histograms for hole-punch requests from the NFS protocol to the volume in the WAFL filesystem.  |
| nfs_protocol_other_latency      | nfs.other_latency_histogram      | Latency histograms for other requests from the NFS protocol to the volume in the WAFL filesystem.   |
| nfs_total_ops                   | nfs.total_ops                    | Number of total NFS operations per second to the volume.  |

# Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter                    | Description   |
|---------------------------------|---|
| nfs.write_histogram             | For each of the labels in this histogram, all white spaces have been removed.   |
| nfs.misaligned_writes_histogram | For each of the labels in this histogram, whitespace is changed to "_" and uppercase letters were changed to lowercase letters. |
| nfs.punch_hole_histogram        | For each of the labels in this histogram, all white spaces have been removed.   |

## Table Aliases

This section describes aliases for aggregated tables.

| Classic Object | Alias       |
|----------------|-------------|
| volume:node    | volume_node |
| volume:svm     | volume_svm  |

# vscan

Counter Manager (CM) table for exporting Offbox Vscan connection status counters. Offbox-scanners maintain one connection per vserver per node of cluster. This table captures statistics around every such connection

**Classic Object:** offbox\_vscan\_server

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description  |
|------------------------|---------------|--|
| node_name              | node.name     | System node name   |
| server_id              | server.id     | Ip Address of the vscan server along with the SVM it belongs to in the format 'SVM:IP' |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter      | REST Counter                 | Description  |
|------------------------------|------------------------------|--|
| scan_latency_base            | scan.requests                | Total number of requests to be used as a base counter for average scan latency calculation |
| scan_latency                 | scan.latency                 | Average scan latency   |
| scan_request_dispatched_rate | scan.request_dispatched_rate | Total number of scan requests sent to the scanner per second                               |



| Classic Numeric Counter        | REST Counter                       | Description   |
|--------------------------------|------------------------------------|---|
| scanner_stats_pct_cpu_used     | scanner.stats_percent_cpu_used     | Percentage CPU utilization on scanner calculated over the last 15 seconds.    |
| scanner_stats_pct_mem_used     | scanner.stats_percent_mem_used     | Percentage RAM utilization on scanner calculated over the last 15 seconds.    |
| scanner_stats_pct_network_used | scanner.stats_percent_network_used | Percentage network utilization on scanner calculated for the last 15 seconds. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

## Table Aliases

This section describes aliases for aggregated tables.

# waf1

Counter Manager object for exporting WAFL performance counters

Classic Object: waf1

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property | Description           |
|------------------------|---------------|-----------------------|
| node_name              | node.name     | Node Name             |
| instance_name          | name          | Name of this instance |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter   | REST Counter                    | Description  |
|---------------------------|---------------------------------|--|
| avg_non_waf1_msg_latency  | average_non_waf1_msg_latency    | Average turnaround time for non-WAFL messages in milliseconds.         |
| avg_waf1_msg_latency      | average_msg_latency             | Average turnaround time for WAFL messages in milliseconds.             |
| avg_waf1_repl_msg_latency | average_replication_msg_latency | Average turnaround time for replication WAFL messages in milliseconds. |
| waf1_msg_total            | msg_total                       | Total number of WAFL messages per second.                              |
| waf1_repl_msg_total       | replication_msg_total           | Total number of replication WAFL messages per second.                  |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>       | <b>Description</b>  |
|--------------------------------|---------------------------|---|
| non_wafl_msg_total             | non_wafl_msg_total        | Total number of non-WAFL messages per second.                                 |
| cp_count                       | cp_count                  | Array of counts of different types of Consistency Points (CP).                |
| cp_phase_times                 | cp_phase_times            | Array of percentage time spent in different phases of Consistency Point (CP). |
| read_io_type                   | read_io_type              | Percentage of reads served from buffer cache, external cache, or disk.        |
| read_io_type_base              | read_io_type_base         | Total number of WAFL reads. Base for read_io_types.                           |
| total_cp_msecs                 | total_cp_msecs            | Milliseconds spent in Consistency Point (CP).                                 |
| total_cp_util                  | total_cp_util             | Percentage of time spent in a Consistency Point (CP).                         |
| cpu_elapsed_time               | cpu_elapsed_time          | Elapsed time since boot.  |
| wafl_memory_used               | memory_used               | The current WAFL memory used in the system.                                   |
| wafl_memory_free               | memory_free               | The current WAFL memory available in the system.                              |
| wafl_reads_from_cache          | reads_from_cache          | WAFL reads from cache.  |
| wafl_reads_from_cloud          | reads_from_cloud          | WAFL reads from cloud storage.  |
| wafl_reads_from_cloud_s2c_bin  | reads_from_cloud_s2c_bin  | WAFL reads from cloud storage via s2c bin.                                    |
| wafl_reads_from_disk           | reads_from_disk           | WAFL reads from disk.   |
| wafl_reads_from_ext_cache      | reads_from_external_cache | WAFL reads from external cache.   |
| wafl_reads_from_fc_miss        | reads_from_fc_miss        | WAFL reads from remote volume for fc_miss.                                    |

| Classic Numeric Counter | REST Counter                 | Description                        |
|-------------------------|------------------------------|------------------------------------|
| wafldb_reads_from_pmem  | reads_from_persistent_memory | WAFL reads from persistent memory. |
| wafldb_reads_from_ssd   | reads_from_ssd               | WAFL reads from SSD.               |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter   | Description  |
|----------------|--|
| cp_count       | Two things were changed about the naming of these array labels:<br>1. All of the spaces and hyphens were converted to underscores.<br>2. All of the letters were converted to lowercase. |
| cp_phase_times | The only difference in the naming for the array labels is that all of the letters were converted to lowercase.   |

## Table Aliases

This section describes aliases for aggregated tables.

# waf\_comp\_aggr\_bin

CM object for exporting Composite Aggregate bin statistics.

**Classic Object:** waf\_comp\_aggr\_bin

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property     | Description                             |
|------------------------|-------------------|---|
| node_name              | node.name         | System node name                        |
| comp_aggr_name         | aggregate.name    | Name of the Composite Aggregate.        |
| bin_uuid               | bin.uuid          | UUID of this bin.                       |
| object_store_name      | cloud_target.name | Name of the object store configuration  |
| object_store_uuid      | cloud_target.uuid | UUID of the object store configuration. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter | Description                   |
|-------------------------|--------------|-------------------------------|
| bin_id                  | bin.id       | Bin ID (0..7)                 |
| config_id               | config_id    | Config ID of the object store |
| cloud_bin_operation     | cloud_bin_op | Cloud bin operation counters. |

| Classic Numeric Counter        | REST Counter                   | Description  |
|--------------------------------|--------------------------------|--|
| cloud_bin_op_size              | cloud_bin_op_size              | Cloud bin operation size counters in bytes.            |
| cloud_bin_op_latency_average   | cloud_bin_op_latency_average   | Cloud bin operation latency average in milliseconds.   |
| cloud_bin_op_latency_histogram | cloud_bin_op_latency_histogram | Cloud bin operation latency histogram in milliseconds. |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter                   | Description  |
|--------------------------------|--|
| cloud_bin_op                   | All of the letters were converted to lowercase while changing the naming of these array labels.  |
| cloud_bin_op_size              | All of the letters were converted to lowercase while changing the naming of these array labels.  |
| cloud_bin_op_latency_average   | All of the letters were converted to lowercase while changing the naming of these array labels.  |
| cloud_bin_op_latency_histogram | <ol style="list-style-type: none"> <li>1. All of the letters were converted to lowercase while changing the naming of these array labels.</li> <li>2. The labels have changed from "0 - &lt;1ms" to "0- &lt;1ms".</li> </ol> |

## Table Aliases

This section describes aliases for aggregated tables.

# waf comp\_aggr\_vol\_bin

CM object for exporting Composite Aggregate bin volume statistics.

**Classic Object:** waf comp\_aggr\_vol\_bin

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property     | Description                             |
|------------------------|-------------------|---|
| node_name              | node.name         | System node name                        |
| comp_aggr_name         | aggregate.name    | Name of the Composite Aggregate.        |
| vserver_name           | svm.name          | Name of the SVM that owns this volume   |
| vserver_uuid           | svm.uuid          | UUID for the SVM that owns this volume  |
| vol_name               | volume.name       | Name of the FlexVol volume.             |
| vol_uuid               | volume.uuid       | UUID of the FlexVol volume.             |
| bin_uuid               | bin.uuid          | UUID of this bin.                       |
| object_store_name      | cloud_target.name | Name of the object store configuration  |
| object_store_uuid      | cloud_target.uuid | UUID of the object store configuration. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| <b>Classic Numeric Counter</b>         | <b>REST Counter</b>                        | <b>Description</b>  |
|--|--|---|
| bin_id                                 | bin.id                                     | Bin ID (0..7)   |
| config_id                              | config_id                                  | Config ID of the object store                                       |
| cloud_bin_space                        | cloud_bin_space                            | Cloud bin capacity counters.  |
| cloud_bin_operation                    | cloud_bin_op                               | Cloud bin operation counters.                                       |
| cloud_bin_op_size                      | cloud_bin_op_size                          | Cloud bin operation sizes in bytes.                                 |
| cloud_bin_op_latency_average           | cloud_bin_op_latency_average               | Cloud bin operation latency average in milliseconds.                |
| bin_error_io_timedout                  | error_io_timedout                          | Number of cloud I/Os that are timed out.                            |
| bin_tlog_reads                         | transfer_log_reads                         | Total reads from transfer log.                                      |
| bin_reads                              | reads                                      | Total number of blocks read.  |
| bin_ra_reads                           | readahead_reads                            | Total number of blocks read via readahead.                          |
| bin_client_proto_reads                 | client_protocol_reads                      | Total number of blocks read via client protocols.                   |
| bin_repl_proto_reads                   | replication_protocol_reads                 | Total number of blocks read via replication protocol.               |
| bin_other_proto_reads                  | other_protocol_reads                       | Total number of blocks read via other protocols.                    |
| bin_read_requests                      | read_requests                              | Total number of cloud IO read requests.                             |
| bin_client_read_requests               | client_read_requests                       | Total number of cloud IO read requests via client protocols.        |
| bin_error_op_expired_before_io         | error_operations_expired_before_i<br>o     | Number of operations expired before cloud I/Os are issued.          |
| bin_error_op_soon_expire_before_i<br>o | error_operations_soon_expire_befo<br>re_io | Number of operations that will expire before cloud I/Os are issued. |



# Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter                 | Description  |
|------------------------------|--|
| cloud_bin_op                 | Two things were changed about the naming of these array labels:<br>1. All of the spaces and hyphens were converted to underscores.<br>2. All of the letters were converted to lowercase. |
| cloud_bin_op_size            | Two things were changed about the naming of these array labels:<br>1. All of the spaces and hyphens were converted to underscores.<br>2. All of the letters were converted to lowercase. |
| cloud_bin_op_latency_average | Two things were changed about the naming of these array labels:<br>1. All of the spaces and hyphens were converted to underscores.<br>2. All of the letters were converted to lowercase. |

## Table Aliases

This section describes aliases for aggregated tables.

# wafI\_hya\_per\_aggregate

CM object for exporting Flash Pool per-aggregate statistics.

**Classic Object:** wafI\_hya\_per\_aggr

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property      | Description                        |
|------------------------|--------------------|------------------------------------|
| node_name              | node.name          | System node name                   |
| hya_aggr_name          | hya_aggregate_name | Name of the Flash Pool.            |
| vol_name               | volume.name        | Name of the volume.                |
| vserver_name           | svm.name           | Name of the SVM owning this volume |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter         | Description                        |
|-------------------------|----------------------|------------------------------------|
| ssd_total               | ssd_total            | Total SSD blocks.                  |
| ssd_total_used          | ssd_total_used       | Total SSD blocks used.             |
| ssd_available           | ssd_available        | Total SSD blocks available.        |
| ssd_direct_allocated    | ssd_direct_allocated | Total SSD blocks direct allocated. |
| ssd_read_cached         | ssd_read_cached      | Total read cached SSD blocks.      |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>             | <b>Description</b>  |
|--------------------------------|---------------------------------|---|
| ssd_write_cached               | ssd_write_cached                | Total write cached SSD blocks.                                  |
| hya_read_hit_latency_hist      | hya_read_hit_latency_histogram  | Flash Pool RAID I/O read latency with cache hit.                |
| hya_read_hit_latency_count     | hya_read_hit_latency_count      | Total read hit latency samples collected.                       |
| hya_read_hit_latency_average   | hya_read_hit_latency_average    | Average of RAID I/O latency on read hit.                        |
| hya_read_miss_latency_hist     | hya_read_miss_latency_histogram | Flash Pool RAID I/O read latency with a cache miss.             |
| hya_read_miss_latency_count    | hya_read_miss_latency_count     | Total read miss latency samples collected.                      |
| hya_read_miss_latency_average  | hya_read_miss_latency_average   | Average read miss latency.                                      |
| hya_write_ssd_latency_hist     | hya_write_ssd_latency_histogram | Flash Pool raidio write latency to SSD.                         |
| hya_write_ssd_latency_count    | hya_write_ssd_latency_count     | Total write SSD latency samples collected.                      |
| hya_write_ssd_latency_average  | hya_write_ssd_latency_average   | Average of RAID I/O latency on write to SSD.                    |
| hya_write_hdd_latency_hist     | hya_write_hdd_latency_histogram | Flash Pool raidio write latency to HDD.                         |
| hya_write_hdd_latency_count    | hya_write_hdd_latency_count     | Total write HDD latency samples collected.                      |
| hya_write_hdd_latency_average  | hya_write_hdd_latency_average   | Average write latency to HDD.                                   |
| read_ops_replaced              | read_ops_replaced               | Number of HDD read operations replaced by SSD reads per second. |
| read_ops_total                 | read_ops_total                  | Number of non-Flash Pool metafile read operations per second.   |
| read_ops_replaced_percent      | read_ops_replaced_percent       | Percentage of HDD read operations replace by SSD.               |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>              | <b>Description</b>   |
|--------------------------------|----------------------------------|--|
| read_ops_blks_total            | read_ops_blocks_total            | Number of non-Flash Pool metafile blocks read operations per second.         |
| read_blks_replaced_percent     | read_blocks_replaced_percent     | Percentage of HDD block reads replaced by SSD.                               |
| write_blks_replaced            | write_blocks_replaced            | Number of HDD write blocks replaced by SSD writes per second.                |
| wc_write_blks_total            | wc_write_blocks_total            | Number of write-cache blocks written per second.                             |
| wc_write_blks_replaced_percent | wc_write_blocks_replaced_percent | Percentage of write-cache block written getting overwritten (to SSD or HDD). |
| est_write_blks_total           | estimated_write_blocks_total     | Estimated number of HDD blocks written per second on a non-Flash Pool.       |
| hdd_write_blks                 | hdd_write_blocks                 | Total HDD blocks written.  |
| write_blks_compensate          | write_blocks_compensate          | Number of write blocks compensation for operation saved calculation.         |
| evict_destage_raw              | evict_destage_raw                | Raw write-cache destaged block count.  |
| write_blks_replaced_percent    | write_blocks_replaced_percent    | Percentage of blocks overwritten to write-cache among all disk writes.       |
| read_rc_nra_hit_type           | read_rc_nra_hit_type             | Block types of non-read-ahead read hits to read cache.                       |
| read_rc_nra_hit_blks_rate      | read_rc_nra_hit_blocks_rate      | Number of blocks of non-read-ahead read hits to read cache.                  |
| read_rc_ra_hit_type            | read_rc_ra_hit_type              | Block types of read-ahead read hits to read cache.                           |
| read_rc_ra_hit_blks_rate       | read_rc_ra_hit_blocks_rate       | Number of blocks of read-ahead read hits to read cache.                      |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>         | <b>Description</b>   |
|--------------------------------|-----------------------------|--|
| read_wc_nra_hit_type           | read_wc_nra_hit_type        | Block types of non-read-ahead read hits to write cache.                    |
| read_wc_nra_hit_blks_rate      | read_wc_nra_hit_blocks_rate | Number of blocks of non-read-ahead read hits to write cache.               |
| read_wc_ra_hit_type            | read_wc_ra_hit_type         | Block types of read-ahead read hits to write cache.                        |
| read_wc_ra_hit_blks_rate       | read_wc_ra_hit_blocks_rate  | Number of blocks of read-ahead read hits to write cache.                   |
| read_miss_type                 | read_miss_type              | Block types of read misses.  |
| read_miss_blks_rate            | read_miss_blocks_rate       | Number of block misses in SSD cache.                                       |
| evict_cool                     | evict_cool                  | Number of temperature decrements done.                                     |
| evict_cool_temp                | evict_cool_temperature      | Number of temperature decrements done per temperature.                     |
| evict_destage                  | evict_destage               | Number of temperature decrements result in a block destage.                |
| evict_destage_rate             | evict_destage_rate          | Number of block destage per second.  |
| evict_remove                   | evict_remove                | Number of temperature decrements that result in a block free.              |
| evict_remove_rate              | evict_remove_rate           | Number of block free per second.   |
| evict_skip_map                 | evict_skip_map              | Number of eviction blocks skipped for map conflict.                        |
| evict_skip_hit                 | evict_skip_hit              | Number of eviction blocks skipped due to pending hit update.               |
| evict_skip_bad                 | evict_skip_bad              | Number of eviction blocks skipped due to corruption in the cachemap entry. |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>             | <b>Description</b>   |
|--------------------------------|---------------------------------|--|
| evict_reasons                  | evict_reasons                   | Reasons for eviction running or not.   |
| hhu_chunk_alloc                | hhu_chunk_allocations           | Number of allocations of buffers containing HyA hit update records.              |
| hhu_chunk_applied              | hhu_chunk_applied               | Number of buffers containing hit update records applied to aggregates.           |
| hhu_chunk_empty                | hhu_chunk_empty                 | Number of buffers containing hit update records freed after all records removed. |
| hhu_chunk_enospace             | hhu_chunk_enospace              | Number of buffers containing hit update records after ssd is full.               |
| hhu_chunk_purged               | hhu_chunk_purged                | Number of buffers containing hit update records purged without applying.         |
| hhu_vbn_added                  | hhu_vbn_added                   | Count of hit update records added.   |
| hhu_vbn_added_lo_ret           | hhu_vbn_added_lo_retention      | Count of hit update records added for low cache retention priority volumes.      |
| hhu_vbn_added_hi_ret           | hhu_vbn_added_hi_retention      | Count of hit update records added for high cache retention priority volumes.     |
| hhu_range_clear                | hhu_range_clear                 | Count of hit update records removed due to eviction.                             |
| hhu_nomem                      | hhu_no_memory                   | Count of hit update records removed due to no memory.                            |
| hhu_susp_mem_alloc             | hhu_suspended_memory_allocation | Count of hit update suspended due to memory allocation.                          |
| hhu_susp_load_active           | hhu_suspended_load_active       | Count of hit update suspended due to load active.                                |
| hhu_susp_load_tmap             | hhu_suspended_load_tmap         | Count of hit update suspended due to Type and Temperature Map (TTMap) load.      |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>                     | <b>Description</b>   |
|--------------------------------|---|--|
| hhu_susp_cow_tmap              | hhu_suspended_cow_tmap                  | Count of hit update suspended due to tmap cow.                               |
| hhu_susp_dirty_tmap            | hhu_suspended_dirty_tmap                | Count of hit update suspended due to Type and Temperature Map (TTMap) dirty. |
| hhu_susp_yield                 | hhu_suspended_yield                     | Count of hit update suspend yield.   |
| hhu_susp_in_cp                 | hhu_suspended_in_cp                     | Count of hit update suspended in CP.   |
| hhu_mod_skipped                | hhu_mod_skipped                         | Count of hit update skipped due to modified buffer.                          |
| hhu_chunk_alloc_pool           | hhu_chunk_allocation_pool               | Allocated memory chunks for tracking the read hits from free pool.           |
| hhu_chunk_free_pool            | hhu_chunk_free_pool                     | Freed memory chunks for tracking the read hits to free pool.                 |
| hhu_msgpool_grown              | hhu_msgpool_grown                       | Number of times hit update apply msgpool grew.                               |
| hhu_msgpool_shrunk             | hhu_msgpool_shrunk                      | Number of times hit update apply msgpool shrunk.                             |
| read_cache_ins_type            | read_cache_inserts_type                 | Block types of cache inserts.  |
| read_cache_ins_blk_total       | read_cache_insert_block_total           | Cache insert block count.  |
| read_cache_ins_rate            | read_cache_insert_rate                  | Cache insert rate blocks/sec.  |
| cache_rej_type                 | cache_reject_type                       | Block types of rejected cache inserts.                                       |
| cache_reject_nospace           | cache_reject_no_space                   | Cache insertion rejects because there is no space.                           |
| cache_rej_reason_dist          | cache_reject_reason_distribution        | Flash Pool cache insert reject reason distribution.                          |
| cache_rej_reason_public_dist   | cache_reject_reason_public_distribution | Flash Pool cache insert reject reason public distribution.                   |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>         | <b>Description</b>  |
|--------------------------------|-----------------------------|---|
| random_cache_dirty             | random_non_mbuf_cache_dirty | Number of non-mbuf random cache dirties.  |
| random_mbuf_cache_dirty        | random_mbuf_cache_dirty     | Number of mbuf random cache dirties.  |
| blkr_rejected_cached_on_ssd    | blkr_rejected_cached_on_ssd | Total HDD blocks that were not relocated because they were SSD cached.  |
| blkr_rejected_ssd_rgs          | blkr_rejected_ssd_rgs       | Total number of times SSD rgs were skipped during blkr or CSC.  |
| ssd_cache_freed                | ssd_cache_freed             | Number of cached SSD blocks freed.  |
| tmap_refs_saturated            | tmap_references_saturated   | Number of references of cached block that do not result in the increase of reference count of Type and Temperature Map (TTMap) due to limited bits.                         |
| dropped_hit_updates            | dropped_hit_updates         | Number of references of cached blocks that happen before the hit update scanner reaches it from last time the reference happens. This results in a dropped reference count. |
| wc_write_blks                  | wc_write_blocks             | Block types of write-cached block.  |
| direct_write_blks              | direct_write_blocks         | Block types of writes of directly allocated SSD blocks.   |
| direct_read_blks               | direct_read_blocks          | Block types of reads of directly allocated SSD blocks.  |
| direct_read_ra_blks            | direct_read_ra_blocks       | Block types of read-aheads of directly allocated SSD blocks.  |
| wc_l0_flag                     | wc_l0_flag                  | Write-caching L0s flagged for WS.   |
| wc_seq                         | wc_seq                      | Write-caching L0s deemed sequential.  |



| <b>Classic Numeric Counter</b> | <b>REST Counter</b>            | <b>Description</b>   |
|--------------------------------|--------------------------------|--|
| wc_wafe_sees_seq               | wc_wafe_deflagged              | Write-caching WAFE deflagged.  |
| wc_blks_ow_to_ssd              | wc_blocks_overwrite_to_ssd     | Write-caching overwrite to SSD.  |
| wc_blks_ow_to_hdd              | wc_blocks_overwrite_to_hdd     | Write-caching overwrite blocks to HDD.   |
| blkr_ssd_read_io               | blkr_ssd_read_io               | Number of block relocation (blkr) SSD read I/Os.                                     |
| blkr_ssd_read_blks             | blkr_ssd_read_blocks           | Number of block relocation (blkr) SSD read blocks.                                   |
| cachemap_partition_blks        | cachemap_partition_blocks      | Number of HDD blocks per cachemap partition.   |
| cachemap_partition_count       | cachemap_partition_count       | Number of cachemap partitions.   |
| hitupdate_complete_scans       | hitupdate_complete_scans       | Number of hitupdate complete scans.  |
| hitupdate_tmap_bufs_dirtied    | hitupdate_tmap_buffers_dirtied | Number of Type and Temperature Map (TTMap) buffers dirtied by the hitupdate scanner. |
| hitupdate_steps                | hitupdate_steps                | Number of hitupdate scan steps.  |
| hitupdate_starts               | hitupdate_starts               | Number of hitupdate scan message starts.   |
| hitupdate_suspends             | hitupdate_suspends             | Number of times hitupdate scan msg suspended voluntarily.                            |
| hitupdate_active_bits_seen     | hitupdate_active_bits_seen     | Number of active flags the hitupdate scanner processed.                              |
| hitupdate_ra_blk_attempts      | hitupdate_ra_block_attempts    | Number of blocks the hitupdate scanner attempted to read ahead.                      |
| hitupdate_blks_accessed        | hitupdate_blocks_accessed      | Number of blocks the hitupdate scanner accessed.                                     |

# Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter                    | Description   |
|---------------------------------|---|
| hya_read_hit_latency_histogram  | The labels have changed from "0 - <1ms" to "0_to_<1ms".   |
| hya_read_miss_latency_histogram | The labels have changed from "0 - <1ms" to "0_to_<1ms".   |
| hya_write_ssd_latency_histogram | The labels have changed from "0 - <1ms" to "0_to_<1ms".   |
| hya_write_hdd_latency_histogram | The labels have changed from "0 - <1ms" to "0_to_<1ms".   |
| read_rc_nra_hit_type            | For each of the new labels in this array, the values have been changed to be lower case and any other non alpha-numeric characters have been replaced by an underscore. |
| read_rc_ra_hit_type             | For each of the new labels in this array, the values have been changed to be lower case and any other non alpha-numeric characters have been replaced by an underscore. |
| read_wc_nra_hit_type            | For each of the new labels in this array, the values have been changed to be lower case and any other non alpha-numeric characters have been replaced by an underscore. |
| read_wc_ra_hit_type             | For each of the new labels in this array, the values have been changed to be lower case and any other non alpha-numeric characters have been replaced by an underscore. |
| read_miss_type                  | For each of the new labels in this array, the values have been changed to be lower case and any other non alpha-numeric characters have been replaced by an underscore. |

| <b>REST Counter</b>                     | <b>Description</b>  |
|---|---|
| evict_cool_temperature                  | The labels have changed from "2→1" to "1_to_2".   |
| evict_reasons                           | The labels have changed from "maint mode" to "maintenance_mode".  |
| read_cache_inserts_type                 | For each of the new labels in this array, the values have been changed to be lower case and any other non alpha-numeric characters have been replaced by an underscore. |
| cache_reject_type                       | For each of the new labels in this array, the values have been changed to be lower case and any other non alpha-numeric characters have been replaced by an underscore. |
| cache_reject_reason_distribution        | The labels have changed from "buf_needs_iron" to "buffer_needs_iron".   |
| cache_reject_reason_public_distribution | The labels have changed from "buf_needs_iron" to "buffer_needs_iron".   |
| wc_write_blocks                         | For each of the new labels in this array, the values have been changed to be lower case and any other non alpha-numeric characters have been replaced by an underscore. |

## Table Aliases

This section describes aliases for aggregated tables.

# wafh\_hya\_sizer

This table is for exporting Automated Working-set Analyzer (AWA) statistics. AWA is an ONTAP capability that estimates the Flash Pool cache size and estimates the cache hit rates across reads and writes. Detailed statistic values of an AWA instance enabled on a specific aggregate are available here.

**Classic Object:** wafh\_hya\_sizer

## Table Row IDs

| ID Format       | Aggregation Type | Comment  |
|-----------------|------------------|--|
| {instance_name} | (not applicable) | This represents the construction of the row ID field, which is a single unique string that identifies a row. |

## Properties

This section describes the mapping between classic (ONTAPI) string counter names and REST property names.

| Classic String Counter | REST Property  | Description            |
|------------------------|----------------|------------------------|
| node_name              | node.name      | System node name       |
| aggr_name              | aggregate.name | Name of the aggregate. |

## Counters

This section describes the mapping between classic (ONTAPI) numeric counter names and REST counter names.

| Classic Numeric Counter | REST Counter   | Description  |
|-------------------------|----------------|--|
| version                 | version        | Automated Working-set Analyzer (AWA) version.                          |
| start_time              | start_time     | Start time in milliseconds.  |
| current_time            | current_time   | Current time based on ONTAP internal clock in milliseconds (sk_msecs). |
| start_cp_count          | start_cp_count | CP (Consistency Point) count when the sizer is enabled.                |

| <b>Classic Numeric Counter</b> | <b>REST Counter</b>      | <b>Description</b>  |
|--------------------------------|--------------------------|---|
| rstart_time                    | relative_start_time      | Relative start time of the scoped interval in milliseconds.   |
| rstart_cp_count                | relative_start_cp_count  | CP (Consistency Point) count when the sizer is started relative to the beginning of scoped intervals. |
| total_intervals                | total_intervals          | Total number of intervals since the sizer was enabled including the current one.                      |
| overwrite_cp_delta             | overwrite_cp_delta       | The histogram of CP (Consistency Point) count delta between overwrites.                               |
| hash_max_unique                | hash_max_unique          | Aggregate space max_unique for hashing.   |
| hash_sampling_factor           | hash_sampling_factor     | Aggregate space hash sampling factor.   |
| target_pcache_mb               | target_pcache_mb         | Configured target pseudo cache memory in megabytes.   |
| pcache_mem                     | pcache_memory            | Derived memory size during configuration.   |
| target_sampling_factor         | target_sampling_factor   | Configured target sampling factor.  |
| derived_sampling_factor        | derived_sampling_factor  | Derived sampling factor.  |
| pcache_no_of_entries           | pcache_number_of_entries | Derived number of pseudo cache entries which is the same as the number of cached blocks.              |
| incore_intervals               | incore_intervals         | Total incore intervals.   |
| interval_msec                  | interval_msec            | Interval length in milliseconds.  |
| num_disks                      | number_disks             | Number of disks.  |

| Classic Numeric Counter | REST Counter          | Description  |
|-------------------------|-----------------------|--|
| sync_with_stats_zero    | sync_with_stats_zero  | If the value is one, the Automated Working-set Analyzer (AWA) interval resets when waf_l_stats_zero command).  |
| enable_rc_after_write   | enable_rc_after_write | If the value is one, the option of Read-cache After Write is enabled.  |
| vvol_stats              | vvol_stats            | Collect per-volume statistics.   |
| debug_level             | debug_level           | Debug Level.   |
| scope_interval          | scope_interval        | Number of intervals of the sliding window scope.   |
| scope_phaseout          | scope_phase_out       | If the value is 1, AWA phases out old accumulations from the histograms.   |
| cache_mode              | cache_mode            | Automated Working-set Analyzer (AWA) cache mode: normal, read-cache only, write-cache only, always cache.  |
| interval_info           | interval_info         | Interval Information.  |
| lookup_type_mismatch    | lookup_type_mismatch  | Number of type mismatches when lookup finds an entry.  |
| entry_matched           | entry_matched         | Number of pseudo cache entry lookup attempts. For random-hash-sampled cases, this is the number of lookup/insert attempts. For statically sampled case, this is the number of lookups that falls into sampled disks. |
| entry_sampled           | entry_sampled         | Number of pseudo cache hashed Volume Block Number (VBN) sampled based on sampling factor.  |

| Classic Numeric Counter  | REST Counter                | Description  |
|--------------------------|-----------------------------|--|
| wc_flagged               | wc_flagged                  | Number of buffers flagged for write-caching after check for random write at buffer-dirty time.   |
| can_wc_dirty             | can_wc_dirty                | Number of buffers that can be write-cached at write-allocation time.   |
| cant_wc_dirty            | cant_wc_dirty               | Number of buffers that cannot be write-cached at write-allocation time.  |
| wc_overwrite             | wc_overwrite                | Number of write-cached blocks being overwritten.   |
| long_chain               | long_chain                  | Number of reads with buffer count long enough to go to hard disk drives (HDD) directly.  |
| wc_ins_for_destage       | wc_ins_for_destage          | Number of Automated Working-set Analyzer (AWA) write-cached block inserts for actual destages on a Flash Pool aggregate.   |
| rc_after_write           | rc_after_write              | Number of buffers flagged for read cache after write.  |
| cache_rej_reason_dist_FP | cache_reject_reason_dist_fp | Automated Working-set Analyzer (AWA) pseudo cache insert reject reason distribution for Floating Point (FP) cache inserts. This array is a table with fields corresponding to the reject reasons defined by enum type <code>hya_cache_rej_reasons_t</code> . |

| Classic Numeric Counter           | REST Counter                            | Description  |
|-----------------------------------|---|--|
| cache_rej_reason_dist             | cache_reject_reason_dist                | Automated Working-set Analyzer (AWA) pseudo cache insert reject reason distribution not for FP aggregate cache inserts. This array is a table with fields corresponding to the reject reasons defined by enum type <code>hya_cache_rej_reasons_t</code> .  |
| aggrwide_regular_stats            | aggregate_wide_regular_stats            | Automated Working-set Analyzer (AWA) aggregate-wide regular statistics for the most recent number of intervals. The number of intervals that are considered recent is defined by <code>CM_WAFL_HYAS_INT_DIS_CNT</code> . This array is a table with fields corresponding to the enum type <code>hya_regular_stat_type_t</code> . |
| aggrwide_cache_stats              | aggregate_wide_cache_stats              | Automated Working-set Analyzer (AWA) aggregate-wide cache statistics for the most recent number of intervals. The number of intervals that are considered as recent is defined as <code>CM_WAFL_HYAS_INT_DIS_CNT</code> . This array is a table with fields corresponding to enum type <code>hyas_cache_stat_type_t</code> .     |
| aggrwide_cache_interarrival_stats | aggregate_wide_cache_interarrival_stats | Automated Working-set Analyzer (AWA) aggregate-wide interarrival distribution for intervals up to <code>CM_WAFL_HYAS_INT_DIS_CNT</code> . This array is a table with fields corresponding to the enum type of <code>hya_cache_ia_stat_type_t</code> .  |



| Classic Numeric Counter  | REST Counter             | Description  |
|--------------------------|--------------------------|--|
| regular_stats            | regular_stats            | Automated Working-set Analyzer (AWA) per-interval regular statistics for the most recent number of intervals. The number of intervals defined as recent is CM_WAFL_HYAS_INT_DIS_CNT. This array is a table with fields corresponding to the enum type of hyas_regular_stat_type_t. |
| cache_stats              | cache_stats              | Automated Working-set Analyzer (AWA) per-interval pseudo cache statistics for the most recent intervals. The number of intervals defined as recent is CM_WAFL_HYAS_INT_DIS_CNT. This array is a table with fields corresponding to the enum type of hyas_cache_stat_type_t.        |
| cache_interarrival_stats | cache_interarrival_stats | Automated Working-set Analyzer (AWA) pseudo cache interarrival distribution for intervals up to CM_WAFL_HYAS_INT_DIS_CNT. This array is a table with fields corresponding to the enum type of hya_cache_ia_stat_type_t.  |

## Property/Counter Content Changes

This section describes any output value differences between the classic (ONTAPI) string counter and the respective REST property. It also describes array label name changes between classic array counters and respective REST array counters.

| REST Counter                 | Description  |
|------------------------------|--|
| aggregate_wide_regular_stats | The labels have changed from "Free overwrite" to "free_overwrite".       |
| aggregate_wide_cache_stats   | The labels have changed from "rd_rc_L0_interval" to "rd_rc_I0_interval". |

| <b>REST Counter</b> | <b>Description</b>   |
|---------------------|--|
| regular_stats       | The labels have changed from "Free overwrite" to "free_overwrite".       |
| cache_stats         | The labels have changed from "rd_rc_L0_interval" to "rd_rc_I0_interval". |

## Table Aliases

This section describes aliases for aggregated tables.

# Legal notices

Legal notices provide access to copyright statements, trademarks, patents, and more.

## Copyright

<https://www.netapp.com/company/legal/copyright/>

## Trademarks

NETAPP, the NETAPP logo, and the marks listed on the NetApp Trademarks page are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.

<https://www.netapp.com/company/legal/trademarks/>

## Patents

A current list of NetApp owned patents can be found at:

<https://www.netapp.com/pdf.html?item=/media/11887-patentspage.pdf>

## Privacy policy

<https://www.netapp.com/company/legal/privacy-policy/>

## Open source

Notice files provide information about third-party copyright and licenses used in NetApp software.

## Copyright information

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

## Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.