NetApp® Manageability SDK 9.8

Reference Manual for Active IQ Unified Manager 9.12 APIs

NetApp, Inc.
1395 Crossman Ave.
Sunnyvale, CA 94089
U.S.

Telephone: +1(408) 822-6000
Fax: +1(408) 822-4501
Support telephone: +1 (888) 463-8277
Web: www.netapp.com
Feedback: doccomments@netapp.com

Part number: 215-15305_A0
November 2022

Updated for 9.8P5

Introduction to Active IQ Unified Manager APIs

You can use the APIs of Active IQ Unified Manager to develop third-party applications to monitor and manage NetApp storage systems. You can manage and monitor storage systems by using the advanced functionality provided by the APIs of the applications included in the Active IQ Unified Manager server.

The following table lists common management tasks and the corresponding API set that you can use:

Tasks to be performed	What to use
Storage system health monitoring	Unified Manager APIs
Data protection	Protection capability APIs
Command-line tasks not covered by any other API	`ssh`, `rsh`, or telnet

API classification

This section provides guidelines on how to use the APIs for various operations related to the Unified Manager server and its components.

The Unified Manager APIs are broadly classified into the following categories:

- Unified Manager APIs
- Protection capability APIs

Note: Some APIs are part of more than one of the above categories.

The following table lists the Unified Manager APIs and the corresponding API families:

Unified Manager product APIs	API family
Unified Manager APIs	• aggregate • cluster • disk • event • igroup • LUN • net • portset • resource • SSL • system • volume • Vserver
Protection capability APIs	• dataprotection • job • resource-pool • snapshot • storage-service

ONTAPI Categories

API version 9.12

The links below will take you to the documentation for individual API's in each of the categories.

aggregate	Enables the user to access collected information about aggregates.
cluster	Enables the user to access collected information about clusters.
dataprotection	Enables the user to manage the protection of storage system data.
disk	Enables the user to access collected information about storage system disks.
event	Enables the user to manage existing events.
igroup	Enables the user to access collected information about igroups.
job	Enables the user to submit, cancel, delete, and list management jobs.
lun	Enables the user to access collected information about LUNs.
net	Enables the user to access collected information about LIFs, ports, and VLANs.
portset	Enables the user to access collected information about portsets.
resource	Enables the user to lookup key values for managed resources.
resource-pool	Enables the user to manage resource pools.
snapshot	Enables the user to manage Snapshot copies.
ssl	Enables the user to manage TLS and SSL communication.
storage-service	Enables the user to enforce storage service policies for data protection.
system	Enables the user to retrieve system information for the application instance.
volume	Enables the user to access collected information about volumes.
vserver	Enables the user to access collected information about Vservers.

aggregate-iter

[top]

Iterate over existing aggregates.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • cluster • cluster node • aggregate If resource-filter identifies a aggregate, that single aggregate will be returned. If resource-filter resolves to more than one aggregate, all of them will be returned. If no resource-filter is provided, all aggregates will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		aggregate-info[] optional	The list of records.

Element definition: **aggregate-info**

[top]

Information about a aggregate.

Name	Range	Type	Description
aggr-volume-		aggr-volume-	Provides information about volumes moving into and out of

move-info		move-info	aggregates.
aggregate-name		string	Full hierarchical name of the aggregate present in the output. The name is of the form, cluster-node:myaggr.
aggregate-resource-key		resource-key	The resource key for this aggregate.
aggregate-size		aggregate-size	Sizes of various parameters of the aggregate.
aggregate-snaplock-type		string optional	Snaplock-type of the aggregate. Possible values - "compliance", "enterprise" or "non-snaplock".
aggregate-space-status		object-space-status	Space status of the aggregate. This indicates the fullness of the aggregate in terms of whether the percentage of used space with respect to total size of the aggregate has reached or crossed the fullness thresholds given in aggregate-nearly-full-threshold and aggregate-full-threshold.
aggregate-state		string	State of aggregate. Following are the possible values: • offline • online • restricted • iron_restricted • creating • failed • partial • destroying • frozen • inconsistent • mounting • quiesced • quiescing • reverted • unmounted • unmounting • unknown
aggregate-status		obj-status	Current status of the aggregate based on all events
block-type		file-system-block-type	File system block type of the aggregate. The volumes on both the source and destination sides of a SnapMirror relationship must be of the same block type.
cluster-name		string	Name of the cluster present in the output. The name is any simple name such as myhost.
cluster-node-name		string	Name of controller. Always present in the output. The name is any simple name such as mynode.
cluster-node-resource-key		resource-key	The resource key for the cluster-node. Always present in the output.
cluster-resource-key		resource-key	The resource key for the cluster. Always present in the output.
compression-space-savings	**[0..2^63-1]**	integer	Space savings as a result of compression on the volume in bytes.
dedupe-space-savings	**[0..2^63-1]**	integer	Space savings as a result of deduplication on the aggregate's volumes in bytes.
has-local-		boolean	Specifies whether the aggregate contains the local root

root			volume.
has-partner-root		boolean	Specifies whether the aggregate contains the partner's root volume.
hybrid-cache-size-total	**[0..2^63-1]**	integer	Total cache size (in bytes) in a hybrid aggregate. If the referenced aggregate is restricted or offline, or if it is not a hybrid aggregate, a value of 0 is returned.
is-cft-precommit		boolean	Specifies whether a given aggregate is in pre-commit state or not.
is-hybrid		boolean	Specifies whether a given aggregate contains a mix of SSD and HHD RAID groups.
is-hybrid-enabled		boolean	Specifies whether a given aggregate is eligible to contain both SSD and HDD RAID groups.
raid-type		string optional	Name of an allowed RAID type. Possible values are "raid0", "raid4", "raid_dp" and "mixed_raid_type".
time-to-full	**[0..31536000]**	integer optional	Estimated amount of time left in seconds for the aggregate to become full. This is returned as empty when the estimated amount of time is more than a year. This can happen due to a very low or negative rate of consumption of space in the aggregate. Also, this field will not be returned if sufficient history data about a given aggregate is unavailable.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **aggr-volume-move-info**

[top]

Provides information about volumes moving into or out of aggregates.

Name	Range	Type	Description
volume-in-count	**[0-2^31-1]**	integer	Total number of volumes currently being moved into this aggregate whose volume move state is in-progress or pause.
volume-in-total-size	**[0..2^63-1]**	integer	Sum of the sizes, in bytes, of all the volumes currently being moved into this aggregate whose volume move state is in-progress or pause.
volume-out-count	**[0-2^31-1]**	integer	Total number of volumes currently being moved out of this aggregate whose volume move state is in-progress or pause.
volume-out-total-size	**[0-2^63-1]**	integer	Sum of the sizes, in bytes, of all the volumes currently being moved out of this aggregate whose volume move state is in-progress or pause.

Element definition: **aggregate-size**

[top]

Sizes of various parameters of an aggregate.

Name	Range	Type	Description
daily-growth-rate	**[0..100]**	integer optional	Percentage of change that occurs every 24 hours in the disk space used by the aggregate.
size-available	**[0..2^63-1]**	integer	Available bytes in the aggregate.
size-total	**[0..2^63-1]**	integer	Aggregate total size in bytes.
size-used	**[0..2^63-1]**	integer	Aggregate bytes used.
size-used-per-day	**[-2^44-1..2^44-1]**	integer optional	The capacity (in bytes) used per day. This can be either positive or negative depending on the growth of used space in the aggregate. The information is based on the regression slope of the aggregate usage history data.
snapshot-reserve-avail	**[0..2^63-1]**	integer optional	The available capacity (in bytes) in the Snapshot reserve for the aggregate. If snapshot-reserve-used is greater than snapshot-reserve-total, this value will be zero.
snapshot-reserve-total	**[0..2^63-1]**	integer optional	Total capacity (in bytes) of Snapshot copy reserve.
snapshot-reserve-used	**[0..2^63-1]**	integer optional	Total capacity (in bytes) for Snapshot data. The number can be greater than the Snapshot reserve size, but does not include any space used out of overwrite reserve.
space-total-committed	**[0..2^63-1]**	integer optional	Total space committed in bytes.

Element definition: **file-system-block-type**

[top]

Block Type of the file system. The volumes on both the source and destination sides of a SnapMirror relationship must be of the same block type. Volumes contained in a larger parent agregate may have a block-type of 64_bit. For upgraded systems it is possible that this value may be unknown until the system can determine the block-type. Possible values are:

- 32_bit
- 64_bit
- unknown

[none]

Element definition: **obj-status**

[top]

A status value which can be associated with an object. This typedef is an alias for the builtin ZAPI type **string**. The severity associated with an event has this type.

Possible values are: 'normal', 'warning', 'error', 'critical'.

- normal: An object has normal status when it is working within the thresholds specified.
- warning: An object has the warning status when an event related to the object occurred that an

administrator should know about. The event will not cause service disruption.
- error: An object has error status when it does not cause any service disruption, but it may affect performance.
- critical: An object has critical status when it is still performing, but service disruption may occur if corrective action is not taken immediately.

In some contexts, it is important that severities are ordered (as above). For example, an alert might be triggered if an event with a given severity "or worse" occurs. In this example, worse means "after" in the list above.

[none]

Element definition: **object-space-status**

[top]

Space status of the object. This indicates the fullness of the object in terms of whether the percentage of used space with respect to total size of the object has reached the fullness thresholds. Possible values:

- ok - when the percentage of used space of the object is within the nearly full and full threshold of the object.
- nearly_full - when the percentage of used space of the object is within the full threshold of the object but has reached or crossed the nearly full threshold.
- full - when the percentage of used space of the object has reached or crossed the full threshold of the object.

[none]

APIs in Category: **cluster**

API version 9.12

cluster-iter

[top]

Iterate over clusters.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • cluster If resource-filter identifies a cluster, that single cluster will be returned. If resource-filter resolves to more than one cluster, all of them will be returned. If no resource-filter is provided, all clusters will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		cluster-info[] optional	The list of records.

Errno	Description
EINVALIDINPUT	
EOBJECTAMBIGUOUS	
EOBJECTNOTFOUND	

cluster-node-iter

[top]

Iterate over cluster nodes.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • cluster • cluster node If resource-filter identifies a cluster node, that single cluster node will be returned. If resource-filter resolves to more than one cluster node, all of them will be returned. If no resource-filter is provided, all cluster nodes will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		cluster-node-info[] optional	The list of records.

Errno	Description
EINVALIDINPUT	
EOBJECTNOTFOUND	

Element definition: **cluster-info**

[top]

Cluster's information.

Name	Range	Type	Description
cluster-address		string	The cluster's primary IP address.
cluster-diagnosis-status		string	The diagnosis status of the cluster. Possible values: "ok", "ok-with-suppressed", "degraded" or "unreachable".
cluster-name		string	This is the name of the cluster. Length: [1..255]
cluster-status		obj-status	Current status of the host based on all events
cluster-version		string	This is the software release for a requested cluster. Length: [1..255]
datasource-id		obj-id	The id of the datasource used to collect information about this cluster.
last-update-time		timestamp	Start time of the last data collection.

licenses		license[]	List of licenses installed on the cluster.
resource-key		resource-key	The resource key for this cluster.
serial-number		string	The serial number of the cluster. Length: [1..64]

Element definition: **cluster-node-info**

[top]

Information for a given node.

Name	Range	Type	Description
bytes-spare	**[0..2^63-1]**	integer	Spare raw capacity of the node in bytes.
bytes-total	**[0..2^63-1]**	integer	Total raw capacity of the node in bytes.
bytes-used	**[0..2^63-1]**	integer	Used raw capacity of the node in bytes.
cluster-name		obj-name	Name of the cluster where the node is present.
cluster-node-aggr-size-info		cluster-node-aggr-size	Information about aggregates on a node.
cluster-node-disk-info		cluster-node-disk-info	Information about disks on a node.
cluster-node-down-timestamp		timestamp	The downtime of the node. This is the number of seconds elapsed since midnight on January 1, 1970.(UTC)
cluster-node-interconnect-info		cluster-node-interconnect-info	Information about the failover interconnect on a node.
cluster-node-name		obj-name	This is the name of the node. Length: [1..255]
cluster-node-port-count	**[1..2^31-1]**	integer	Number of physical ports (both ethernet and fibrechannel) in the node.
cluster-node-resource-key		resource-key	The resource key of the node.
cluster-node-state		string	State of the node. Possible values are: "online", "offline"
cluster-node-status		string	Current status of the node.
cluster-node-uptime	**[0..2^32-1]**	integer	The total time, in seconds, that the node has been up.
cluster-node-uuid		string	The univerally unique identifier for the node. It is a 36-character string composed of 32 hexadecimal characters. For example, '542366ea-a024-11dd-9caa-7302e474c5ae'.
cluster-resource-key		resource-key	Resource-key of the cluster where the node is present.
contact		string optional	The owner of the node.
cpu-percent-busy	**[0..2^31-1]**	integer optional	Percentage of the time that the node's CPU is busy.

cpu-percent-busy-duration		integer optional	Time in seconds between measurements of CPU usage.
env-failed-fan-status		string optional	Indication of the number of chassis fans which are not operating within the recommended RPM range. Possible values are: "normal", "one_failed", "many_failed"
env-failed-power-supply-status		string optional	Indication of the number of failed power supply units. Possible values are: "normal", "one_failed", "many_failed"
env-over-temperature-status		string optional	An indication of whether the hardware is currently operating outside of its recommended temperature range. The hardware will shutdown if the temperature exceeds critical thresholds. Possible values are: "normal", "hot"
firmware-version		string optional	Firmware version of the controller.
flash-cards		flash-card-info[] optional	Information about flash cards on a node.
location		string optional	The physical location of the node as reported by Data ONTAP.
model		string optional	The model of the node.
nvram-battery-status		string optional	Status of the NVRAM battery. Possible values include: • battery_ok • battery_partially_discharged • battery_fully_discharged • battery_not_present • battery_near_end_of_life • battery_at_end_of_life • battery_unknown • battery_over_charged • battery_fully_charged
nvram-id		integer optional	Vendor specific NVRAM identifier of the node.
os-version		string	Data ONTAP version running on the node.
serial-number		string	Serial number of the node.
sfo-info		sfo-info	Information about the storage failover configuration of this node.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **cluster-node-aggr-size**

[top]

Information about the size of the aggregates in a node.

Name	Range	Type	Description
aggr-bytes-total	**[0..2^63-1]**	integer	Total capacity of the aggregates (in bytes) in a node.
aggr-bytes-used	**[0..2^63-1]**	integer	Used capacity of the aggregates (in bytes) in a node.

Element definition: **cluster-node-disk-info**

[top]

Collected information about disks on a node. Optional items will not be returned if the value is not known.

Name	Range	Type	Description
disk-failed-message		string optional	Provides information about a disk failure.
failed-disk-count		integer optional	Number of failed disks on the node.
reconstructing-disk-count		integer optional	Number of disks that are being reconstructing on the node.
reconstructing-parity-disk-count		integer optional	Number of parity disks that are being reconstructed on the node.
scrubbing-disk-count		integer optional	Number of disks that are being scrubbed on the node.
verifying-parity-disk-count		integer optional	Number of parity disks that are being verified on the node.

Element definition: **cluster-node-interconnect-info**

[top]

Collected information about a failover interconnect between two nodes.

Name	Range	Type	Description
interconnect-links		string optional	States of the individual interconnect links e.g. VIA Interconnect is down (link 0 down, link 1 down) VIA Interconnect is up (link 0 down, link 1 up) VIA Interconnect is up (link 0 up, link 1 down)
interconnect-type		string optional	Type and vendor of the interconnect e.g. Infiniband (Mellanox Arbel) Infiniband (Mellanox Tavor) FCVI (Qlogic 2462)
is-interconnect-up		boolean	True, if storage clustering interconnect is up

Element definition: **flash-card-info**

[top]

Collected information about a flash card on a node. Optional items will not be returned if the value is not known.

Name	Range	Type	Description
capacity	**[0..2^31-1]**	integer	Advertised capacity of the device, in gigabytes.
firmware-revision		string	Firmware revision of the FPGA on the flash device.
model-name		string	model name of the flash device in XXXXXX-XX format.
percent-online	**[0..100]**	integer	Percentage of device capacity that is currently online.
serial-number		string	unique 10 digit serial number of the flash device.
slot-number	**[1..255]**	integer	PCI-e slot number of the flash device.
status		string	The current status of the device. Possible values are "online", "ok", "failed", "erasing", "erased", "offline_failed" and "offline_threshold".

Element definition: **license**

[top]

License information of the Data ONTAP service.

Name	Range	Type	Description
description		string	Short description of the license package.
expiration-time		timestamp	Time this license expires. This is the number of seconds elapsed since midnight on January 1, 1970.(UTC)
owner-name		string	Name of the owner (cluster, or cluster-node) Length: [1..255]
owner-resource-key		resource-key	Resource-key of the owner of this license. Can be resource-key of a cluster or cluster-node depending on owner-type.
owner-type		resource-type	Resource Type of cluster or cluster-node owning the license.
package-name		string	Name of the licensed Data ONTAP service. Possible values: • base • nfs • cifs • iscsi • fcp • cdmi • snaprestore • snapmirror • flexclone • snapvault • snaplock • snapmanagersuite • snapprotectapps • v_storageattach • snaplock_enterprise • insight_balance
serial-number		string	The serial number of the owner (cluster, or cluster-node). Length: [1..64]

Element definition: **obj-id**

[top]

Identification number (ID) for an object. This typedef is an alias for the builtin ZAPI type **integer**. Object IDs are unsigned integers in the range [1..2^31 - 1]. In some contexts, an object ID is also allowed to be 0, which is interpreted as a null value, e.g., a reference to no object at all.

The ID for an object is always assigned by the system; the user is never allowed to assign an ID to an object. Therefore, an input element of type **obj-id** is always used to refer to an existing object by its ID. The ZAPI must specify the object's object type (e.g. cluster, volume, aggregate, etc.). Some ZAPIs allow the object to be one of several different types.

If the value of an **obj-id** input element does not match the ID of any existing object of the specified type or types, then typically the ZAPI fails with error code **EOBJECTNOTFOUND**. A ZAPI may deviate from this general rule, for example, it may return a more specific error code. In either case, the ZAPI specification must document its behavior.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **obj-status**

[top]

A status value which can be associated with an object. This typedef is an alias for the builtin ZAPI type **string**. The severity associated with an event has this type.

Possible values are: 'normal', 'warning', 'error', 'critical'.

- normal: An object has normal status when it is working within the thresholds specified.
- warning: An object has the warning status when an event related to the object occurred that an administrator should know about. The event will not cause service disruption.
- error: An object has error status when it does not cause any service disruption, but it may affect performance.
- critical: An object has critical status when it is still performing, but service disruption may occur if corrective action is not taken immediately.

In some contexts, it is important that severities are ordered (as above). For example, an alert might be triggered if an event with a given severity "or worse" occurs. In this example, worse means "after" in the list above.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **sfo-info**

[top]

Information about the storage failover configuration of this node.

Name	Range	Type	Description
current-mode		string optional	Current HA mode
			Giveback state of the node. Possible values are: • nothing_to_gb - No partner aggregates owned by local node. • not_attempted_yet - Local node owns partner's aggregates and a giveback has not been attempted yet. • giveback_in_progress - Giveback is in progress. Refer to current-giveback-module field to get the current module, the giveback process is in. • giveback_failed_autogiveback_disabled- Previous giveback failed and auto giveback is disabled. Refer to current-giveback-module to get the module in which giveback process failed. • giveback_failed_autogiveback_scheduled- Previous giveback failed and an auto giveback is scheduled. Refer to current-giveback-module to get the module in which giveback process failed. Refer time-until-autogiveback field to check time remaining before an auto giveback is initiated.

giveback-state		string optional	• previous_giveback_failed - Previous giveback failed. Refer to current-giveback-module to get the module in which giveback process failed. • giveback_vetoed_no_di - Normal giveback not possible as disk inventory from partner has not yet been received. • giveback_vetoed_missing_disks - Normal giveback not possible as the partner is missing some of its files system disks. • autogiveback_scheduled - An auto giveback is scheduled Refer time-until-autogiveback field to check time remaining before an auto giveback is initiated. • autogiveback_deferred - Auto giveback is deferred because the partner node was not ready to receive aggregates when the auto giveback timer expired. An auto giveback will be initiated as soon as the partner node is up and ready to receive aggregates. • node_upgrade_in_progress - Local node owns partner's aggregates as part of node upgrade process. • sfo_aggr_giveback_failed - Giveback of SFO aggregates failed. Call cf-aggregate-giveback-status API for more information regarding giveback failure. • sfo_aggr_giveback_in_progress - Giveback of SFO aggregates is in progress. Call cf-aggregate-giveback-status API for more information regarding giveback status. • partial_giveback - Local node owns partner's SFO aggregates. • partner_spare_disks_giveback_pending- Local node owns partner's spare disks.
partner-name		obj-name	The fully qualified domain name of the partner controller if this node is configured as part of an HA pair. Length: [1..255]
partner-node-status		string	Status of the partner node. This field is not returned if this node is not part of an HA pair.
partner-resource-key		resource-key	This is the resource key of the partner controller if this node is configured as part of an HA pair.
sfo-state		string	Storage failover configuration state. Possible values: "connected", "takeover_scheduled", "takeover_started", "takeover", "taken over", "takeover_failed", "giving_back", "giveback_partial_waiting", "giveback_partial_connected", "waiting_for_root_aggr", "waiting", "in_maintenance_mode", "pending_shutdown", "error".
sfo-status		string	Status of the SFO of the node. Possible values: "not_configured", "enabled", "disabled".
takeover-by-partner-not-possible-reason		string optional	If takeover by the partner is not possible, list of one or more reasons why.
takeover-failure-reason		string optional	Only returned if sfo-state is equal to takeover_failed. This is the reason for the takeover failure.
takeover-of-partner-not-possible-reason		string optional	If the storage failover facility is disabled, list of one or more reasons why.

Element definition: **timestamp**

[top]

Seconds since 1/1/1970 in UTC.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **resource-type**

[top]

A canonical type name for a managed resource. Includes both Data ONTAP and OnCommand resource types. Valid Data ONTAP resource types:

- aggregate
- objectstore_config
- capacity_tier
- cifs_share
- cifs_share_acl
- cluster
- cluster_node
- disk
- export_policy
- export_rule
- fcp_lif
- fcp_port
- flash_device
- igroup
- iscsi_portal_group
- lun
- lun_mapping
- network_lif
- network_port
- ontap_job_schedule
- plex
- portset
- qtree
- raid_group
- routing_group
- service_processor
- sis_policy
- snap_mirror
- snapshot
- snapshot_policy
- snapshot_policy_schedule
- storage_class
- storage_shelf
- volume
- volume_move
- vserver
- vserver_name_mapping

Valid OnCommand resource types:

- management_station
- resource_pool
- service_workflow
- storage_service
- storage_service_connection
- storage_service_node

[none]

Data protection means backing up data and being able to recover it. Data is protected by making copies of it so that it is available for restoration even if the original data is no longer available. Data protection involves two key aspects: one is the ability to replicate the data, and the other is restoring it to the original location or new location for use as if it was the original. The ability to restore data that has been protected is core functionality of data protection solution.

dp-relationship-iter

[top]

Lists data protection relationships. For example, SnapVault or SnapMirror relationships. List relationships for a single storage service connection or for a particular source or destination name or id.

Input Name	Range	Type	Description
is-managed		boolean optional	If true, only list relationships which are managed by a storage-service. If false, only list relationships which are not in a storage service. If unspecified, list all relationships.
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
relationship-resource-key		resource-key optional	Resource key of relationship. relationship-resource-key cannot be specified with any other inputs.
relationship-states		relationship-state[] optional	If specified, only the relationships in any of these states are returned. If not present, do not filter by state.
relationship-statuses		relationship-status[] optional	If specified, only the relationships in any of these statuses are returned. If not present, do not filter by status.
relationship-type		relationship-type optional	If present, list only relationships of specified type. If not present, do not filter by type.
source-or-destination-resource-key		resource-key optional	Resource key of either a source or a destination object. If empty, all relationships are listed. The source or destination object must be either a Volume or Vserver. If Vserver is specified, relationships for all volumes in that Vserver are listed.
storage-service-connection-resource-key		resource-key optional	Resource key of the storage service connection. If present, only relationships on the specified connection will be listed. storage-service-connection-resource-key cannot be specified with storage-service-resource-key and is-managed inputs.
storage-service-resource-key		resource-key optional	Resource key of storage service. Lists relationships managed by the storage service. storage-service-resource-key cannot be specified with is-managed, and storage-service-connection-resource-key input.
			Specify the tag from the last call. It is not specified for the first call.

tag		string optional	For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		dp-relationship-info[] optional	The list of records.

Errno	Description
EINVALIDINPUT	
EOBJECTNOTFOUND	

dp-restore-start

[top]

Starts a restore/retrieval operation on part of a storage object. This API copies storage objects (files, LUNS, directories, qtrees, and volumes) from a given Snapshot copy to a specified location.

Users can run one of the following APIs to obtain the source snapshot resource key.

1. resource-lookup
2. snapshot-get-location

This API returns the job-id of the job created for the restore operation. The job ID can be used to lookup detailed information on the progress of the restore operation.

Input Name	Range	Type	Description
restore-requests		dp-restore-request-info[]	List of the objects to be restored.

Output Name	Range	Type	Description
job-id		job-id	The identification of the job created by this API request. The job is a manifest that expresses the set of different tasks and the sequence in which they are executed in order to accomplish a high level work-flow.

Errno	Description
EOBJECTNOTFOUND	
EINVALIDINPUT	
E_INPUT_NOT_IMPLEMENTED	

Element definition: **dp-relationship-info**

[top]

Information about relationship.

Name	Range	Type	Description
current-transfer-error		string optional	A message describing the last retryable error encountered by the current transfer. Applicable to Data ONTAP 8.2 and higher.
destination-name		obj-full-name	Full name of storage object where this relationship ends.
destination-resource-key		resource-key	Storage resource key of storage object where this relationship ends.
is-healthy		boolean	True if the most recent update succeeded and if it was a scheduled update occurred as scheduled. Otherwise false.
lag-duration		integer optional	The amount of time in seconds by which the data on the mirror lags behind the source. Applicable to Data ONTAP 8.2 and higher.
last-transfer-duration		integer optional	The amount of time in seconds it took for the last transfer to complete. Applicable to Data ONTAP 8.2 and higher.
last-transfer-error		string optional	A message describing the cause of the last transfer failure. Present only if the last transfer was unsuccessful. Applicable to Data ONTAP 8.2 and higher.
max-transfer-rate	**[0..2^31 - 1].**	integer	Specifies the upper bound at which data is transferred between clusters, in kilobytes per second. 0 is unlimited which permits the relationship to fully utilize the available network bandwidth. The max-transfer-rate option does not affect relationships confined to a single cluster.
relationship-name		obj-name	Name of the relationship.
relationship-progress	**[0..2^63 - 1].**	integer optional	The total number of bytes that have been processed so far for the current activity of the relationship as returned in the relationship-status. This is set only when the relationship-status indicates activity is in progress.
relationship-resource-key		resource-key	Resource key of the relationship.
relationship-state		relationship-state	State of the relationship.
relationship-status		relationship-status	Status of the relationship.
relationship-type		relationship-type	Type of the relationship.
schedule-name		obj-full-name	Full name of the schedule associated with this relationship.
schedule-resource-key		resource-key	Resource key of the schedule associated with this relationship.
source-name		obj-full-name	Full name of storage object where this relationship originates.
source-resource-key		resource-key	Resource key of storage object where this relationship originates.
storage-service-connection-resource-key		resource-key optional	Resource key of the storage service connection.
storage-service-destination-		resource-key	Resource key of the storage service node for the destination of

node-resource-key		optional	this relationship.
storage-service-name		obj-name optional	Name of the storage-service owning this relationship. Not present if this relationship is not owned by any storage service.
storage-service-resource-key		resource-key optional	Resource key of the storage-service owning this relationship. Not present if this relationship is not owned by any storage-service.
storage-service-source-node-resource-key		resource-key optional	Storage object resource key of the storage service node for the source of this relationship.

Element definition: **dp-restore-request-info**

[top]

Restore request information pertaining to a single restore request.

Name	Range	Type	Description
destination-path		string optional	Path to the destination directory where the data is restored. Path is relative to the root of the destination volume and should not contain name of the file. If omitted, data is restored to the root of the volume. For in-place volume restore, a destination-path must not be specified. If both destination-path and use-snapshot-restore-volume inputs are specified and use-snapshot-restore-volume is set to true, then EINVALIDINPUT is returned. Leading '/' is optional in the destination-path. Examples of the destination-path are "dir1/dir2", "/dir1/dir2" or "/". If the destination-path is "/dir1/dir2", it is interpreted as "<destination-volume>/dir1/dir2". '/' must be used in place of '\' when restoring to or from a Windows path. If is-use-snapmirror-restore flag is specified, then the destination-path can only specify a file or a LUN like "/dir1/file1", "file2", "qtree1/lun1", "lun2". Otherwise EINVALIDINPUT is returned.
destination-volume-resource-key		resource-key	Resource key of the target volume to restore the data to.
is-network-compression-enabled		boolean optional	When this flag is set to true, network compression is enabled during snapmirror restore and if set to false network compression is disabled. If using this parameter, is-use-snapmirror-restore must also be set to true. Otherwise, EINVALIDINPUT is returned. This flag is available for source and destinations running Data ONTAP 8.3 or later. The default value for this parameter is 'false'.
is-use-snapmirror-restore		boolean optional	This optional parameter specifies whether to use Data ONTAP's snapmirror-restore API for restoring the selected files or LUNs. If using this parameter, source-path parameter must also be specified whereas destionation-path is optional. The snapmirror-restore API creates a temporary SnapMirror restore relationship. For this reason the destination volume cannot be the destination volume of another SnapMirror relationship. Also the destination volume should be of the type RW Otherwise, EINVALIDINPUT is returned. When this flag is not specified or false, the snapmirror-restore API is not used for restore. The default value for this parameter is 'false'. When this flag is set to true, use-snapshot-restore-volume flag should not be specified or if specified should be set to false. Otherwise, EINVALIDINPUT is

			returned. This flag is available for source and destinations running Data ONTAP 8.3 or later. Otherwise, EINVALIDINPUT is returned. The default value for this parameter is 'false'.
preserve-directory-hierarchy		boolean optional	Flag to preserve the directory hierarchy in the restore path. When true the directory hierarchy is recreated during restore. For example, if the path of the storage object 'c' to be restored is originally located on /a/b/c within the given Snapshot copy and the destination path to where it needs to be stored is /dest, then after restore, 'c' is stored at /dest/a/b/c. When false, 'c' is stored at /dest/c. Default is false.
source-path		string optional	Path within the Snapshot copy of the storage object that is being restored. If use-snapshot-restore-volume flag is specified, then the source-path element must not be specified. Source path is relative to the Snapshot copy location and it can be a directory or a file. If a directory is specified, the files and subdirectories of that directory are restored. Leading '/' is optional in the source-path. Examples of the source-path are "dir1/dir2" or "/dir1/dir2" or "/dir1/dir2/file1". If the source-path is "/dir1/dir2", it is interpreted as "<source-volume>/.snapshot/<snapshot>/dir1/dir2". '/' must be used in place of '\' when restoring to or from a Windows path. If is-use-snapmirror-restore flag is specified, then the following applies: The source-path element must be specified. In this case source path can only specify a file or a LUN of the Snapshot copy. Otherwise EINVALIDINPUT is returned. Upto a maximum of 8 files or LUNs can be restored simultaneously and this multiple file restore option within snapmirror-restore zapi is available on Data ONTAP 8.3 or later. Otherwise, EINVALIDINPUT is returned. Users can provide multiple source-paths as part of multiple dp-restore-request-infos.
source-snapshot-resource-key		resource-key	Resource key of the snapshot copy. This is the immutable natural key to uniquely identify a snapshot on clustered Data ONTAP.
use-snapshot-restore-volume		boolean optional	The flag to use the snapshot-restore-volume API. This flag is applicable only for in-place restoration of a volume from a local Snapshot copy. When true, this flag indicates that Data ONTAP's snapshot-restore-volume API is used for volume restoration. As a side-effect, any Snapshot copies created after this Snapshot copy creation no longer remain on the source volume after the restore. In effect, after the reversion, the volume is in the same state as it was when the Snapshot copy was created. If the volume is in a SnapMirror relationship, then only Snapshot copies that were created after the last SnapMirror update could be used. Otherwise, the job will fail indicating that the volume has a locked Snapshot copy. If this flag is set true, then both source-path and destination-path must not be specified. Otherwise, EINVALIDINPUT is returned. **WARNING**: The storage system will reboot after the restore if the specified snapshot-identifier belongs to the root volume of a node Vserver. See Data ONTAP's guide for the "snapshot-restore-volume" API for more information. When this flag is not specified or false, the snapshot-restore-volume API is not used. Default is false. When this flag is set to true, is-use-snapmirror-restore flag should not be specified or if specified should be set to false. Otherwise, EINVALIDINPUT is returned.

Element definition: **job-id**

[top]

Opaque identifier for a job.

[none]

Element definition: **relationship-state**

[top]

State of the relationship. Possible values are:

- uninitialized
- snapmirrored
- broken-off

[none]

Element definition: **relationship-status**

[top]

Status of the relationship. Possible values are:

- idle
- transferring
- checking
- quiescing
- quiesced
- queued (Data ONTAP 8.2 and higher)
- preparing (Data ONTAP 8.2 and higher)
- waiting (Data ONTAP 8.2 and higher)
- finalizing (Data ONTAP 8.2 and higher)
- aborting (Data ONTAP 8.2 and higher)

[none]

Element definition: **relationship-type**

[top]

Type of a relationship. Possible values are:

- data_protection (aka "mirror")
- load_sharing
- vault (Data ONTAP 8.2 and higher)
- restore (Data ONTAP 8.2 and higher)
- transition_data_protection (Data ONTAP 8.2 and higher)

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **obj-full-name**

[top]

Full name of an object. This typedef is an alias for the builtin ZAPI type **string**. An object full name conforms to all the rules of an obj-name, except that the full name may be up to 255 characters long.

Full names are created by concatenating an object name with any parent object names, so as to create a unique name for an object. The format of full names is as follows:

- **cluster** full names are the either the fully-qualified domain name or the IP address of the cluster.
- **cluster-node** full names are the either the fully-qualified domain name or the IP address of the cluster.
- **aggregate** full names are the cluster-node name and the aggregate name, separated by a colon, e.g. *cluster-node:aggr0*.
- **volume** full names are the vserver name and the volume name, separated by ":/", e.g. *vserver:/volume*. Note this does not include the "/vol" prefix. Volume and aggregate full names are distinguished by the presence of a forward slash after the colon.
- **qtree** full names are the containing volume full name and the qtree name, separated by a slash, e.g. *vserver:/volume/qtree*. The data not contained by any qtree may be represented by "-", e.g. *vserver:/volume/-*.
- **lun** full names are either a volume or qtree full name and the LUN path, separated by a slash, e.g. *vserver:/volume/LUN* or *vserver:/volume/qtree/LUN*.
- **initiator-group** full names are vserver name and the initiator group name, separated by a colon, e.g. *vserver:igroup*.
- **export-policy** full names are vserver name and the policy name, separated by a colon, e.g. *vserver:policy-name*.
- **lif** full names are a cluster, cluster-node, or vserver name and the interface name, separated by a colon, e.g. *cluster-name|cluster-node-name|vserver-name:lif*.
- **port-set** full names are the vserver name and the portset name, separated by a colon, e.g. *vserver:portset*.
- **fcp-target** full names are the cluster-node name and the target name, separated by a colon, e.g. *cluster-node:target*.

For any object not listed above, the obj-name and obj-full-name are identical.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

disk-iter

[top]

Iterate over disks.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • cluster • cluster node • aggregate • disk If resource-filter identifies a disk, that single disk will be returned. If resource-filter resolves to more than one disk, all of them will be returned. If no resource-filter is provided, all disks will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		disk-info[] optional	The list of records.

Errno	Description
EINVALIDTAG	

disk-physical-summary-get

[top]

Retrieve data, parity, spare, and total space usage (by number of disks and bytes) for the specified aggregate, cluster, or cluster node. For a V-Series storage system, data LUNs, spare LUNs and total

LUNs are also reported, if array LUNs exist. For an aggregate, spares and spare space is not reported.
Error Conditions:

- EINVALIDINPUT - When invalid input specified.
- EOBJECTNOTFOUND - When the object-name-or-id does not correspond to an appropriate object.
- ENODISKSFOUND - When no disk information has been retrieved yet for the object-name-or-id.

Input Name	Range	Type	Description
resource-filter		resource-key	Resource key of an object to list the physical summary for. The allowed object types for this argument are: • cluster • cluster node • aggregate

Output Name	Range	Type	Description
disk-info		disk-physical-summary-disk-info optional	Disk information. This item is returned for disk based systems.
lun-info		disk-physical-summary-lun-info optional	LUN information. This item is returned for V-Series systems.

Errno	Description
EINVALIDINPUT	
ENODISKSFOUND	
EOBJECTNOTFOUND	

Element definition: **disk-info**

[top]

Information about a disk.

Name	Range	Type	Description
aggregate-name		obj-full-name optional	Name of the aggregate to which the disk belongs. When the aggregate the disk belongs to is not known or disk is a spare disk, aggregate-name will not be returned. The name is any simple name such as myaggr.
aggregate-resource-key		resource-key optional	Resource key of the aggregate to which the disk belongs. When the aggregate the disk belongs to is not known or disk is a spare disk, aggregate-resource-key will not be returned.
capacity-sectors	**[0..2^63-1]**	integer	Total number of disk sectors on this disk, given in units of 'bytes-per-sector'.
cluster-resource-key		resource-key	Resource key of the cluster to which the disk belongs.
container-type		container-type	Type of overlying disk container.
disk-name		string	Name of the disk. Always present in the output. The name will look like "data disk 0b.18", "parity disk 0b.17", "dparity disk 0b.16" etc. Maximum length of 64 characters.
disk-path-		disk-	

info		path-info[]	Information about the paths of nodes attached to this disk.
disk-resource-key		resource-key	Resource key of the disk. Always present in the output.
disk-size	**[0..2^63-1]**	integer	Disk size in bytes.
disk-type		string	Type of the disk. Maximum length of 64 characters.
disk-uid		string optional	Identifier of the disk. This will be the Unique Identifier (UID) of the disk. When UID of a disk is not known, this field will not be returned. Maximum length of 90 characters. Format of disk UUID will look like: 2000000C:50A9022F:00000000:00000000:00000000:00000000:00000000:00000000:00000000:00000000
effective-disk-type		string	Effective type of the disk. Disks can report different disk-type, but the same effective-disk-type. Disks with the same effective-disk-type are compatible for use within the same aggregate.
failure-reason		string optional	The reason the disk is not in service.
is-permanently-failed		boolean	Whether the disk is in the failed disk registry.
node-name		obj-name	Name of host to which the disk belongs. Always present in the output. The name is any simple name such as myhost.
node-resource-key		resource-key	Resource key of the host to which the disk belongs.
plex-name		string optional	Name of the plex to which the disk belongs. The name is any simple name such as plex0. When the plex the disk belongs to is not known or disk is a spare disk, plex-name will not be returned. Maximum length of 64 characters.
plex-resource-key		resource-key optional	Resource key of the plex to which the disk belongs. When the plex the disk belongs to is not known or disk is a spare disk, plex-resource-key will not be returned.
raidgroup-name		string optional	Name of the raidgroup to which the disk belongs. The name is any simple name such as rg0. When the raidgroup the disk belongs to is not known or disk is a spare disk, raidgroup-name will not be returned. Maximum length of 64 characters.
raidgroup-resource-key		resource-key optional	Resource key of the raidgroup to which the disk belongs. When the raidgroup the disk belongs to is not known or disk is a spare disk, raidgroup-resource-key will not be returned.

Element definition: **disk-physical-summary-disk-info**

[top]

Summary Disk Information.

Name	Range	Type	Description
data-disk-count	**[1..2^31-1]**	integer	Number of data disks.
parity-disk-count	**[1..2^31-1]**	integer	Number of parity disks.
parity-space	**[1..2^96-1]**	integer	Total parity disk space in bytes.
	[1..2^96-		

raw-space	**1]**	integer	Total data disk space in bytes.
spare-disk-count	**[1..2^31-1]**	integer optional	Number of spare disks. This is not returned if the specified object is an aggregate.
spare-space	**[1..2^96-1]**	integer optional	Total spare disk space in bytes. This is not returned if the specified object is an aggregate.
total-disk-count	**[1..2^31-1]**	integer	Number of disks. This is the sum of the data disk, parity disk, and spare disk counts.
total-space	**[1..2^96-1]**	integer	Total disk space in bytes. This is the sum of the raw space, parity space, and spare space.

Element definition: **disk-physical-summary-lun-info**

[top]

Summary LUN Information.

Name	Range	Type	Description
data-lun-count	**[1..2^31-1]**	integer	Number of data LUNs.
raw-space	**[1..2^96-1]**	integer	Total data LUN space in bytes.
spare-lun-count	**[1..2^31-1]**	integer optional	Number of spare LUNs. This is not returned if the specified object is an aggregate.
spare-space	**[1..2^96-1]**	integer optional	Total spare LUN space in bytes. This is not returned if the specified object is an aggregate.
total-lun-count	**[1..2^31-1]**	integer	Number of LUNs. This is the sum of the data LUN and spare LUN counts.
total-space	**[1..2^96-1]**	integer	Total LUN space in bytes. This is the sum of the raw space and spare space.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **container-type**

[top]

Type of overlying disk container. Possible values:

- "aggregate" - Container is an aggregate.
- "broken" - Container is broken pool.
- "labelmaint" - Container is online label maintenance list.
- "maintenance" - Container is disk maintenance center.
- "spare" - Container is spare pool.
- "unassigned" - Disk ownership has not been assigned.
- "unknown" - Container is currently unknown.
- "volume" - Container is a traditional volume.

[none]

Element definition: **disk-path-info**

[top]

Contains per path statistics, errors and other related data.

Name	Range	Type	Description
disk-port		string optional	Disk port associated with this path. Possible values are "A" or "B". Omitted for non-disk target.
disk-port-name		string optional	Disk port name associated with this path. This has the form :, where is either "FC" for FibreChannel, or "SA" for SAS, and is either "A" or "B". Omitted for non-disk target. Possible values: • "FC:A" • "FC:B" • "SA:A" • "SA:B"
node-name		obj-name	Name of the controller with the initiator port for this path. Always present in the output. The name is any simple name such as myhost.
node-resource-key		resource-key	Resource key of the controller with the initiator port for this path.

Element definition: **obj-full-name**

[top]

Full name of an object. This typedef is an alias for the builtin ZAPI type **string**. An object full name conforms to all the rules of an obj-name, except that the full name may be up to 255 characters long.

Full names are created by concatenating an object name with any parent object names, so as to create a unique name for an object. The format of full names is as follows:

- **cluster** full names are the either the fully-qualified domain name or the IP address of the cluster.
- **cluster-node** full names are the either the fully-qualified domain name or the IP address of the cluster.
- **aggregate** full names are the cluster-node name and the aggregate name, separated by a colon, e.g. *cluster-node:aggr0*.
- **volume** full names are the vserver name and the volume name, separated by ":/", e.g. *vserver:/volume*. Note this does not include the "/vol" prefix. Volume and aggregate full names are distinguished by the presence of a forward slash after the colon.
- **qtree** full names are the containing volume full name and the qtree name, separated by a slash, e.g. *vserver:/volume/qtree*. The data not contained by any qtree may be represented by "-", e.g. *vserver:/volume/-*.
- **lun** full names are either a volume or qtree full name and the LUN path, separated by a slash, e.g. *vserver:/volume/LUN* or *vserver:/volume/qtree/LUN*.
- **initiator-group** full names are vserver name and the initiator group name, separated by a colon, e.g. *vserver:igroup*.
- **export-policy** full names are vserver name and the policy name, separated by a colon, e.g. *vserver:policy-name*.

- **lif** full names are a cluster, cluster-node, or vserver name and the interface name, separated by a colon, e.g. *cluster-name*|*cluster-node-name*|*vserver-name:lif*.
- **port-set** full names are the vserver name and the portset name, separated by a colon, e.g. *vserver:portset*.
- **fcp-target** full names are the cluster-node name and the target name, separated by a colon, e.g. *cluster-node:target*.

For any object not listed above, the obj-name and obj-full-name are identical.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

APIs in Category: **event**

API version 9.12

event-acknowledge

[top]

Acknowledge events. This terminates repeated notifications due to that event.

Input Name	Range	Type	Description
event-id-list		event-id-type[]	The event identifiers to be acknowledged.

Output Name	Range	Type	Description
events-acknowledged-list		event-action-info[]	List of timestamps for acknowledged events along with any errors.

Errno	Description
EINVALIDINPUT	
ENOSUCHEVENT	
EALREADYACKWED	

event-assign

[top]

Assign the events to the specified user.

Input Name	Range	Type	Description
event-assignee-name		string	Name of the user to which the events are to be assigned.
event-id-list		event-id-type[]	The event identifiers to be assigned.

Output Name	Range	Type	Description
assigned-events-list		event-action-info[]	List of timestamps for assigned events and error codes in case of failure.

Errno	Description
ENOSUCHEVENT	
EINVALIDINPUT	
ENOTFOUNDUSER	
E_INVALID_EVENT_ASSIGNEE	

event-class-iter

[top]

Iterate and list event-classes.

Input Name	Range	Type	Description
event-generation-enabled		boolean optional	If false, event generation is disabled for this event class.
impact-level		event-impact-level optional	Impact level of generated event.
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
severity		event-severity optional	Severity of generated event.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		event-class-info[] optional	The list of records.

event-class-modify

[top]

Modify one or more properties of an event class.

Error Conditions:

- ENOSUCHEVENTCLASS - If no matching id or class is found.

Input Name	Range	Type	Description
event-class-name-or-id		object-name-or-id	Id or unique-name of the event-class.
event-generation-enabled		boolean	If false, event generation will be disabled for this class.

Errno	Description
ENOSUCHEVENTCLASS	

event-iter

[top]

Iterate over events ordered by event-id.

Input Name	Range	Type	Description
event-id		integer optional	The identifier of the event that is to be listed.
event-impact-levels		event-impact-level[] optional	Lists events of the specified impact levels.
event-severities		obj-status[] optional	Lists events of the specified severities.
event-state-filter-list		event-state[] optional	List events of specified states. This filter can be specified in addition to other filters. if not specified, events of all states are returned.
event-type-filter-list		event-type-filter[] optional	If specified, the list of event types (interpreted as prefixes) will filter which events are returned. Events not matching will not be returned. Default is empty, meaning that all event types match.
greater-than-id		integer optional	If specified, the API will only return events whose identifier is greater than this value. Default is 0.
include-notes		boolean optional	If true, specifies that events should be returned with their related notes. The default is false, meaning that events are returned without notes.
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
source		resource-key optional	Lists events against the specified source. If the provided resource key specifies a group, lists events against all members in that group.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.
time-range		event-timestamp-range optional	Lists all events which were generated in the range specified. Default is all events. time-out will be ignored if time-range is set.
			Number of seconds after which the API should terminate, if no events are received matching the input criteria. If the value is 0, or

timeout		integer optional	not specified, the API will terminate immediately (acting as an instantaneous poll for events). If the timeout expires with no matching events, the API returns successfully with an empty list of events. If a specific event-id is specified, then the timeout value is ignored. If time-range is set, timeout is also ignored.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		event-info[] optional	The list of records.

Errno	Description
EINVALIDEVENTSEVERITY	
EINVALIDEVENTIMPACTLEVEL	
ENOSUCHEVENT	

event-purge

[top]

Purges the events that are older than the specified interval.

Input Name	Range	Type	Description
interval	**[1..2^16-1]**	integer	The time interval in seconds. All non-current events with timestamp prior to the given interval will be purged.

Output Name	Range	Type	Description
cutoff-timestamp		timestamp	The timestamp prior to which all the non-current events were purged.
purged-events-count	**[1..2^16-1]**	integer	Number of events purged.

Errno	Description
EAPIMISSINGARGUMENT	

event-resolve

[top]

Resolve events. Terminates repeated notifications due to the event.

Input Name	Range	Type	Description
event-id-list		event-id-type[]	The event identifiers to be resolved.

Output Name	Range	Type	Description
events-resolved-list		event-action-info[]	Lists of timestamps for resolved ids along with any error returns.

Errno	Description
EINVALIDINPUT	
ENOSUCHEVENT	

event-status-change-list-iter-end

[top]

event-status-change-list-iter-end is used to indicate that the temporary store used to support the event-status-change-list-iter-next API for the particular tag is no longer necessary.

Input Name	Range	Type	Description
tag		string	An internal opaque handle.

Errno	Description
EINVALIDTAG	

event-status-change-list-iter-next

[top]

The event-status-change-list-iter-next API is used to iterate over the list of events stored in the temporary store created by the event-status-change-list-iter-start API.

A cursor is maintained pointing to the last record returned. Subsequent calls to this API will return the records after the cursor up to the specified "maximum" or number of actual records left.

Input Name	Range	Type	Description
maximum		integer	The maximum number of events to return.
tag		string	An opaque handle used to identify the temporary store created by event-status-change-list-iter-start.

Output Name	Range	Type	Description
events		event-info[]	Array of events.
records	**[0..(2^31)-1]**	integer	The number of records actually returned.

Errno	Description
EINVALIDTAG	

event-status-change-list-iter-start

[top]

List events that had status changes (acknowledged or resolved) within the specified time range.

The event-status-change-list-iter-* set of APIs are used to retrieve the list of events that had status changes.

The event-status-change-list-iter-start API is used to load the list of events into a temporary store. The API returns a tag to temporary store so that subsequent APIs can be used to iterate over the list in the temporary store.

The returned list of events will be sorted according to when they had their status changed (either eventAcked timestamp or eventresolved timestamp). An event that's both acked and resolved within the requested timeframe would appear twice in the returned list of events, because those would count as 2 status changes, and appear in the returned list based on acked timestamp and resolved timestamp respectively.

Note that, depending on the input parameters, this API may take up to "timeout" seconds to return. Subsequent calls to event-status-change-list-iter-next() will return immediately.

Input Name	Range	Type	Description
max-events	**[0..2^31-1]**	integer optional	If specified, this is the maximum number of events that the client wishes to receive at once. If set to zero, return all events. The default value of this parameter is 50,000.
time-range		event-timestamp-range	Lists all events which were generated in the range specified. If the end-time of the time-range is not sometime in the future, time-out will be ignored.
timeout		integer optional	Number of seconds after which the API should terminate, if no events are received matching the input criteria. If the value is 0, or not specified, the API will terminate immediately (acting as an instantaneous poll for events). If the timeout expires with no matching events, the API returns successfully with an empty list of events.

Output Name	Range	Type	Description
records		integer	The number of events matching the specified input criteria. This is the number of records that will be returned by subsequent calls to event-status-change-list-iter-next().
tag		string	An opaque handle you must pass to event-status-change-list-iter-next() and event-status-change-list-iter-end() to refer to this list of events.

event-unassign

[top]

Unassign the owner of an event or a set of events.

Input Name	Range	Type	Description
event-id-list		event-id-type[]	Identifiers of the events whose owner has to be unassigned.

Output Name	Range	Type	Description
unassigned-		event-action-	List consisting only identifiers of events which are unassigned. For any identifiers which were unable to be processed, an error codes with error

events-list		info[]	message specifying the failure reason will also be in the output.

Errno	Description
ENOSUCHEVENT	
EINVALIDINPUT	

Element definition: **event-action-info**

[top]

Result of action taken on event. Timestamp returned on success, and error code on failure.

Name	Range	Type	Description
error-code	**[1..2^32-1]**	integer optional	Error code corresponding to errno returned from event acknowledge/resolve/assign. Absent on success.
error-message		string optional	Error message returned from event acknowledge/resolve/assign. Absent on success.
event-id	**[0..2^32-1]**	integer	The input event identifier.
timestamp	**[0..2^32-1]**	integer optional	Timestamp when the event was acknowledged/resolved/assigned. Timestamps absent for IDs that can not be found, or have already been acknowledged/resolved.
warning-message		string optional	Warning message returned from event acknowledge/resolve/assign.

Element definition: **event-class-info**

[top]

Event class structure.

Name	Range	Type	Description
event-class-id		integer	ID of the event class.
event-generation-enabled		boolean	If false, event generation is disabled for this event class.
impact-level		event-impact-level	Impact level of generated event.
pretty-name		string	Pretty name of the event class.
severity		event-severity	Severity of generated event.
unique-name		string	Name of the event class.

Element definition: **event-id-type**

[top]

Event identifier.

[none]

Element definition: **event-impact-level**

[top]

Impact level of the event.

Possible values are:

- incident: An error or critical event.
- risk: A warning event which could lead to an incident.
- event: A simple event from Data ONTAP.

[none]

Element definition: **event-info**

[top]

Event information structure

Name	Range	Type	Description
event-about		string	Description of the event type.
event-acknowledged-timestamp		timestamp optional	Timestamp when event was acknowledged.
event-acknowledged-user		string optional	User who acknowledged the event.
event-arguments		key-value-pair[] optional	Argument list for this particular event. Present only if include-event-arguments was set to true in the event-list-iter-start call. If the event has no arguments, this element will be empty. The arguments returned are dependent on the event type and status.
event-assigned-user		string optional	User to whom the event is assigned.
event-category		event-category optional	Category of the event.
event-condition		string	Condition of the event.
event-id	**[1..2^31 - 1]**	integer	Id of the event.
event-impact-area		event-impact-area	Impact area of the event.
event-impact-level		event-impact-level	Impact level of the event.
event-name		string	Name of the event.
event-notes		event-note[] optional	List of notes for this particular event. Present only if include-notes is set to true in the event-list-iter-start call. If the event has no notes, this element will be empty.
event-obsolete-		string optional	Condition that caused event to become obsolete. An event becomes obsolete when a newer event of the same type is

condition			generated.
event-obsolete-timestamp		timestamp optional	Timestamp when event became obsolete. An event becomes obsolete when a newer event of the same type is generated.
event-resolved-timestamp		timestamp optional	Timestamp when event was resolved.
event-resolved-user		string optional	User who resolved the event.
event-severity		event-severity	Severity of the event.
event-source-name		string	Name of the source of the event. Example: "storage01.example.com".
event-source-resource-key		resource-key	Resource key of the source of the event.
event-source-type		resource-type	Type of object that generated the event.
event-state		event-state	State of the event.
event-time		integer	Time when the event was fired.
event-type		string	Type or class to which the event belongs to.

Element definition: **event-severity**

[top]

Severity of the event.

Possible values are:

- normal
- information
- warning
- error
- critical

[none]

Element definition: **event-state**

[top]

State of the event.

Possible values are: 'new', 'acknowledged', 'resolved', 'obsolete'.

- new: The state of an event when it is first triggered.
- acknowledged: The state of an event which has been acknowledged by a user (and is being worked on). Both new and acknowledged events identify events which are open and need user attention.
- resolved: The state of an event after a user explicitly resolves it.
- obsolete: The state of an event which is no longer deemed valid by the system. Both resolved and obsolete events together identify events that no longer need user attention.

[none]

Element definition: **event-timestamp-range** [top]

range of event timestamps

Name	Range	Type	Description
end-time		integer	End timestamp, in seconds elapsed since midnight on January 1, 1970.(UTC)
start-time		integer	Start timestamp, in seconds elapsed since midnight on January 1, 1970. (UTC)

Element definition: **event-type-filter** [top]

Array of event filters.

Name	Range	Type	Description
event-filter		string	name of the event-filter.

Element definition: **obj-status** [top]

A status value which can be associated with an object. This typedef is an alias for the builtin ZAPI type **string**. The severity associated with an event has this type.

Possible values are: 'normal', 'warning', 'error', 'critical'.

- normal: An object has normal status when it is working within the thresholds specified.
- warning: An object has the warning status when an event related to the object occurred that an administrator should know about. The event will not cause service disruption.
- error: An object has error status when it does not cause any service disruption, but it may affect performance.
- critical: An object has critical status when it is still performing, but service disruption may occur if corrective action is not taken immediately.

In some contexts, it is important that severities are ordered (as above). For example, an alert might be triggered if an event with a given severity "or worse" occurs. In this example, worse means "after" in the list above.

[none]

Element definition: **object-name-or-id** [top]

Name or ID of an object.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **timestamp**

[top]

Seconds since 1/1/1970 in UTC.

[none]

Element definition: **event-category**

[top]

Severity of the event.

Possible values are:

- aggregate_availability
- aggregate_capacity
- controller_availability
- disk_failure
- disk_shelf_availability
- lun_availability
- lun_capacity
- node_availability
- node_hardware_failure
- port_failure
- qtree_capacity
- volume_availability
- volume_capacity
- vserver_data_availability

[none]

Element definition: **event-impact-area**

[top]

Impact area of the event.

Possible values are:

- availability
- capacity
- configuration

performance
- protection

[none]

Element definition: **event-impact-level** [top]

Impact level of the event.

Possible values are:

- incident: An error or critical event.
- risk: A warning event which could lead to an incident.
- event: A simple event from Data ONTAP.

[none]

Element definition: **event-note** [top]

Event note.

Name	Range	Type	Description
author		string	Author of the note.
content		string	Content of the note.
timestamp		timestamp	Timestamp when the note was added.

Element definition: **key-value-pair** [top]

The key/value for a generic object attribute.

Name	Range	Type	Description
key		string	key of the generic object attribute
value		string	Value of the generic object attribute

Element definition: **resource-type** [top]

A canonical type name for a managed resource. Includes both Data ONTAP and OnCommand resource types. Valid Data ONTAP resource types:

- aggregate
- objectstore_config
- capacity_tier

- cifs_share
- cifs_share_acl
- cluster
- cluster_node
- disk
- export_policy
- export_rule
- fcp_lif
- fcp_port
- flash_device
- igroup
- iscsi_portal_group
- lun
- lun_mapping
- network_lif
- network_port
- ontap_job_schedule
- plex
- portset
- qtree
- raid_group
- routing_group
- service_processor
- sis_policy
- snap_mirror
- snapshot
- snapshot_policy
- snapshot_policy_schedule
- storage_class
- storage_shelf
- volume
- volume_move
- vserver
- vserver_name_mapping

Valid OnCommand resource types:

- management_station
- resource_pool
- service_workflow
- storage_service
- storage_service_connection
- storage_service_node

[none]

Element definition: **timestamp**

[top]

Seconds since 1/1/1970 in UTC.

[none]

igroup-iter

[top]

Iterate over initiator groups.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • cluster • vserver • lun • igroup If resource-filter identifies a igroup, that single igroup will be returned. If resource-filter resolves to more than one igroup, all of them will be returned. If no resource-filter is provided, all igroups will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		igroup-info[] optional	The list of records.

Element definition: **igroup-info**

[top]

Information about one igroup.

Name	Range	Type	Description
alua-enabled		boolean optional	Indicates whether ALUA (Asymmetric Logical Unit Access) features are enabled for luns mapped to this initiator group.

igroup-name		obj-name	Name of the igroup.
igroup-type		string	Specifies the type (protocol) of the igroup. Possible values are: 'fcp', 'iscsi', and 'mixed'
initiators		initiator-name[]	Lists the initiators belonging to this igroup.
os-type		string	Specifies the operating system of the igroup. Possible values are: 'windows', 'linux', 'aix', 'hpux', 'solaris', 'hyper_v', 'vmware', 'xen', 'netware', 'openvms' and 'default'.
portset-name		obj-name optional	Name of the portset to which the igroup is bound to. This element will be present only if the igroup is bound to a portset. Portset contains a set of iSCSI Target Portal Groups and/or FCP logical interfaces. The initiators in the igroup can access the mapped LUNs only through the ports added to the portset.
portset-resource-key		resource-key optional	Resource key of the portset to which the igroup is bound to. This element will be present only if the igroup is bound to a portset.
resource-key		resource-key	Resource key of this igroup.
vsa-enabled		boolean optional	Indicates whether this initiator group has Volume Set Addressing (VSA) enabled or disabled.
vserver-name		obj-name	Name of the Vserver to which the igroup belongs.
vserver-resource-key		resource-key	Resource key of the Vserver to which the igroup belongs.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **initiator-name**

[top]

Name of the initiator.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').

- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

APIs in Category: **job**

API version 9.12

Job operations include submission, cancellation, deletion, and listing of management jobs, meaning jobs which perform some task on or gather some data from a managed system.

job-abort

[top]

Abort a running management job.

Input Name	Range	Type	Description
job-id		job-id	Identifier for a management job.

Errno	Description
EJOBDOESNOTEXIST	
EJOBALREADYCOMPLETED	

job-delete

[top]

Delete the results of a completed job.

Input Name	Range	Type	Description
job-id		job-id	Identifier for a management job.

Errno	Description
EJOBDOESNOTEXIST	

job-iter

[top]

Retrieve information about management jobs.

- More than one job may be returned.
- If job-id or task-id is specified, it should be the only query criterion. If other inputs are specified as multiple criteria, they are ANDed. In other words, the intersection of the jobs specified by the matching criteria is returned.

Input Name	Range	Type	Description
		job-id	Specify the job identity matching criteria. If other inputs are specified

job-id		optional	with job-id, EINVALIDINPUT error is returned.
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
state		job-state optional	Specify the job state matching criteria.
status		job-status optional	Specify the job status matching criteria.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.
task-id		job-task-id optional	Specify the task identity matching criteria. If other inputs are specified with task-id, EINVALIDINPUT error is returned.
type		job-type optional	Specify the job type matching criteria.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		job-info[] optional	The list of records.

Errno	Description
EINVALIDINPUT	

job-task-iter

[top]

Starts iteration to list tasks and their status.

- More than one tasks may be returned.
- The criteria are ANDed. That is: we return the intersection of the tasks specified by the matching criteria if input is specified other than task-id. No other input is allowed with Task-id input, in case other input is specified with task-id then EINVALIDINPUT error is returned. If specified job-id is not found then EJOBDOESNOTEXIST error is returned.

Input Name	Range	Type	Description
job-id		job-id optional	Job identity matching criteria.
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.

state		task-state optional	Task state matching criteria.
status		task-status optional	Task status matching criteria.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.
task-id		job-task-id optional	Task identity matching criteria.
task-type		string optional	Task type matching criteria.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		job-task-info[] optional	The list of records.

Errno	Description
EINVALIDINPUT	
EOBJECTNOTFOUND	
EJOBDOESNOTEXIST	

job-wait-for-state

[top]

Blocks until a specified timeout period elapses or until a job state is reached by at least one job.

Input Name	Range	Type	Description
job-list		job-id[]	List of job-ids to wait for to reach the specified state. If a listed job-id is not found, then EJOBDOESNOTEXIST is returned along with the first invalid/missing job-id. The API blocks until at least one of jobs reaches the specified state.
state		job-state optional	Specify the job state to wait for. If state specified is other than 'completed' then E_FILTER_ARGUMENT_NOT_IMPLEMENTED error will be returned. Default is 'completed'. The 'completed' state implies all terminal job states viz. 'completed' and 'aborted'.
wait-for-state-timeout	**[0..2^31-1]**	integer	Number of seconds to wait for job(s) to reach the specified state. The API blocks until at least one of the jobs reaches the specified state. Jobs already in the specified state will cause this API to return immediately with the corresponding list of job-ids. If timeout occurs before any job reaches the specified state, the API returns an empty job-id list. If wait-for-state-timeout is specified as '0' then API returns immediately without blocking.

Output Name	Range	Type	Description
job-list		job-id[]	List of job-ids corresponding to the job(s) matching the specified state.

Errno	Description
EINVALIDINPUT	
EJOBDOESNOTEXIST	
E_OPERATION_TIMED_OUT	
E_FILTER_ARGUMENT_NOT_IMPLEMENTED	

Element definition: **job-id**

[top]

Opaque identifier for a job.

[none]

Element definition: **job-info**

[top]

Information about a job. A job may be triggered by an explicit user action or through scheduled operations. Each info includes current state of execution and status of the operation.

Name	Range	Type	Description
end-time		timestamp optional	Time at which the job completed.
estimated-percent-complete		integer optional	Estimated percent of the work completed for a job. Hold the aggregate estimated percent of work completed based on the completion of all job tasks. This information is an estimate and may not always be available.
estimated-time-to-complete		integer optional	Estimated time (measured in seconds) for job completion. This will hold the aggregate estimated time to completion based on time to completion of all job tasks and in accordance to their execution dependency order. This information is an estimate and may not always be available.
job-description		string	Description of the job.
job-detail		job-detail optional	Additional information relevant to job base on its type.
job-id		job-id	Identifier of this job.
job-name		string	Name of the job.
job-state-events		state-event-info[]	Job life cycle execution state audit log
start-time		timestamp optional	Time at which the job started.
state		job-state	State of the job indicating if the job is currently running etc.
status		job-status	Status of the job indicating the overall success or failure. Overall status is computed based on status of the actions performed as part of the job.

type		job-type	Type of the job.

Element definition: **job-state** [top]

The state of the job. The possible values are:

- running
- completed
- aborting
- aborted

[none]

Element definition: **job-status** [top]

The status of the job. Rolled up status of the whole job based on all tasks. The possible values are:

- normal
- warning
- partial_failures
- error

[none]

Element definition: **job-task-id** [top]

Opaque identifier for a job task.

[none]

Element definition: **job-task-info** [top]

Information about the action for executing one task.

Name	Range	Type	Description
dependency-list		job-task-id[]	List of job-task-id(s) on which this task depends.
end-time		timestamp optional	Time at which the task completed.
estimated-percent-complete		integer optional	Estimated percent of the work completed for a task. Some task allow tracking of progress, such as in the case of data transfer. For this type of tasks, this element will hold the estimated percent completed.

estimated-time-to-complete		integer optional	Estimated time for task completion (measured in seconds).
failure-reason		string optional	Reason causing task failure.
job-id		job-id	Identifier of the parent job.
messages		task-message-info[] optional	Messages generated by the task during execution.
object-interactions		task-interaction-object[] optional	Objects the task interacted with during execution.
start-time		timestamp optional	Time at which the task started.
state		task-state	State of the task indicating the task current execution state.
status		task-status	Status of the task indicating the task current execution status.
task-description		string	Description of the task e.g. "Add Vault node to the storage service Gold Mirror".
task-detail		task-detail optional	Additional information relevant to task based on its type.
task-id		job-task-id	Identifier of the task.
task-state-events		state-event-info[]	Task life cycle execution state audit log.
type		task-type	Type of the task. Each task type will have additional type specific data.

Element definition: **job-type**

[top]

Job types. The possible values are:

- protect_storage_service_subscribe
- protect_storage_service_unsubscribe
- protect_storage_service_update
- protect_storage_service_conform
- protect_storage_service_import
- protect_storage_service_cleanup
- protect_storage_service_modify
- protect_storage_service_destroy
- protect_restore
- volume_create
- monitor
- active_management
- compensation
-

[none]

Element definition: **task-state**

[top]

The state of the task. The possible values are:

- waiting
- running
- completed

[none]

Element definition: **task-status**

[top]

The status of the task. Valid values are:

- normal
- error
- skipped

[none]

Element definition: **job-detail**

[top]

Detail information specific to the task base on task type.

Name	Range	Type	Description
storage-service-job-info		storage-service-job-info optional	Information specific to storage service job. Returned only when the job-type is • protect_storage_service_subscribe • protect_storage_service_unsubscribe • protect_storage_service_update • protect_storage_service_conform • protect_storage_service_import • protect_storage_service_cleanup • protect_storage_service_modify • protect_storage_service_destroy

Element definition: **job-id**

[top]

Opaque identifier for a job.

[none]

Element definition: **job-task-id**

[top]

Opaque identifier for a job task.

[none]

Element definition: **state-event-info**

[top]

Information specific to execution state audit events.

Name	Range	Type	Description
event		string	Possible values can be: • submitted • started • canceled • aborted • completed
timestamp		timestamp	Time at which the state changed.

Element definition: **task-detail**

[top]

Detailed information specific to the task based on task type.

Name	Range	Type	Description
relationship-info		relationship-task-info optional	Information specific to setting up relationships. Returned only when the task-type is: • protect_relationship_create • protect_relationship_initialize • protect_relationship_update • protect_relationship_transfer_progress • protect_relationship_destroy • protect_relationship_abort • protect_relationship_quiesce

Element definition: **task-interaction-object**

[top]

An object that a task interacted with.

Name	Range	Type	Description
task-interaction-object-name		string	Name of the object that the task reported interaction with. Example: "storage01.example.com".
task-interaction-object-resource-key		resource-key	Resource key of the object that the task reported interaction with.

task-interaction-object-type		resource-type	Type of object that the task reported interaction with.

Element definition: **task-message-info**

[top]

Task message information.

Name	Range	Type	Description
message		string	Generated message.
timestamp		timestamp optional	Timestamp when the message was generated.
type		task-message-type	Type of message.

Element definition: **task-type**

[top]

Task types. Valid values are:

- protect_secondary_provision
- protect_secondary_destroy
- protect_secondary_offline
- protect_relationship_create
- protect_relationship_initialize
- protect_relationship_modify
- protect_relationship_update
- protect_relationship_transfer_progress
- protect_relationship_destroy
- protect_relationship_abort
- protect_relationship_quiesce
- protect_restore
- protect_restore_ndmp_utility
- protect_storage_service_utility
- protect_vserver_peering
- monitor_discover
- active_quota_management
- empty
- administrative
- compensation
-

[none]

Element definition: **timestamp**

[top]

Seconds since 1/1/1970 in UTC.

[none]

Element definition: **relationship-task-info**

[top]

Details specific to the SnapMirror relationship operations.

Name	Range	Type	Description
connection-resource-key		resource-key optional	Resource key of the destination node in topology graph of this storage service.
destination-member-name		obj-name optional	Name of the destination object for this relationship.
destination-member-resource-key		resource-key optional	Resource key of the destination object for this relationship.
destination-node-name		obj-name optional	Name of the destination node in topology graph of this storage service.
destination-node-resource-key		resource-key optional	Resource key of the destination node in topology graph of this storage service.
last-progress-info		transfer-progress-task-info optional	Most recent progress information for a data transfer. Returned only when the task-type is • protect_relationship_transfer_progress
relationship-resource-key		resource-key optional	Resource key of the relationship between source-member and destination-member. Returned only when the task-type is: • protect_relationship_initialize • protect_relationship_update • protect_relationship_transfer_progress • protect_relationship_destroy • protect_relationship_abort • protect_relationship_quiesce
source-member-name		obj-name optional	Name of the source object for this relationship.
source-member-resource-key		resource-key optional	Resource key of the source object for this relationship.
source-node-name		obj-name optional	Name of the source node in topology graph of this storage service.
source-node-resource-key		resource-key optional	Resource key of the source node in topology graph of this storage service.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **resource-type**

[top]

A canonical type name for a managed resource. Includes both Data ONTAP and OnCommand resource types. Valid Data ONTAP resource types:

- aggregate
- objectstore_config
- capacity_tier
- cifs_share
- cifs_share_acl
- cluster
- cluster_node
- disk
- export_policy
- export_rule
- fcp_lif
- fcp_port
- flash_device
- igroup
- iscsi_portal_group
- lun
- lun_mapping
- network_lif
- network_port
- ontap_job_schedule
- plex
- portset
- qtree
- raid_group
- routing_group
- service_processor
- sis_policy
- snap_mirror
- snapshot
- snapshot_policy
- snapshot_policy_schedule
- storage_class
- storage_shelf
- volume
- volume_move
- vserver
- vserver_name_mapping

Valid OnCommand resource types:

- management_station
- resource_pool
- service_workflow
- storage_service
- storage_service_connection
- storage_service_node

[none]

Element definition: **storage-service-job-info**

[top]

Detail information specific to the task base on task type.

Name	Range	Type	Description
storage-service-id		obj-id	ID of storage service used in operation.
storage-service-name		obj-name	Name of storage service used in operation.
subscription-context		string optional	Subscription context used in the request which generated the job. Only present for a jobs generated with a request containing specific subscription context.

Element definition: **task-message-type**

[top]

Task message types. The possible values are:

- info
- warning
- error

[none]

Element definition: **obj-id**

[top]

Identification number (ID) for an object. This typedef is an alias for the builtin ZAPI type **integer**. Object IDs are unsigned integers in the range [1..2^31 - 1]. In some contexts, an object ID is also allowed to be 0, which is interpreted as a null value, e.g., a reference to no object at all.

The ID for an object is always assigned by the system; the user is never allowed to assign an ID to an object. Therefore, an input element of type **obj-id** is always used to refer to an existing object by its ID. The ZAPI must specify the object's object type (e.g. cluster, volume, aggregate, etc.). Some ZAPIs allow the object to be one of several different types.

If the value of an **obj-id** input element does not match the ID of any existing object of the specified type or types, then typically the ZAPI fails with error code **EOBJECTNOTFOUND**. A ZAPI may deviate from this general rule, for example, it may return a more specific error code. In either case, the ZAPI specification must document its behavior.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it

may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **transfer-progress-task-info**

[top]

Data transfer progress update details.

Name	Range	Type	Description
ontap-transfer-job-id		integer optional	The Data ONTAP job identifier of the current relationship transfer.
progress-last-updated		timestamp	A timestamp indicating when the progress of the transfer was last updated.
relationship-progress		integer optional	The total number of bytes that have been processed so far for the current transfer activity of the relationship.
snapshot-checkpoint		integer optional	The number of bytes transferred as recorded for the checkpoint of the current or most recent transfer snapshot.
snapshot-progress		integer optional	The number of bytes transferred for the transfer-snapshot.
transfer-snapshot		string optional	The name of the current snapshot copy being transferred.

Element definition: **timestamp**

[top]

Seconds since 1/1/1970 in UTC.

[none]

lun-iter

[top]

Iterate over a list of lun objects.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • resource-group • cluster • vserver • aggregate • volume • qtree • lun If resource-filter identifies a lun, that single lun will be returned. If resource-filter resolves to more than one lun, all of them will be returned. If no resource-filter is provided, all luns will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		lun-info[] optional	The list of records.

Errno	Description
EOBJECTNOTFOUND	
EOBJECTAMBIGUOUS	
EINVALIDINPUTERROR	

Element definition: **lun-info**

[top]

Information about a lun.

Name	Range	Type	Description
alignment		string	Alignment of the LUN. Possible values: • aligned • misaligned • partial_writes • indeterminite
comment		string	User-specified comment for the LUN. This field is unavailable when the LUN is in a snapshot or while the LUN fenced for a restore operation.
igroups		igroup-reference[] optional	list of igroups to which the LUN is mapped to. Returns emtpy array incase the LUN is not mapped to any igroup.
is-lun-space-reserved		boolean	Indicates if the space for LUN is reserved the containing volume.
is-space-alloc-enabled		boolean	Whether or not the LUN has space allocation enabled. This field is unavailable when the LUN is in a snapshot or while fenced for a restore operation.
lun-class		string	The class of the LUN. Possible values: • regular - The LUN is intended for normal blocks access • protocol_endpoint - The LUN is a VMware vvol protocol endpoint • vvol - The LUN is a VMware vvol data LUN
lun-path		obj-name	Path name of the lun including the volume or qtree where the lun exists. The name will be similar to myvol/mylun or myvol/myqtree/mylun.
lun-size	**[0..2^63-1]**	integer	Lun size in bytes.
lun-status		obj-status	Current status of the lun based on all events
lun-used-space	**[0..2^63-1]**	integer	Number of bytes used by the LUN.
mapped		boolean	Whether or not the LUN is mapped to any initiators. "true" if mapped, "false" otherwise. This field is not applicable to LUNs where the lun-class attribute is set to 'vvol'.
multiprotocol-type		string	The image type of the lun. This value determines the proper alignment settings for the desired host filesystem layout. Possible values: • aix - The LUN will be used to store an AIX filesystem. • hpux - The LUN will be used to store an HP-UX filesystem. • hyper_v - The LUN will be used to store Hyper-V VHDs (Virtual Hard Disks). • image - The default type indicating no assumptions will be made about the data stored in the LUN. • linux - The LUN will be used to store a Linux filesystem with no partition table. • netware - The LUN will be used to store a Netware filesystem. • openvms - The LUN will be used to store an OpenVMS filesystem. • solaris - The LUN will be used to store a Solaris filesystem, in a single slice partition.

			• solaris_efi - The LUN will be used to store a Solaris filesystem with an EFI partition table. • vmware - The LUN will be used to store a VMware Virtual Machine File System (VMFS) containing Virtual Machine Disk Files (VMDKs). • windows - The LUN will be used to store a Windows filesystem with a Master Boot Record (MBR) partition table. • windows_2008 - The LUN will be used to store a Windows filesystem with a Master Boot Record(MBR) partition table on Windows 2008 or later. • windows_gpt - The LUN will be used to store a Windows filesystem with a GUID Partition Table (GPT).
qtree-name		obj-name optional	Name of qtree on which the lun resides. Present in the output only if the lun resides on a qtree. The name is any simple name such as myqtree.
qtree-resource-key		resource-key	Resource key of the Qtree to which the lun belongs.
resource-key		resource-key	Resource key of this lun.
serial-number		string	Serial number of the LUN. The serial number is a 12-character string formed of upper and lower-case letters, numbers, slashes (/), and hyphen (-) characters.
volume-name		obj-name optional	Name of volume on which the lun resides. The name is any simple name such as myvol. volume-name is not returned if the lun belongs to a qtree and the authenticated admin does not have the required capability.
volume-resource-key		resource-key	Resource key of the Volume to which the lun belongs.
vserver-name		obj-name	Name of the vserver on which the lun resides. Always present in the output. The name is any simple name such as myvserver.
vserver-resource-key		resource-key	Resource key of the Vserver to which the lun belongs.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **igroup-reference**

[top]

Name and Id of an igroup.

Name	Range	Type	Description
igroup-mapping-id		obj-id	ID of the mapping of this igroup to the corresponding LUN.
igroup-name		obj-name	Name of the igroup.
igroup-resource-key		resource-key	Resource key of the igroup.

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **obj-status**

[top]

A status value which can be associated with an object. This typedef is an alias for the builtin ZAPI type **string**. The severity associated with an event has this type.

Possible values are: 'normal', 'warning', 'error', 'critical'.

- normal: An object has normal status when it is working within the thresholds specified.
- warning: An object has the warning status when an event related to the object occurred that an administrator should know about. The event will not cause service disruption.
- error: An object has error status when it does not cause any service disruption, but it may affect performance.
- critical: An object has critical status when it is still performing, but service disruption may occur if corrective action is not taken immediately.

In some contexts, it is important that severities are ordered (as above). For example, an alert might be triggered if an event with a given severity "or worse" occurs. In this example, worse means "after" in the list above.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **obj-id**

[top]

Identification number (ID) for an object. This typedef is an alias for the builtin ZAPI type **integer**. Object IDs are unsigned integers in the range [1..2^31 - 1]. In some contexts, an object ID is also allowed to be 0, which is interpreted as a null value, e.g., a reference to no object at all.

The ID for an object is always assigned by the system; the user is never allowed to assign an ID to an object. Therefore, an input element of type **obj-id** is always used to refer to an existing object by its ID. The ZAPI must specify the object's object type (e.g. cluster, volume, aggregate, etc.). Some ZAPIs allow the object to be one of several different types.

If the value of an **obj-id** input element does not match the ID of any existing object of the specified type or types, then typically the ZAPI fails with error code **EOBJECTNOTFOUND**. A ZAPI may deviate from this general rule, for example, it may return a more specific error code. In either case, the ZAPI specification must document its behavior.

[none]

net-interface-iter

[top]

Iterate over logical interfaces. This API lists ethernet interfaces only. It does not list FCP interfaces.

Input Name	Range	Type	Description
is-cluster-interface-only		boolean optional	If this flag is set, only the ethernet interfaces owned by the cluster are returned. Interfaces owned by a vserver or cluster node will not be returned. This input can only be provided with a resource-filter value for a cluster.
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • cluster • vserver • cluster_node • network_lif If resource-filter identifies a network_lif, that single network_lif will be returned. If resource-filter resolves to more than one network_lif, all of them will be returned. If no resource-filter is provided, all network_lifs will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		net-interface-info[] optional	The list of records.

Element definition: **net-interface-info**

[top]

Information of about one interface.

Name	Range	Type	Description
address-type		string optional	Type of IP address configured for the interface. Possible values are: • ipv4 • ipv6
administrative-status		string optional	The administrative status of the interface. The administrative status can differ from the operational status; for instance, if you specify the status as up but a network problem prevents the interface from functioning, the operational status remains as down. Possible values: • 'up', • 'down', • 'unknown'
cluster-name		string	Name of the cluster that the interface resides on.
current-node		string optional	Name of the interface's current node.
current-port		string optional	Name of the interface's current port.
data-protocols		data-protocol[] optional	The list of data protocols configured on the interface.
failover-group		string optional	The failover group name of this interface.
failover-policy		string	Failover policy: "nextavail", "priority", "disabled", "unmapped".
home-node		string optional	Name of the interface's home node.
home-port		string optional	Name of the interface's home port.
is-home		boolean optional	True if the interface is currently on 'home-node' and 'home-port'.
name		obj-name	Name of the interface (e.g.: e0a).
network-address		string optional	Network address for the interface.
network-prefix-length	**[0..127]**	integer optional	Prefix length for the network mask of the interface.
operational-status		string	Operational status of the interface, valid values are "up", "down", "testing", "unknown".
owner-name		obj-name	Name of the owner of the interface.
owner-resource-key		resource-key	The resource key for the owner of the interface. The owner is determined from the network interface's role. If the role is "data" then the owner will be a vserver. If the role is "node_mgmt" then the owner will be a cluster node. Otherwise the owner will be a cluster.
owner-type		obj-type	Type of the interface's owner. Possible values are: • cluster • cluster_node • vserver
resource-key		resource-	The resource key for the interface.

		key	

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **data-protocol**

[top]

Data protocol configured on an interface. Possible values:

- "nfs" - Used for NFS connections,
- "cifs" - Used for CIFS connections,
- "iscsi" - Used for iSCSI connections,
- "fcp" - Used for Fibre Channel connections,
- "fcache" - Used for FlexCache connections,
- "none" - Used for management. Does not serve any file protocols.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **obj-type**

[top]

Type of a managed object. Possible values:

- "cluster"
- "cluster-node"
- "vserver"
- "resource-group"
- "volume"
- "qtree"
- "disk"
- "network-interface"
- "management-station"
- "quota-user"
- "initiator-group"
- "lun"
- "fcp-target"
- "aggregate"
- "port"
- "port-set"
- "lif"
- "ifgrp"
- "export-policy"
- "role"
- "storage-service"
- "service-workflow"

[none]

portset-iter

[top]

Iterate over portsets.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • resource group • cluster • vserver • portset • initiator group If resource-filter identifies a portset, that single portset will be returned. If resource-filter resolves to more than one portset, all of them will be returned. If no resource-filter is provided, all portsets will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		portset-info[] optional	The list of records.

Element definition: **portset-info**

[top]

Information about a portset.

Name	Range	Type	Description

portset-igroups		portset-igroup[]	List of initiator group bound to this portset.
portset-members		portset-member[]	List of portset members. The portset members can be tpgroups, FCP data LIFs.
portset-name		obj-name	Name of the portset.
portset-type		string	Portset type. The type of portset can be • iscsi • fcp • mixed
resource-key		resource-key	Resource Key for this portset.
vserver-name		obj-name	Name of vserver on which the portset resides. Always present in the output. The name is any simple name such as 'myhost'.
vserver-resource-key		resource-key	Identifier of vserver on which the portset resides. Always present in the output.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one

of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **portset-igroup**

[top]

List of initiator group bound to the portset.

Name	Range	Type	Description
igroup-name		string	Name of initiator group bound to this portset.
igroup-resource-key		resource-key	Identifier of initiator group.

Element definition: **portset-member**

[top]

Name of the portset member (tpgroups or FCP data LIF).

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

resource-lookup	APIs are provided for resolving managed resources to their corresponding resource keys. The resource keys can then be used as input to other management APIs.

resource-lookup

[top]

Lookup a managed resource by its fully qualified name. Returns the resource key for the managed resource.

Input Name	Range	Type	Description
qualified-name		string	The fully qualified name of the managed resource. The format of the qualified name for each resource type is defined as follows ({placeholders} that should be substituted with actual values): Data ONTAP resource types: • aggregate: {cluster-node-name}:{aggregate-name} • cifs_share: {vserver-name}:{cifs_share} • cluster: {cluster-name} • cluster_node: {cluster-node-name} • disk: {disk-name} • export_policy: {vserver-name}:{export-policy-name} • lun: {vserver-name}:/{volume-name}/{lun-name} or {vserver-name}:/{volume-name}/{qtree-name}/{lun-name} • network_lif: {owner-name}:{lif-name} where {owner-name} can be any of: {cluster-name} or {cluster-node-name} or {vserver-name} • network_port: {cluster-node-name}:{network-port-name} • qtree: {vserver-name}:/{volume-name}/{qtree-name} • snap_mirror: {source-vserver-name}:{source-volume-name} ->{dest-vserver-name}:{dest-volume-name} • snapshot: {vserver-name}:/{volume-name}:{snapshot-name} • storage_class: {vserver-name}:{storage-class-name} • storage_shelf: {cluster-node-name}:{channel-name}:{shelf-identifier} • volume: {vserver-name}:/{volume-name} • vserver: {vserver-name} OnCommand resource types: • resource_pool: {resource-pool-name} • service_workflow: {service-workflow-name} • storage_service: {storage-service-name} • storage_service_node: {storage-service-name}:{storage-service-node-name} • storage_service_connection: {storage-service-name}:{source-node-name} ->{storage-service-name}:{destination-node-name}
type		resource-type	Specifies the type of the managed resource being looked up. Not all resource types support lookup by name. See the documentation of qualified-name for a list of supported resource types and their name

			formats.

Output Name	Range	Type	Description
resource-key		resource-key	The resource key that represents the managed resource.

Errno	Description
EOBJECTNOTFOUND	
EOBJECTAMBIGUOUS	
EINVALIDOBJECTTYPE	
EINVALIDINPUT	

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **resource-type**

[top]

A canonical type name for a managed resource. Includes both Data ONTAP and OnCommand resource types. Valid Data ONTAP resource types:

- aggregate
- objectstore_config
- capacity_tier
- cifs_share
- cifs_share_acl
- cluster
- cluster_node
- disk
- export_policy
- export_rule
- fcp_lif
- fcp_port
- flash_device
- igroup
- iscsi_portal_group
- lun
- lun_mapping
- network_lif
- network_port
- ontap_job_schedule
- plex
- portset
- qtree
- raid_group
- routing_group
- service_processor
- sis_policy
- snap_mirror

- snapshot
- snapshot_policy
- snapshot_policy_schedule
- storage_class
- storage_shelf
- volume
- volume_move
- vserver
- vserver_name_mapping

Valid OnCommand resource types:

- management_station
- resource_pool
- service_workflow
- storage_service
- storage_service_connection
- storage_service_node

[none]

APIs in Category: **resource-pool**

API version 9.12

A resource pool is a collection of aggregates used for storage service provisioning operations.

resource-pool-aggregate-add

[top]

Add an aggregate to a resource pool. An aggregate may only belong to a single resource pool.

Input Name	Range	Type	Description
aggregate		resource-key	The resource key of the aggregate to be added.
resource-pool		resource-key	The resource key of the resource pool to which the aggregate is to be added.

Errno	Description
EOBJECTNOTFOUND	
E_AGGREGATE_ALREADY_IN_RESOURCE_POOL	
E_AGGREGATE_IN_PRECOMMIT_STATE	

resource-pool-aggregate-remove

[top]

Remove an aggregate from a resource pool.

Input Name	Range	Type	Description
aggregate		resource-key	The resource key of the aggregate to be removed.
resource-pool		resource-key	The resource key of the resource pool to which the aggregate belongs.

Errno	Description
EOBJECTNOTFOUND	
E_AGGREGATE_NOT_IN_RESOURCE_POOL	

resource-pool-create

[top]

Create a new resource pool.

Input Name	Range	Type	Description
description		string optional	Description of the new resource pool.
name		string	Name of the new resource pool.
snaplock-type		string optional	Snaplock-type of the new resource pool. The default value is non-snaplock. Possible values - "compliance", "enterprise" or "non-snaplock".

Output Name	Range	Type	Description
resource-key		resource-key	The resource key of the newly created resource pool.

Errno	Description
EOBJECTEXISTS	
EINVALIDINPUT	

resource-pool-destroy

[top]

Destroy an existing resource pool.

Input Name	Range	Type	Description
resource-pool		resource-key	The resource key of the resource pool to be destroyed.

Errno	Description
EOBJECTNOTFOUND	

resource-pool-iter

[top]

Iterate over resource pools.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • resource_pool If resource-filter identifies a resource pool, that single resource pool will be returned. If resource-filter resolves to more than one resource pool, all of them will be returned. If no resource-filter is provided, all resource pools will be listed.
			Specify the tag from the last call. It is not specified for the first call.

tag		string optional	For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		resource-pool-info[] optional	The list of records.

Errno	Description
EINVALIDTAG	
EINVALIDINPUT	

resource-pool-modify

[top]

Modify an existing resource pool.

Input Name	Range	Type	Description
description		string optional	Modified description of the resource pool. Specifying this input with a blank value will remove the existing description from the resource pool.
name		string optional	Modified name of the resource pool. Must not match the name of any existing resource pools.
resource-pool		resource-key	The resource key of the resource pool to be modified.
snaplock-type		string optional	Modified snaplock-type of the resource pool. The default value is non-snaplock. Possible values - "compliance", "enterprise" or "non-snaplock".

Errno	Description
EOBJECTNOTFOUND	
EINVALIDINPUT	
EOBJECTEXISTS	

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **resource-pool-info**

[top]

Information about a resource pool.

Name	Range	Type	Description
aggregates		resource-key[]	The collection of aggregates that belong to the resource pool.
capacity-available	**[0..2^63-1]**	integer	The combined available capacity of all the aggregates in the resource pool (in bytes).
capacity-total	**[0..2^63-1]**	integer	The combined total capacity of all the aggregates in the resource pool (in bytes).
capacity-used	**[0..2^63-1]**	integer	The combined used capacity of all the aggregates in the resource pool (in bytes).
description		string	The description of the resource pool.
name		string	The name of the resource pool.
resource-key		resource-key	The resource key for the resource pool.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

APIs in Category: **snapshot**

API version 9.12

Enables the user to display Snapshot copy locations and volume Snapshot metadata, and to set metadata values.

snapshot-expire

[top]

Deletes an existing volume snapshot. If the snapshot is currently active (e.g. involved in SnapMirror transfer, baseline snapshot of SnapVault etc.), then the snapshot will be marked as expired. Such expired snapshots will be deleted subsequently when no longer needed.

Input Name	Range	Type	Description
snapshot-resource-key		resource-key	Resource key of the volume snapshot to expire. If the snapshot specified is not found on the storage system, ESNAPSHOTNOTFOUND is returned.

Errno	Description
EOBJECTNOTFOUND	
EINVALIDINPUT	
ESNAPSHOTNOTFOUND	
EINVALIDOBJECTTYPE	

snapshot-get-location

[top]

Gets location of the replicas of the requested Snapshot copies.

Input Name	Range	Type	Description
snapshot-identifiers		snapshot-identifier[]	Information about the snapshot copies whose replica locations are being requested. Specified snapshot-identifier is first resolved to a version-uuid and then the snapshot copies for all the volumes currently in the input storage service and connected to the input volume, with the matching version-uuid are returned. You can specify the following combination of attributes in each snapshot-identifier: • version-uuid and volume-resource-key, or • snapshot-resource-key only. Any other combination of these attributes specified in the snapshot-identifier is invalid and EINVALIDINPUT error is returned. If an invalid version-uuid is specified in input snapshot-identifier, empty results will be returned for that identifier.
storage-service-		resource-	Resource key of the storage service node. Location of snapshots are

node-resource-key		key optional	returned for only those volumes which are currently in this storage service node.
storage-service-resource-key		resource-key	Resource key of the storage service. Location of snapshots are returned for only those volumes which are currently in this storage service. Matching snapshot locations on the relationships that are relinquished, or on the relationships in a different storage service, are not returned.

Output Name	Range	Type	Description
snapshot-location-results		snapshot-location-result-info[]	Snapshot locations.

Errno	Description
EINVALIDINPUT	
EOBJECTNOTFOUND	
ESNAPSHOTNOTFOUND	
E_INVALID_RESOURCE_KEY	
EINVALIDOBJECTTYPE	

snapshot-get-metadata

[top]

Returns metadata associated with a snapshot.

Input Name	Range	Type	Description
snapshot-resource-key		resource-key	Resource key of the volume snapshot associated with the metadata.

Output Name	Range	Type	Description
metadata		key-value-pair[]	Opaque metadata for this context. Metadata is usually set and interpreted by an external application. OnCommand Unified Manager does not look into the contents of the metadata.

Errno	Description
EOBJECTNOTFOUND	
EINVALIDINPUT	

snapshot-has-expired

[top]

Checks if the specified volume snapshot has expired.

Input Name	Range	Type	Description
snapshot-resource-key		resource-key	Resource key of the volume snapshot whose expiration status is being determined. If the snapshot specified is not found on the storage system, ESNAPSHOTNOTFOUND is returned.

Output Name	Range	Type	Description
expiration-status		boolean	Returns true if the specified snapshot is currently marked as expired, false otherwise.

Errno	Description
EOBJECTNOTFOUND	
EINVALIDINPUT	
ESNAPSHOTNOTFOUND	
EINVALIDOBJECTTYPE	

snapshot-set-metadata

[top]

Sets metadata for an existing volume snapshot. The metadata is automatically deleted when associated snapshot is removed.

Input Name	Range	Type	Description
metadata		key-value-pair[]	Opaque metadata for this snapshot. Metadata is usually set and interpreted by an external application. OnCommand Unified Manager does not look into the contents of metadata. Existing metadata will be removed if not included in the input. A maximum of 16 key value pairs are allowed per snapshot. If number of entries exceed this number, E_MAX_ENTRY_COUNT_EXCEEDED is returned. Keys in each pair can be 1 to 255 characters in length and are case-sensitive. Duplicate keys are not allowed. Values are opaque to the server and must not exceed 16384 (16k) characters in length. If any of these rules are violated, EINVALIDINPUT error will be returned.
snapshot-resource-key		resource-key	Resource key of the volume snapshot to be associated with the metadata.

Errno	Description
EOBJECTNOTFOUND	
EINVALIDINPUT	
E_MAX_ENTRY_COUNT_EXCEEDED	

Element definition: **key-value-pair**

[top]

The key/value for a generic object attribute.

Name	Range	Type	Description
key		string	key of the generic object attribute
value		string	Value of the generic object attribute

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **snapshot-identifier**

[top]

Information used to identify a Snapshot copy within the storage system. In some cases the snapshot-identifier element might match multiple Snapshot copies. A unique match in the result is guaranteed, if 'snapshot-resource-key' is provided in the input of snapshot-identifier.

Name	Range	Type	Description
snapshot-name		string optional	Name of the Snapshot copy. A volume-resource-key value is required when snapshot-name is specified.
snapshot-resource-key		resource-key optional	Resource key of the Snapshot copy. This is the immutable natural key to uniquely identify a snapshot on Data ONTAP.
snapshot-version-uuid		uuid optional	A unique identifier of this Snapshot copy and its logical data layout. If any two Snapshot copies exist that have the same version UUID, their contents must be logically equivalent. Snapshot copies within the same volume might have the same snapshot-version-uuid.
volume-resource-key		resource-key optional	Resource key of the original source volume where the Snapshot copy resides.

Element definition: **snapshot-location-result-info**

[top]

Results for the replica locations of the specified snapshot copy.

Name	Range	Type	Description
requested-snapshot-identifier		snapshot-identifier	Snapshot identifier as specified in the input.
snapshot-locations		snapshot-location-info[]	Information about where the replicas of the specified snapshot copy are located. List is empty if no replicas were found for the specified criteria.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **snapshot-identifier**

[top]

Information used to identify a Snapshot copy within the storage system. In some cases the snapshot-identifier element might match multiple Snapshot copies. A unique match in the result is guaranteed, if 'snapshot-resource-key' is provided in the input of snapshot-identifier.

Name	Range	Type	Description
snapshot-name		string optional	Name of the Snapshot copy. A volume-resource-key value is required when snapshot-name is specified.
snapshot-resource-key		resource-key optional	Resource key of the Snapshot copy. This is the immutable natural key to uniquely identify a snapshot on Data ONTAP.
snapshot-version-uuid		uuid optional	A unique identifier of this Snapshot copy and its logical data layout. If any two Snapshot copies exist that have the same version UUID, their contents must be logically equivalent. Snapshot copies within the same volume might have the same snapshot-version-uuid.
volume-resource-key		resource-key optional	Resource key of the original source volume where the Snapshot copy resides.

Element definition: **snapshot-location-info**

[top]

Information about where a replica of the specified snapshot copy is located.

Name	Range	Type	Description
is-dedupe-running		boolean	Returns true if the specified volume is currently running dedupe, false otherwise.
node-name		obj-name optional	Name of the storage service node that the volume is a member of.
node-resource-key		resource-key optional	Resource key of the storage service node that the volume is a member of.
snapshot-instance-uuid		uuid	This ID uniquely identifies a snapshot copy and its physical data layout. If any two snapshot copies in the world have the same instance UUID, they must be different instances of the exact same snapshot copy. Snapshot copies within the same volume must have different snapshot-instance-uuids.
snapshot-name		string	Name of the snapshot copy that was found.
snapshot-resource-key		resource-key	Resource key of the snapshot copy that was found.
snapshot-version-uuid		uuid	This ID identifies a snapshot copy and its logical data layout. If any two snapshot copies in the world have the same version UUID, their contents must be logically equivalent. Snapshot copies within the same volume may have the same snapshot-version-uuid.
storage-service-name		obj-name	Name of the storage service that the volume is a member of.

storage-service-resource-key		resource-key	Resource key of the storage service that the volume is a member of.
volume-name		obj-full-name	Name of the volume which holds the snapshot copy that was found.
volume-resource-key		resource-key	The resource key for the volume which holds the snapshot copy that was found.

Element definition: **uuid**

[top]

The 128-bit universally-unique identifier (UUID). UUIDs are formatted as 36-character strings. These strings are composed of 32 hexadecimal characters broken up into 5 groupings separated by '-'s. The first grouping consists of 8 hex characters, the second through fourth groupings consist of 4 hex characters each, and the fifth and final grouping consists of 12 hex characters. Note that a leading '0x' is not used. An example of an actual UUID is: 73a010ec-3d28-11df-84e8-123478563412.

[none]

Element definition: **obj-full-name**

[top]

Full name of an object. This typedef is an alias for the builtin ZAPI type **string**. An object full name conforms to all the rules of an obj-name, except that the full name may be up to 255 characters long.

Full names are created by concatenating an object name with any parent object names, so as to create a unique name for an object. The format of full names is as follows:

- **cluster** full names are the either the fully-qualified domain name or the IP address of the cluster.
- **cluster-node** full names are the either the fully-qualified domain name or the IP address of the cluster.
- **aggregate** full names are the cluster-node name and the aggregate name, separated by a colon, e.g. *cluster-node:aggr0*.
- **volume** full names are the vserver name and the volume name, separated by ":/", e.g. *vserver:/volume*. Note this does not include the "/vol" prefix. Volume and aggregate full names are distinguished by the presence of a forward slash after the colon.
- **qtree** full names are the containing volume full name and the qtree name, separated by a slash, e.g. *vserver:/volume/qtree*. The data not contained by any qtree may be represented by "-", e.g. *vserver:/volume/-*.
- **lun** full names are either a volume or qtree full name and the LUN path, separated by a slash, e.g. *vserver:/volume/LUN* or *vserver:/volume/qtree/LUN*.
- **initiator-group** full names are vserver name and the initiator group name, separated by a colon, e.g. *vserver:igroup*.
- **export-policy** full names are vserver name and the policy name, separated by a colon, e.g. *vserver:policy-name*.
- **lif** full names are a cluster, cluster-node, or vserver name and the interface name, separated by a colon, e.g. *cluster-name|cluster-node-name|vserver-name:lif*.
- **port-set** full names are the vserver name and the portset name, separated by a colon, e.g. *vserver:portset*.
- **fcp-target** full names are the cluster-node name and the target name, separated by a colon, e.g. *cluster-node:target*.

For any object not listed above, the obj-name and obj-full-name are identical.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

APIs in Category: **ssl**

API version 9.12

These apis allow for operations related to TLS/SSL communication, such as: generating the certificate used by the HTTPS server port, viewing the certificate, or creating certificate signing requests.

ssl-server-cert-generate

[top]

Generates the self-signed certificate to be used by the HTTPS port.

The newly generated certificate will become active after a reboot.

The certificate will be generated with the following parameters:

- Public Key Algorithm: RSA
- Key Length: 2048
- Signature Algorithm: SHA-256 with RSA Encryption
- Validity: 5 years

Output Name	Range	Type	Description
certificate		certificate	The generated certificate in Privacy Enhanced Mail (PEM) format.

ssl-server-cert-get

[top]

Retrieves the certificate chain used by the HTTPS port. The certificate chain can be retrieved by anybody.

Output Name	Range	Type	Description
certificate-chain		certificate-chain	The certificate chain in Privacy Enhanced Mail (PEM) format.

ssl-server-cert-request

[top]

Creates a PKCS#10 Certification Request from the HTTPS port's certificate that can be submitted to a Certificate Authority for signing. The signed certificate received from a CA can then be imported via ssl-server-cert-set.

Output Name	Range	Type	Description
certificate-request		string	The generated PKCS#10 Certification Request in Privacy Enhanced Mail (PEM) format.

ssl-server-cert-set

[top]

Sets the certificate to be used by the HTTPS port. The certificate must be a CA-signed certificate generated from the Certification Request created via ssl-server-cert-request.

The newly set certificate will become active after a reboot.

Input Name	Range	Type	Description
certificate-chain		certificate-chain	The certificate chain in Privacy Enhanced Mail (PEM) format. Each certificate in the chain from the new certificate to the Certificate Authority must be included.

Errno	Description
EINVALIDINPUT	

Element definition: **certificate**

[top]

An X.509 certificate in Privacy Enhanced Mail (PEM) format.

[none]

Element definition: **certificate-chain**

[top]

A chain of X.509 certificates in Privacy Enhanced Mail (PEM) format.

[none]

APIs in Category: **storage-service**

API version 9.12

Storage service is a user-created entity that defines the properties for data protection in terms of a service topology. Storage service specifies the replication topology and how to provision storage for each non-primary node in the topology. Volumes can be subscribed to the storage service in order to create one or more SnapMirror relationships.

The storage service and its APIs only provide data protection capability.

A cascaded vault connection of a storage service can have only one preceding mirror connection.

A Storage Service maintains a list of its member objects. The service tracks the number of unique subscription subscription-contexts that were used to subscribe/unsubscribe a member object. The subscription-context tag is an arbitrary string defined by the caller. The subscription-context may be used as a handle in other APIs referring to the group of objects marked with the same subscription-context. There can be only one subscription of this type. For example, volumeA can be subscribed to a storage service S1 via subscription-context foo or subscription-context bar. In this case, the volume will have 2 subscriptions. If you try to subscribe the same volume again using any one of those two ways, an error is returned. In order to unsubscribe the volume fully, you must unsubscribe the volume using both subscriptions (in other words, storage-service-unsubscribe API must be called once with subscription-context foo and once with subscription-context bar). When all subscriptions have been removed (through unsubscribe), the storage service will not perform any more operations on behalf of that member and it will be marked as 'disabled'. Artifacts (secondary volumes and relationships) created from the subscription are preserved until an explicit API call is made to clean them up.

storage-service-cleanup

[top]

Users can invoke this API when they feel no more need to maintain the member artifacts for any kind of restore purposes. Specified members must not have any active subscriptions. For every specified member, all the relevant data protection relationships and relevant provisioned storage objects are removed from the storage service. Users may also opt to destroy these artifacts from the storage system.

Input Name	Range	Type	Description
keep-storage-artifacts		boolean optional	If true, protection artifacts are removed only from storage service but not destroyed from storage system. If false, protection artifacts are removed from the storage service and destroyed from storage system as well. Default is true.
			Resource keys of the members to clean up. Members must belong to the root node of the storage service, otherwise

members		resource-key[] optional	E_MEMBER_NOT_IN_REQUIRED_NODE error is returned. If specified and 'storage-service-nodes' input is not provided, members should not have any active subscriptions with the storage service, otherwise an E_OBJECT_CURRENTLY_SUBSCRIBED error is returned. If an empty list is provided or this input is not provided, and 'storage-service-nodes' input is absent, then all the unsubscribed members will be cleaned up from the storage service.
storage-service-nodes		resource-key[] optional	Resource keys of the nodes of the storage service, from which the protection artifacts should be removed. If this input is provided, protection artifacts are cleaned up regardless of whether the member has subscription or not. Each node in this list must be a non-root node of the storage service, otherwise an EINVALIDINPUT error will be returned. If provided and 'members' input is not provided, protection artifacts of all the members are removed only from the specified nodes and all their downstream nodes of the storage service. If provided, and 'members' input is also provided, protection artifacts of the input 'members' are removed only from the specified nodes and all their downstream nodes of the storage service.
storage-service-resource-key		resource-key	Resource key of the storage service.

Output Name	Range	Type	Description
job-id		job-id	Object-id of the job created by this API call. The job's task list is a manifest that expresses the set of different tasks and the sequence in which they will be executed in order to accomplish a high level workflow.

Errno	Description
EOBJECTNOTFOUND	
EINVALIDINPUT	
E_OBJECT_CURRENTLY_SUBSCRIBED	
E_MEMBER_NOT_IN_REQUIRED_NODE	
E_STORAGE_SERVICE_LOCKED	

storage-service-conform

[top]

Checks configuration on members of a storage service. This API runs an on-demand check on one or more members of the storage service to bring the storage system configuration into conformance with the strategy defined by the storage service. A configuration check determines the volumes' current conformance status and the set of "actions" needed to make them conformant. Then this API starts the actions to bring the storage system configuration into conformance. Successful completion of this API indicates that "corrective" conformance actions are started though not necessarily completed.

Input Name	Range	Type	Description
members		resource-key[] optional	Resource keys of the member-objects on which to check conformance. Members must belong to the root node of the storage service. Members should have at least one active subscription with the storage service, otherwise E_OBJECT_NOT_SUBSCRIBED error is returned. If an empty list is provided or this input is not provided, conformance check will run on all the subscribed members of the storage service. If a subscription-context is specified as input then 'members' input must not be provided or must be empty.

storage-service-resource-key		resource-key	Resource key of the storage service.
subscription-context		string optional	An optional client-defined string used to tag client-associated members when subscribing them to a storage service. Members that are subscribed to a storage service via subscription-context can be specified with the subscription-context element or individually by their members element when APIs are called that require identification of the storage service members they affect. An API call cannot have both the subscription-context element and the members element as inputs. If subscription-context is specified for a storage-service-conform API call, only the members subscribed via that specific context are checked. This value can be a maximum of 255 characters.

Output Name	Range	Type	Description
job-id		job-id	Object-id of the job created by this API call. The job's task list is a manifest that expresses the set of different tasks and the sequence in which they are executed in order to accomplish a high level workflow.

Errno	Description
EOBJECTNOTFOUND	
E_OBJECT_NOT_SUBSCRIBED	
E_CONTEXT_NOT_FOUND	
EINVALIDINPUT	
E_STORAGE_SERVICE_MARKED_FOR_DELETION	
E_STORAGE_SERVICE_LOCKED	

storage-service-create

[top]

Creates a new storage service by specifying protection and provisioning strategy.

Input Name	Range	Type	Description
storage-service-client-tag		string	Arbitrary string describing the name of the client program that is creating the storage service. This value can be a maximum of 32 characters.
storage-service-contact-list		email-address[] optional	List of contact email addresses, each of which can be no longer than 255 characters.
storage-service-description		string optional	Description of the new storage service. This value can be a maximum of 255 characters.
storage-service-name		obj-name	Name of the new storage service. This value can be a maximum of 255 characters.
storage-service-owner		string optional	Name of the owner of the storage service. This value can be a maximum of 255 characters.
storage-service-topology-info		storage-service-topology-info	Composite entity representing the nodes and connections of a protection and provisioning strategy.

Output Name	Range	Type	Description
storage-			

service-resource-key		resource-key	Resource key of the new storage service.

Errno	Description
EINVALIDTOPOLOGY	
ETOPOLOGYNOTSUPPORTED	
EOBJECTNOTFOUND	
EINVALIDINPUT	

storage-service-destroy

[top]

Destroys a storage service. This API also destroys all protection artifacts, such as volumes and relationships, that are managed by this storage service. Managed protection artifacts include those that are either created by or imported into the storage service.

Input Name	Range	Type	Description
storage-service-resource-key		resource-key	Resource key of the storage service to destroy.

Output Name	Range	Type	Description
job-id		job-id	Object-id of the job created by this API call. The job's task list is a manifest that expresses the set of different tasks and the sequence in which they are executed in order to accomplish a high level workflow.

Errno	Description
EOBJECTNOTFOUND	
EINVALIDINPUT	
E_STORAGE_SERVICE_LOCKED	

storage-service-import

[top]

Imports non-managed storage objects into an existing storage service. When you use the storage-service-import API, you must adhere to the following rules. If a rule is violated, the error code EINVALIDINPUT is returned, unless otherwise specified.

- Each relationship must satisfy one of the following three conditions:
  - The source volume is being imported into the root node.
  - The source volume already exists in the connection's source node.
  - The preceding relationship is also being imported.
- A relationship can only be imported into one storage service. If the same relationship exists in this or any storage service, the error code E_RELATIONSHIP_ALREADY_MANAGED is returned.
- If the subscription for the specified subscription context does not exist, a new subscription is created by the API.
- Source and destination volumes of the relationship can be associated with only a single node of the same storage service.
- Source or destination volume being imported into a non-root node of the storage service cannot be in the non-root node of any other storage service.
- If multiple relationships have the same source volume, each of these relationships must be

imported into a different connection.
- Clients will need to call storage-service-conform to provision destination volumes and create relationships according to the topology of the storage service for the relationships imported in here.

Input Name	Range	Type	Description
import-info		storage-service-import-info[]	SnapMirror relationship or destination volume (one must be provided), and storage service connection pair for import.
storage-service-resource-key		resource-key	Resource key of the storage service.
subscription-context		string optional	The subscription context to use when importing root node member(s). The same rules and validations apply as described in the storage-service-subscribe API. When importing a relationship that contains a primary volume, the subscription context must be specified; if it is not, the E_CONTEXT_MISSING error code is returned. When importing a relationship that contains only non-primary volumes, the subscription context is ignored. This value can be a maximum of 255 characters. There are number of use cases for storage service import: • use case 1 - the primary volume of the relationship is currently not a member of the root node (i.e. import a relationship to one of the primary hops). The subscription context must be specified. • use case 2 - the primary volume of the relationship is already a member of the root node (i.e. import a relationship to the second leg of a fanout storage service). The subscription context is optional. • use case 3 - import a relationship from a secondary to a tertiary to a cascading storage service. The subscription context input will be ignored for that particular imported member.

Output Name	Range	Type	Description
job-id		job-id	ID of the job created by this API call. The job's task list is a manifest that expresses the set of different tasks and the sequence in which they are executed in order to accomplish a high level workflow.

Errno	Description
EOBJECTNOTFOUND	
EINVALIDINPUT	
E_RELATIONSHIP_ALREADY_MANAGED	
E_CONTEXT_MISSING	
E_STORAGE_SERVICE_LOCKED	

storage-service-iter

[top]

Starts iteration to list storage services.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
storage-service-client-tag		string optional	Arbitrary client provided description of the program utilizing the storage service. If storage-service-client-tag is specified as input, then storage-service-resource-key input must not be provided.
storage-service-resource-key		resource-key optional	Resource key of a storage service. If specified, only the specified storage service is returned. If storage-service-resource-key is specified as input, then storage-service-client-tag input must not be provided.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		storage-service-info[] optional	The list of records.

storage-service-member-iter

[top]

Starts iteration to list members of the storage service.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
node-resource-key		resource-key optional	Resource key of the node on which the members to be listed reside. If no node is specified, members on all nodes are listed.
storage-service-resource-key		resource-key	Resource key of the storage service for which members are listed.
subscription-context		string optional	An optional client-defined string used to tag client-associated members when subscribing them to a storage service. The maximum length of a subscription-context is 255 characters.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
		string	Tag for the next call. Not present when there are no more objects to

next-tag		optional	return.
num-records		integer	The number of records returned in this call.
records		storage-service-member-info[] optional	The list of records.

Errno	Description
E_CONTEXT_NOT_FOUND	

storage-service-modify

[top]

Modify the attributes of an existing storage service. Clients will need to call storage-service-conform to provision destination volumes and create relationships to align the topology of the storage service and the modifications made in this API.

Input Name	Range	Type	Description
storage-service-client-tag		string optional	Arbitrary string describing the name of the client program that is creating the storage service. This value can be a maximum of 32 characters.
storage-service-contact-list		email-address[] optional	List of contact email addresses, each of which can be no longer than 255 characters.
storage-service-description		string optional	Description of the storage service, up to 255 characters long.
storage-service-name		obj-name optional	Modified name of the storage service object. This value can be a maximum of 255 characters.
storage-service-owner		string optional	Name of the owner of the storage service, up to 255 characters long.
storage-service-resource-key		resource-key	Resource key of a storage service object.
storage-service-topology-info		storage-service-topology-info optional	Topology to be modified. If any of the changes to the topology will cause the storage service to become non-conformant this will be mentioned in the job results. To change an existing element of a topology, you should specify all elements of the storage-service-topology-info. Any optional elements that are not specified will be set to their default value. For example, to modify the max-transfer-rate element of storage-service- topology-connection-info from 1024 KB/sec to 2048 KB/sec you should specify values for other storage-service-topology-connection-info elements such as lag-error-threshold and lag-warning-threshold otherwise default values for these elements are used. Similarly, omitting a node or connection from storage-service-topology-info will cause it to be removed from the storage service. A node omitted from storage-service-topology-info must not contain any protection artifacts, else an EINVALIDINPUT error will be returned. 'storage-service-cleanup' API should be used to remove the protection artifacts from the node, before deleting it. A storage service can only have maximum of one mirror relationship upstream from

			vault relationship in a mirror-vault cascade topology.

Output Name	Range	Type	Description
job-id		job-id	Refers to the job created for this call to storage-service-modify.

Errno	Description
EINVALIDTOPOLOGY	
ETOPOLOGYNOTSUPPORTED	
EOBJECTNOTFOUND	
EINVALIDINPUT	
E_STORAGE_SERVICE_LOCKED	

storage-service-protection-update-start

[top]

Updates data protection relationships for one or more members of the storage service. Storage service has a built-in topology relationship defined between the nodes. When the storage-service-conform API is called, protection relationships are created but not initialized. The first call to the storage-service-protection-update-start API triggers initialization (baseline data transfer). Subsequent calls trigger protection relationship updates.

Input Name	Range	Type	Description
destination-node-resource-key		resource-key optional	Resource key of destination node of the storage service. When destination-node-resource-key specified, members will be updated until the destination node; if not specified, members will be updated to the edge node(s) of the storage service topology. If provided destination node doesn't belong to this storage service, EINVALIDINPUT is returned. If source and destination nodes are specified, there must be a continuous downstream path between the two nodes, otherwise, EINVALIDINPUT is returned.
member-snapshots		resource-key[] optional	Resource keys of snapshot copies of the member whose relationships are to be updated. Snapshot copy existence on destination is verified after transfer completion. Snapshot copies must be of a member storage object which belongs to the root node or source node of the storage service if source node is specified. That member must have at least one active subscription. Otherwise, E_OBJECT_NOT_SUBSCRIBED error is returned. If an empty list is provided or this input is not provided, relationships for all the subscribed members are updated and snapshot copy existence is not verified. If specified, 'members' and 'subscription-context' input must not be provided, otherwise, EINVALIDINPUT is returned.
members		resource-key[] optional	Resource keys of the storage objects whose relationships are to be updated. Members must belong to the root node of the storage service. Members should have at least one active subscription with the storage service, otherwise E_OBJECT_NOT_SUBSCRIBED error is returned. Only relationships for storage objects specified in the 'members' list are updated. If specified, 'member-snapshots' and 'subscription-context' must not be provided, otherwise, EINVALIDINPUT will be returned. If an empty list is provided or this input is not provided, relationships for all the subscribed members are updated. When this input is used, Data ONTAP snapmirror-update will be invoked and will not verify snapshot existence. If storage service contains vault relationships, snapshot snapmirror-label attribute must be set to be used by the vaulting system to identify a vaulting scheme.

source-node-resource-key		resource-key optional	Resource key of source node of storage service. When source-node-resource-key specified, members will be updated starting at the source node; if not specified, members will be updated starting at root node of the storage service. If provided source node doesn't belong to this storage service, EINVALIDINPUT is returned.
storage-service-resource-key		resource-key	Resource key of the storage service.
subscription-context		string optional	An optional client-defined string used to tag client-associated members when subscribing them to a storage service. Members that are subscribed to a storage service via subscription-context can be specified with the subscription-context element or individually by their members element when APIs are called that require identification of the storage service members they affect. An API call cannot have both the subscription-context element and the members element as inputs. If specified, only the relationships for members subscribed via this context are updated. If omitted, then relationships for all the members of the root node are updated. If either 'members' or 'member-snapshots' input are specified, 'subscription-context' input must not be provided or must be empty, otherwise, EINVALIDINPUT is returned. This value can be a maximum of 255 characters. When this input is used, Data ONTAP snap-mirror-update will be invoked for members of this context and will not verify snapshot existence. If storage service contains vault relationships, snapshot snapmirror-label attribute must be set to be used by the vaulting system to identify a vaulting scheme.

Output Name	Range	Type	Description
job-id		job-id	Object-id of the job created by this API call. The job's task list is a manifest that expresses the set of different tasks and the sequence in which they are executed in order to accomplish a high level workflow.

Errno	Description
EOBJECTNOTFOUND	
E_CONTEXT_NOT_FOUND	
EINVALIDINPUT	
E_OBJECT_NOT_SUBSCRIBED	
E_STORAGE_SERVICE_MARKED_FOR_DELETION	
E_STORAGE_SERVICE_LOCKED	

storage-service-subscribe

[top]

Subscribes objects to a storage service. Currently, volumes are subscribable objects. Clients will need to call storage-service-conform to provision destination volumes and create relationships for the objects subscribed in here.

Input Name	Range	Type	Description
storage-service-resource-key		resource-key	Resource key of the storage service.
			An optional client-defined string used to tag client-associated members when subscribing them to a storage service. Members that are subscribed to a storage service via subscription-context can be

subscription-context		string	specified with the subscription-context element or individually by their members element when APIs are called that require identification of the storage service members they affect. An API call cannot have both the subscription-context element and the members element as inputs. The subscription context is provided as a convenience to the caller. The context may be used as a handle referring to all objects subscribed under the same context. The same object can be subscribed multiple times under different contexts. An object is fully unsubscribed from the storage service only after it is unsubscribed via all the subscription contexts. This string must be unique within the storage service so that no two clients confuse their subscriptions. The API creates a subscription for each new context, then adds storage objects to that subscription as subscription members. If all subscription members are removed, the subscription object becomes unnecessary, and the subscription is deleted at that point. The maximum length is 255 characters.
subscription-members		resource-key[] optional	Resource keys of the object(s) subscribing to the storage service. If an empty list is provided or this input is not provided, then an empty subscription object is created. Subscription members are always added to that storage service's primary node. Only a read-writable volume can be subscribed to a storage service with mirror-vault cascade topology. A volume can be subscribed to only one storage service with mirror-vault cascade topology. If an object is already subscribed with the context provided in subscription-context, an E_OBJECT_ALREADY_SUBSCRIBED is returned.

Output Name	Range	Type	Description
job-id		job-id	Object-id of the job created by this API call. The job's task list is a manifest that expresses the set of different tasks and the sequence in which they are executed in order to accomplish a high level workflow.

Errno	Description
EOBJECTNOTFOUND	
E_OBJECT_ALREADY_SUBSCRIBED	
EINVALIDINPUT	
E_STORAGE_SERVICE_MARKED_FOR_DELETION	
E_STORAGE_SERVICE_LOCKED	
E_OBJECT_IN_PRECOMMIT_STATE	

storage-service-subscription-context-get-metadata

[top]

Returns metadata of a subscription context set by storage-service-subscription-context-set-metadata

API.

Input Name	Range	Type	Description
storage-service-resource-key		resource-key	Resource key of the storage service associated with the specified context.
subscription-context		string	The client-defined context string used to tag client-associated members when subscribing them to a storage service. If the specified subscription-context does not exist, E_CONTEXT_NOT_FOUND is returned. This value can be a maximum of 255 characters

Output Name	Range	Type	Description
metadata		keyvalue[]	Opaque metadata for this context. Metadata is usually set and interpreted by an external application.

Errno	Description
EOBJECTNOTFOUND	
E_CONTEXT_NOT_FOUND	
EINVALIDINPUT	

storage-service-subscription-context-set-metadata

[top]

Sets metadata for an existing subscription context for a given storage service. The metadata is automatically deleted if the subscription context is removed for the storage service.

Input Name	Range	Type	Description
metadata		keyvalue[]	Opaque metadata for this context. Metadata is usually set and interpreted by an external application. A maximum of 16 key value pairs are allowed per subscription context. If number of entries exceed this number with this request, E_MAX_ENTRY_COUNT_EXCEEDED is returned.
storage-service-resource-key		resource-key	Resource key of the storage service associated with the specified context. If the storage service is being deleted, E_STORAGE_SERVICE_MARKED_FOR_DELETION is returned.
subscription-context		string	The client-defined context string used to tag client-associated members when subscribing them to a storage service. If the specified subscription-context does not exist, E_CONTEXT_NOT_FOUND is returned. This value can be a maximum of 255 characters.

Errno	Description
EOBJECTNOTFOUND	
E_CONTEXT_NOT_FOUND	
E_STORAGE_SERVICE_MARKED_FOR_DELETION	
E_MAX_ENTRY_COUNT_EXCEEDED	
EINVALIDINPUT	

storage-service-subscription-iter

[top]

Iterate over subscriptions of a storage service. The user must specify as input: the storage-service-resource-key element alone, the subscription-resource-key element alone, or a combination of (storage-service-resource-key, subscription-context) elements. The user cannot leave all three inputs unspecified.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
storage-service-resource-key		resource-key optional	Resource key of the storage service whose subscriptions are listed.
subscription-context		string optional	An optional client-defined string used to tag client-associated members when subscribing them to a storage service. The maximum length of a subscription-context is 255 characters.
subscription-resource-key		resource-key optional	Resource key of the subscription returned. Subscription resource keys are unique, so the storage-service-resource-key input does not need to be specified. If specified, only subscription information for this specific subscription-resource-key is returned. If subscription-resource-key and (storage-service-resource-key, subscription-context) are specified, an error is returned.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		storage-service-subscription-info[] optional	The list of records.

Errno	Description
E_CONTEXT_NOT_FOUND	

storage-service-unsubscribe

[top]

Unsubscribes members from an existing storage service. If member is unsubscribed, it remains in the storage service but is marked disabled. When a member is marked disabled, all of its protection artifacts, such as relationships and storage objects, are preserved; however, it does not actively participate in future replication updates or conformance checks.

NOTE: To relinquish a member, or relinquish or destroy its related protection artifact members you can use the storage-service-cleanup API.

Input Name	Range	Type	Description

storage-service-resource-key		resource-key	Resource key of the storage service.
subscription-context		string	A client-defined string used to tag client-associated members when subscribing them to a storage service. Members that are subscribed to a storage service via subscription-context can be specified with the subscription-context element or individually by their members element when APIs are called that require identification of the storage service members they affect. The given 'subscription-members' subscribed to the storage service with this context are unsubscribed. If 'subscription-members' input is not specified, all objects subscribed to the storage service with this context are unsubscribed. If all subscription members are removed for a given context, the subscription-context is automatically deleted. This value can be a maximum of 255 characters.
subscription-members		resource-key[] optional	Resource keys of the objects to unsubscribe from the storage service. Members must belong to the root node of the storage service. Members should be subscribed to the storage service via input 'subscription-context' tag, otherwise E_OBJECT_NOT_SUBSCRIBED error is returned. If an empty list is provided or this input is not provided, all the subscribed members under the specified context tag are unsubscribed.

Output Name	Range	Type	Description
job-id		job-id	Identifier of the job created by this API call. The job's task list is a manifest that expresses the set of different tasks and the sequence in which they are executed in order to accomplish a high level workflow.

Errno	Description
EOBJECTNOTFOUND	
E_OBJECT_NOT_SUBSCRIBED	
E_CONTEXT_NOT_FOUND	
E_STORAGE_SERVICE_MARKED_FOR_DELETION	
EINVALIDINPUT	
E_STORAGE_SERVICE_LOCKED	

storage-service-workflow-list-info

[top]

Lists all the available workflows that storage service supports. The workflows returned in output are uniquely identifiable across workflow listings in OnCommand product suite.

Output Name	Range	Type	Description
storage-service-workflows		storage-service-workflow-info[]	List of storage service workflows that can be assigned to the non-root node of a storage service.

Element definition: **email-address**

[top]

A single email address. It cannot contain spaces, semi-colons or unprintable characters.

[none]

Element definition: **job-id**

[top]

Opaque identifier for a job.

[none]

Element definition: **keyvalue**

[top]

Name-value field pair in the metadata.

Name	Range	Type	Description
field-name		string	Name of the metadata field. Field names can be 1 to 255 characters in length and are case-insensitive.
field-value		string	Arbitrary, user-defined data expressed as a string. The string is opaque to the server and must not exceed 16384 (16k) characters in length.

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **resource-key** [top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **storage-service-import-info** [top]

Information about members to be imported into the storage service. Members are implied through the relationship source and destination objects.

Name	Range	Type	Description
connection-resource-key		resource-key	Resource key of the storage service connection which the relationship will be associated with.
destination-volume-resource-key		resource-key optional	Resource key of the relationship's destination volume to be imported. The source and destination volumes of the relationship will be imported into their respective storage service nodes. If specified, relationship-resource-key should not be specified.
relationship-resource-key		resource-key optional	Resource key of the relationship to be imported. The source and destination volumes of the relationship will be imported into their respective storage service nodes. If specified, destination-volume-resource-key should not be specified.

Element definition: **storage-service-info** [top]

Name	Range	Type	Description
is-storage-service-marked-for-deletion		boolean	Whether the storage service is marked for deletion. This flag is set for storage services for which attempted removal was not completely succesful. Periodic configuration check or storage-service-destroy may complete the removal later.
storage-service-client-tag		string	Arbitrary client provided description of the program utilizing the storage service. This value can be a maximum of 32 characters.
storage-service-contact-list		email-address[] optional	List of contact email addresses, each of which can be no longer than 255 characters.
storage-		string	

service-description		optional	Description of the storage service.
storage-service-name		obj-name	Name of the storage service. This value can be a maximum of 255 characters.
storage-service-owner		string optional	Name of the owner of the storage service. This value can be a maximum of 64 characters.
storage-service-resource-key		resource-key	Resource key of the storage service.
storage-service-topology-info		storage-service-topology-info	Composite entity representing the nodes and connections of a protection and provisioning strategy.

Element definition: **storage-service-member-info**

[top]

Information about one member of a storage service.

Name	Range	Type	Description
incoming-connection-resource-key		resource-key optional	Resource key of the connection between member's source-node and the node to which the member belongs. Returned only for non-root members.
incoming-relationship-resource-key		resource-key optional	Resource key of the relationship between source-member and this member. Returned only for non-root members when relationship exists.
is-enabled		boolean	Whether this member is enabled. If the member is disabled, it does not actively participate in the replication updates and conformance checks. All storage objects and relationships are preserved. However, relationships are not updated and no new relationships are created. Member is marked as disabled only if all its subscriptions have been removed.
is-member-missing		boolean optional	If the value is true, it indicates that the storage service member volume no longer exists on the storage system or the member volume may exist but not associated with this node member.
member-name		obj-full-name	Name of the member.This string can be a maximum of 255 characters.
member-resource-key		resource-key	Resource key of the member.
member-type		string	Type of the member.Possible values are:'volume'.
node-name		string	Name of the node to which the member belongs. This value can be a maximum of 255 characters.
node-resource-key		resource-key	Resource key of the node to which the member belongs.
source-member-name		obj-full-name optional	Name of the source member. Returned only for non-root members. This value can be a maximum of 255 characters.
source-member-		resource-key	Resrouce key of the source member. For example, if this member is the destination volume for a mirror relationship, then source member

resource-key		optional	is the corresponding source volume. Returned only for non-root members.
source-node-name		string optional	Name of the node to which the source member belongs. Returned only for non-root members. This value can be a maximum of 255 characters.
source-node-resource-key		resource-key optional	Resource key of the node to which the source member belongs. Returned only for non-root members.
subscriptions		resource-key[] optional	Subscription resource keys of the context that was used when subscribing. Present only for the members of the root node. When member is fully unsubscribed via all contexts, this list is empty. In such case, no actions such as updating the SnapMirror relationship or provisioning secondary storage are taken for that member. However, existing storage and relationships are left intact. These leftover artifacts can be relinquished or destroyed using storage-service-cleanup. Subscription information can be obtained by calling storage-service-subscription-iter API.

Element definition: **storage-service-subscription-info**

[top]

Information about one storage service subscription.

Name	Range	Type	Description
storage-service-name		obj-name	Name of the storage service.
storage-service-resource-key		resource-key	Resource key of the storage service.
subscription-context		string	Subscription context used to subscribe the members. This value can be a maximum of 255 characters.
subscription-members		storage-service-subscription-member-info[]	List of member objects subscribed with this subscription. Subscription members are always on the root node of the storage service.
subscription-resource-key		resource-key	Resource key of the subscription object.

Element definition: **storage-service-topology-info**

[top]

Information about the topology of a storage service which composite entity representing the nodes and connections of a protection and provisioning strategy.

Name	Range	Type	Description
connections		storage-service-topology-connection-info[]	Set of connections. Connection is a component of a topology. A connection has configurable properties of the relationship between two nodes. This field will not be set for single node topologies. But for multi-node topologies, it must be set.

		optional	
nodes		storage-service-topology-node-info[]	Set of nodes. Node is a component of a topology. A node has configurable properties of storage service primary sources as well as mirror and vault destinations.

Element definition: **storage-service-workflow-info**

[top]

Name	Range	Type	Description
storage-service-workflow-description		string	Description of the storage service workflow.
storage-service-workflow-name		obj-name	Name of the storage service workflow. This value can be a maximum of 255 characters.
storage-service-workflow-resource-key		resource-key	Resource key of the storage service workflow.
storage-service-workflow-type		string	Type of storage service workflow. Possible values are "mirror" and "vault".

Element definition: **email-address**

[top]

A single email address. It cannot contain spaces, semi-colons or unprintable characters.

[none]

Element definition: **obj-full-name**

[top]

Full name of an object. This typedef is an alias for the builtin ZAPI type **string**. An object full name conforms to all the rules of an obj-name, except that the full name may be up to 255 characters long.

Full names are created by concatenating an object name with any parent object names, so as to create a unique name for an object. The format of full names is as follows:

- **cluster** full names are the either the fully-qualified domain name or the IP address of the cluster.
- **cluster-node** full names are the either the fully-qualified domain name or the IP address of the cluster.
- **aggregate** full names are the cluster-node name and the aggregate name, separated by a colon, e.g. *cluster-node:aggr0*.
- **volume** full names are the vserver name and the volume name, separated by ":/", e.g. *vserver:/volume*. Note this does not include the "/vol" prefix. Volume and aggregate full names are distinguished by the presence of a forward slash after the colon.
- **qtree** full names are the containing volume full name and the qtree name, separated by a slash, e.g. *vserver:/volume/qtree*. The data not contained by any qtree may be represented by "-", e.g. *vserver:/volume/-*.
- **lun** full names are either a volume or qtree full name and the LUN path, separated by a slash, e.g. *vserver:/volume/LUN* or *vserver:/volume/qtree/LUN*.
- **initiator-group** full names are vserver name and the initiator group name, separated by a colon, e.g. *vserver:igroup*.

- **export-policy** full names are vserver name and the policy name, separated by a colon, e.g. *vserver:policy-name*.
- **lif** full names are a cluster, cluster-node, or vserver name and the interface name, separated by a colon, e.g. *cluster-name|cluster-node-name|vserver-name:lif*.
- **port-set** full names are the vserver name and the portset name, separated by a colon, e.g. *vserver:portset*.
- **fcp-target** full names are the cluster-node name and the target name, separated by a colon, e.g. *cluster-node:target*.

For any object not listed above, the obj-name and obj-full-name are identical.

[none]

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **storage-service-subscription-member-info**

[top]

Information about one member of a storage service subscription.

Name	Range	Type	Description
is-member-deleted		boolean optional	If the value is true, it indicates that the storage service member volume no longer exists on the storage system.
member-name		obj-full-name	Name of the member. This value can be a maximum of 255 characters.
member-resource-key		resource-key	Resource key of the member.
member-type		string	Type of the member. Possible values are: 'volume'.

Element definition: **storage-service-topology-connection-info**

[top]

Contains the configurable properties of the relationship between two nodes.

Name	Range	Type	Description
connection-resource-key		resource-key optional	Resource key of the Connection. Always present in the output. Auto-assigned for the new connections. Must be included when referring to existing connections, for example while modifying existing connection attributes.
connection-type		string	Data ONTAP relationship between two nodes. Possible values are: 'mirror', 'vault', 'version_flexible_mirror'.
destination-node-name		string	Destination node name. The name of the connection's destination node end point. Must be present for newly created nodes. For modification, if node name changes, the new node name must be used. This value can be a maximum of 255 characters.
destination-node-resource-key		resource-key optional	Resource key of the destination node. Must not be provided if the node is new. For modification of existing topologies, this value must be provided.
is-lag-error-enabled		boolean optional	Indicates whether the system should generate an error event when the last update of relationship is older than lag-error-threshold value. The default value is true.
is-lag-warning-enabled		boolean optional	Indicates whether the system should generate a warning event when the last update of relationship is older than the lag-warning-threshold value. The default value is true.
is-network-compression-enabled		boolean optional	Indicates whether the network compression should be enabled for transferring snapshots across snapmirror relationships, if set to true it will be enabled. The default value is false. NOTE: Inorder to take the effect of this parameter, both source and destination volumes of snapmirror relationship should be running Data ONTAP 8.3 or later otherwise this parameter has no effect.
lag-error-threshold	**[1..2^31 - 1].**	integer optional	Lag, in seconds. If the last update of relationship is older than this value, and if the element is-lag-error-enabled is set, the system generates an error event for that relationship. The default value is 172800 seconds (which is 2 days).
lag-warning-	**[1..2^31**	integer	Lag, in seconds. If the last update of relationship is older than this value, and if the element is-lag-warning-enabled is set, the system

threshold	**- 1].**	optional	generates a warning event for that relationship. The default value is 129600 seconds (which is 1.5 days).
max-transfer-rate		integer optional	Specifies the upper bound for each relationship, in kilobytes per second, at which data is transferred between ONTAP clusters. The default is NULL, which indicates that the value set on ONTAP will be used. If a value is specified, then that value is used as the data transfer rate. The max-transfer-rate option does not affect relationships confined to a single cluster.
source-node-name		string	Source node name. The name of the connection's source node end point. Must be present for newly created nodes. For modification, if node name changes, the new node name must be used. This value can be a maximum of 255 characters.
source-node-resource-key		resource-key optional	Resource key of the source node. Must not be provided if node is new. For modification of existing topologies, this value must be provided.

Element definition: **storage-service-topology-node-info**

[top]

storage-service-topology-node-info contains the information about a node.

Name	Range	Type	Description
node-name		string	Name of the node. This value can be a maximum of 255 characters. Node name must be unique within the topology. When renaming an existing node during modify, the new node name must be specified in all the storage-service-topology-connection-info, that have this node as either the source or the destination.
node-resource-key		resource-key optional	Resource key of the node. Always present in the output. Auto-assigned for the new nodes. Must be included when referring to existing nodes, for example when modifying existing node attributes.
provision-attributes		provision-attributes optional	All Provisioning attributes for secondary storage. Should not be specified for the root node, and optional for non-root nodes.
resource-pools		resource-key[] optional	Pool of Data ONTAP resources that are used to provision secondary storage. Should not be specified for the root node, and optional for non-root nodes.
service-workflow-resource-key		resource-key optional	Provisioning workflow that is used to provision secondary storage. Should not be specified for the root node, and optional for non-root nodes. The name of storage-service-workflow is obtainable from storage-service-workflow-list-info API. This will be deprecated in future releases. Need to use "provision-attributes".

Element definition: **obj-full-name**

[top]

Full name of an object. This typedef is an alias for the builtin ZAPI type **string**. An object full name conforms to all the rules of an obj-name, except that the full name may be up to 255 characters long.

Full names are created by concatenating an object name with any parent object names, so as to create a unique name for an object. The format of full names is as follows:

- **cluster** full names are the either the fully-qualified domain name or the IP address of the cluster.
- **cluster-node** full names are the either the fully-qualified domain name or the IP address of the

cluster.

- **aggregate** full names are the cluster-node name and the aggregate name, separated by a colon, e.g. *cluster-node:aggr0*.
- **volume** full names are the vserver name and the volume name, separated by ":/", e.g. *vserver:/volume*. Note this does not include the "/vol" prefix. Volume and aggregate full names are distinguished by the presence of a forward slash after the colon.
- **qtree** full names are the containing volume full name and the qtree name, separated by a slash, e.g. *vserver:/volume/qtree*. The data not contained by any qtree may be represented by "-", e.g. *vserver:/volume/-*.
- **lun** full names are either a volume or qtree full name and the LUN path, separated by a slash, e.g. *vserver:/volume/LUN* or *vserver:/volume/qtree/LUN*.
- **initiator-group** full names are vserver name and the initiator group name, separated by a colon, e.g. *vserver:igroup*.
- **export-policy** full names are vserver name and the policy name, separated by a colon, e.g. *vserver:policy-name*.
- **lif** full names are a cluster, cluster-node, or vserver name and the interface name, separated by a colon, e.g. *cluster-name|cluster-node-name|vserver-name:lif*.
- **port-set** full names are the vserver name and the portset name, separated by a colon, e.g. *vserver:portset*.
- **fcp-target** full names are the cluster-node name and the target name, separated by a colon, e.g. *cluster-node:target*.

For any object not listed above, the obj-name and obj-full-name are identical.

[none]

Element definition: **provision-attributes**

[top]

Contains all provisioning attributes of secondary storage.

Name	Range	Type	Description
is-thin-provision		boolean optional	The flag to provision secondary storage as thin or thick. If true, secondary storage is provisioned as thin. By default true.
service-workflow-resource-key		resource-key optional	The resource key of storage-service-workflow is obtainable from storage-service-workflow-list-info API.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

APIs in Category: **system**

API version 9.12

system-about

[top]

Retrieve system information.

Output Name	Range	Type	Description
host-name		string	Hostname.
snmp-local-engine-id		string	The unique hexadecimal engine id used when sending SNMP V3 traps.
system-id		string	The system id for this installed instance of the application. This identifier is used to identify this application instance within MyAutosupport.
system-time		timestamp	System time in seconds since 00:00:00 Jan 1, 1970, UTC.
version		string	A string that adheres to the following regular expression: [1-9][0-9]*\.[0-9]+\.* The first number is the major version. The second number is the minor version. Example: 7.1RC1

Element definition: **timestamp**

[top]

Seconds since 1/1/1970 in UTC.

[none]

volume-by-aggregate-iter

[top]

Iterate over volumes contained entirely within a single aggregate. Currently returns flexible volumes and infinite volume constituents.

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • cluster • cluster-node • aggregate • volume If resource-filter identifies a volume, that single volume will be returned. If resource-filter resolves to more than one volume, all of them will be returned. If no resource-filter is provided, all volumes will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		volume-by-aggregate-info[] optional	The list of records.

Errno	Description
EINVALIDTAG	

volume-create

[top]

Create a flexible volume.

Input Name	Range	Type	Description
is-autoSize-enable		boolean optional	Enable autoSize for the volume. By default true.
is-compression-enable		boolean optional	Enable compression on the volume. By default false. If true, is-dedupe-enable flag should be true. Otherwize fails as EINVALIDINPUT error.
is-dedupe-enable		boolean optional	Enable dedupe for the volume. By default false.
is-thin-provision-enable		boolean optional	If true, secondary storage is provisioned as thin. By default true.
resource-pools		resource-key[]	List of resource pools used to provision storage.
size		string	**Flexible** volumes only. The initial size of the new flexible volume. The format to use is: kb|mb|gb|tb or KB|MB|GB|TB where 'kb'/'KB' means kilobytes, 'mb'/'MB' means megabytes, 'gb'/'GB' means gigabytes, and 'tb'/'TB' means terabytes. If the trailing unit character doesn't appear, then is interpreted as the number of bytes desired.
volume-name		string	Name of the volume to create. The volume name can contain letters, numbers, and the underscore character (_), but the first character must be a letter or an underscore. In Data ONTAP Cluster-Mode, the volume names must be unique within a Vserver.
vserver-resource-key		resource-key optional	Vserver that is used to provision storage. If not provided selects the vserver from Storage Virtual Machine Associations.

Output Name	Range	Type	Description
job-id		job-id	The identification of the job created by this API request. The job is a manifest that expresses the set of different tasks and the sequence in which they are executed in order to accomplish a high level work-flow.
vserver-resource-key		resource-key	Vserver that is used to provision storage.

Errno	Description
EINVALIDINPUT	
EOBJECTNOTFOUND	

volume-iter

[top]

Iterate over volumes.

Input Name	Range	Type	Description
			The maximum number of records per return batch the caller wants to

max-records		integer optional	receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • cluster • vserver • volume If resource-filter identifies a volume, that single volume will be returned. If resource-filter resolves to more than one volume, all of them will be returned. If no resource-filter is provided, all volumes will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		volume-info[] optional	The list of records.

Errno	Description
EINVALIDTAG	

volume-move-history-list-info-iter-end

[top]

Ends the iteration to list the history of volume moves.

Input Name	Range	Type	Description
tag		string	Tag from a previous volume-move-history-list-info-iter-start.

Errno	Description
EINVALIDTAG	

volume-move-history-list-info-iter-next

[top]

Get next few records in the iteration started by volume-move-history-list-info-iter-start.

Input Name	Range	Type	Description
maximum		integer	The maximum number of entries to retrieve.
tag		string	Tag from a previous volume-move-history-list-info-iter-start.

Output Name	Range	Type	Description
records		integer	The number of records actually returned.
volume-move-history		volume-move-history-info[]	List of volume move histories.

Errno	Description
EINVALIDINPUT	
EINVALIDTAG	
EOBJECTNOTFOUND	

volume-move-history-list-info-iter-start

[top]

Starts the iteration to list the history of volume moves.

Input Name	Range	Type	Description
volume-resource-key		resource-key optional	Name or identifier of a volume to list move histories for. If no volume-resource-key is provided, all volume move histories will be listed.

Output Name	Range	Type	Description
records		integer	Number which tells you how many items have been saved for future retrieval with volume-move-history-list-info-iter-next.
tag		string	An opaque handle used to identify a temporary store. Used in subsequent calls to volume-move-history-list-info-iter-next or volume-move-history-list-info-iter-end.

Errno	Description
EINVALIDINPUT	
EOBJECTNOTFOUND	

Element definition: **job-id**

[top]

Opaque identifier for a job.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **volume-by-aggregate-info**

[top]

Information about an aggregate-based volume.

Name	Range	Type	Description
actual-volume-size		integer optional	Actual size in bytes of the volume. For volumes which are destinations of a Volume SnapMirror relationship, the actual size of the volume may differ from the logical size (reported by the df command). The logical size for such volumes is equal to size of the source volume. For all other volume **actual-volume-size** will be same as **total** size.
afs-avail	**[0..2^63-1]**	integer optional	Number of bytes available in active file system. This will be (afs-total - afs-used) or the available space in the aggregate, whichever is lower.
afs-total	**[0..2^63-1]**	integer optional	Total number of bytes in active file system (total volume less snapshot reserve).
afs-used	**[0..2^63-1]**	integer optional	Number of bytes used to hold active file system data. This is what "df" reports as used for the volume. It includes data, hole reserves, overwrite reserves and snapshot overflow.
aggregate-name		string	Name of aggregate on which the volume resides.
aggregate-resource-key		resource-key	The resource key for the aggregate on which the volume resides.
block-type		file-system-block-type	File system block type of the volume.
constituent-volume-by-aggregate-info		constituent-volume-by-aggregate-info optional	This element contains data applicable exclusively to constituent volumes.
is-snapshot-enabled		boolean	Specifies whether the Snapshot copies are enabled for the volume.
snapshot-reserve-avail	**[0..2^63-1]**	integer optional	Number of available bytes in snapshot reserve for this volume. If snapshot-reserve-used is greater than snapshot-reserve-total, this value will be zero.
snapshot-reserve-total	**[0..2^63-1]**	integer optional	Total number of bytes of snapshot reserve.
snapshot-reserve-used	**[0..2^63-1]**	integer optional	Total number of bytes used to hold snapshot data. This can be greater than the snapshot reserve size but will not include any space used out of the overwrite reserve.
space-guarantee		string optional	The space reservation style associated with the volume. Possible values: • volume - Indicates that the entire size of the volume is pre-allocated. • file - Indicates that the space will be pre-allocated for all the space-reserved files and LUNs within the volume. Storage is not pre-allocated for files and LUNs that are not space-reserved. Writes to these can fail if the underlying aggregate has no space available to store the written data. • partial - Indicates that a FlexCache volume to reserve some

			amount of space that is less than its total size • none - Indicates that no space will be pre-allocated. This field does not appear if volume-state is restricted or offline.
total	**[0..2^63-1]**	integer optional	Total number of bytes in volume. This includes the snapshot reserve.
volume-efficiency-info		volume-efficiency-info optional	Volume efficiency information. This is returned only if deduplication or compression have ever been run on the volume.
volume-name		string	Name of the volume.
volume-resource-key		resource-key	The resource key for this volume.
volume-state		string	State of volume. Possible values are: • offline • online • restricted • unknown
volume-type		string	Type of volume. Possible values are: • rw - read-write • ls - load-sharing • dp - data-protection • dc - data-cache (FlexCache) • tmp temporary
vserver-name		string	Name of vserver on which the volume resides.
vserver-resource-key		resource-key	The resource key for the vserver on which the volume resides.

Element definition: **volume-info**

[top]

Information about a volume.

Name	Range	Type	Description
block-type		file-system-block-type	File system block type of the volume.
flexible-volume-info		flexible-volume-info optional	This element contains data applicable exclusively to flexible volumes.
infinite-volume-info		infinite-volume-info optional	This element contains data applicable exclusively to infinite volumes.
is-cft-		boolean	Whether the volume is in pre-commit state or not.

precommit			
is-junction-path-active		boolean optional	Whether the junction path of the volume is active. The value will not be set if the junction path is unavailable.
junction-parent-name		string optional	The name of the volume that contains the junction inode of the volume.
junction-parent-resource-key		resource-key optional	The resource key of the volume that contains the junction inode of the volume.
junction-path		string optional	Junction path of the volume.
space-guarantee		string optional	The space reservation style associated with the flexible volume. Possible values: • volume - Indicates that the entire size of the volume is pre-allocated. • file - Indicates that the space will be pre-allocated for all the space-reserved files and LUNs within the volume. Storage is not pre-allocated for files and LUNs that are not space-reserved. Writes to these can fail if the underlying aggregate has no space available to store the written data. • partial - Indicates that a FlexCache volume to reserve some amount of space that is less than its total size • none - Indicates that no space will be pre-allocated. This field does not appear if volume-state is restricted or offline.
volume-efficiency-info		volume-efficiency-info optional	Volume efficiency information. This is returned only if deduplication or compression have ever been run on the volume.
volume-name		string	Name of the volume.
volume-resource-key		resource-key	The resource key for this volume.
volume-security-info		volume-security-info optional	The Unix-oriented security settings associated with this volume.
volume-size		volume-size	Volume size information.
volume-snaplock-type		string optional	Snaplock-type of the volume. Possible values - "compliance", "enterprise" or "non-snaplock".
volume-state		string	State of volume. Possible values are: • offline • online • restricted • unknown
volume-status		obj-status	Current status of the volume based on all events
			Type of volume. Possible values are:

volume-type		string	• rw - read-write • ls - load-sharing • dp - data-protection • dc - data-cache (FlexCache) • tmp temporary
vserver-name		string	Name of vserver on which the volume resides.
vserver-resource-key		resource-key	The resource key for the vserver on which the volume resides.

Element definition: **volume-move-history-info**

[top]

Information about a volume move.

Name	Range	Type	Description
destination-aggregate-name		obj-name optional	Name of the aggregate where the volume is already moved or being moved to. If the destination aggregate is deleted, this element is not returned.
destination-aggregate-resource-key		resource-key optional	Resource key of the aggregate where the volume is already moved or being moved to. If the destination aggregate is deleted, this element is not returned.
source-aggregate-name		obj-name optional	Name of the aggregate where the volume that is being moved originally resided or is still residing. If the source aggregate is deleted, this element is not returned.
source-aggregate-resource-key		resource-key optional	Resource key of the aggregate where the volume that is being moved originally resided or is still residing. If the source aggregate is deleted, this element is not returned.
volume-move-completion-time		timestamp optional	Timestamp, in seconds, indicating the completion of the volume move operation since Jan 1, 1970 00:00:00 UTC.
volume-move-details		string	Provides details about the state of the volume move operation.
volume-move-estimated-completion-time		timestamp optional	Timestamp, indicating the estimated completion time of the volume move operation. Note that this time may keep increasing when the move goes into 'cutover', 'cutover hard deferred', 'cutover soft deferred' phase. In those cases where the input for cutover-action is wait, during data copy phase, estimated time of completion will approximate the time to reach cutover point and wait for user intervention. When the move is in queued phase this element is not returned.
volume-move-percent-complete	**[0..100]**	integer optional	Percentage of volume move completed thus far. When the move is in queued phase this element will show zero percent completion.
volume-move-phase		string	Phase of the volume move operation. The possible phases are 'queued', 'initializing', 'replicating', 'cutover', 'cutover hard deferred', 'cutover soft deferred', 'completed', 'cleaning up', 'failed', and 'restarting'.

volume-move-start-time		timestamp	Timestamp, in seconds, indicating the start of the volume move operation since Jan 1, 1970 00:00:00 UTC.
volume-move-state		string	State of the volume move operation. The possible states are 'in-progress', 'completed', 'failed', and 'paused'. The state 'paused' indicates that the volume move operation is moving into a cutover-deferred phase and is waiting for user intervention in the case of errors.
volume-name		obj-name	Name of the volume.
volume-resource-key		resource-key	Resource key of the volume.
vserver-name		obj-name	Name of vserver on which the volume resides.
vserver-resource-key		resource-key	Resource key of the vserver on which the volume resides.

Element definition: **constituent-volume-by-aggregate-info**

[top]

Attributes applicable only for constituent volumes.

Name	Range	Type	Description
infinite-volume-name		string	Name of the constituent's infinite volume.
infinite-volume-resource-key		resource-key	The resource key for constituent's infinite volume.
storage-class-name		string	Name of storage class on which the constituent resides.
storage-class-resource-key		resource-key	The resource key for the storage class on which the constituent resides.

Element definition: **file-system-block-type**

[top]

Block Type of the file system. The volumes on both the source and destination sides of a SnapMirror relationship must be of the same block type. Volumes contained in a larger parent agregate may have a block-type of 64_bit. For upgraded systems it is possible that this value may be unknown until the system can determine the block-type. Possible values are:

- 32_bit
- 64_bit
- unknown

[none]

Element definition: **flexible-volume-info**

[top]

Attributes applicable only for flexible volumes.

Name	Range	Type	Description
aggregate-name		string	Name of aggregate on which the volume resides.
aggregate-resource-key		resource-key	The resource key for the aggregate on which the volume resides.
autogrow-increment-size	**[0..2^63-1]**	integer optional	The increment size (in bytes) by which the volume would be grown.
is-autosize-enabled		boolean	Indicates whether this volume has autosize feature enabled or not.
is-snapshot-autodelete-enabled		boolean	Indicates whether this volume has snapshot autodelete enabled or not.
maximum-size	**[0..2^63-1]**	integer optional	Maximum size in bytes that this volume will be grown up to automatically by Data ONTAP. This is returned only if is-autosize-enabled is true.
volume-clone-info		volume-clone-info optional	Volume FlexClone information.

Element definition: **infinite-volume-info**

[top]

Attributes applicable only for infinite volumes.

Name	Range	Type	Description
is-managed-by-service		boolean	Specifies whether the infinite volume is managed by GUI services.
is-snapdiff-enabled		boolean	Specifies whether Snapdiff is enabled for the infinite volume.
max-data-constituent-size	**[0..2^63-1]**	integer optional	The maximum size (in bytes) of each data constituent in bytes.
max-namespace-constituent-size	**[0..2^63-1]**	integer optional	The maximum size (in bytes) of the namespace constituent.

Element definition: **obj-name**

[top]

Name of an object. This typedef is an alias for the built in ZAPI type **string**. An object name must conform to the following format:

- It must contain between 1 and 64 characters.
- It may start with any character and may contain any combination of characters, except that it may not consist solely of decimal digits ('0' through '9').
- In some contexts, a name may be the empty string (""), which is interpreted as a null value, e.g., a reference to no object at all.

The behavior of a ZAPI when it encounters an error involving an **obj-name** input element depends on how the ZAPI uses the input element. Here are the general rules:

- If the input name element is used to create a new object with the given name, or rename an existing object to that name, and the name does not conform to the above format, then the ZAPI fails with error code **EINVALIDINPUT**. Note that because **EINVALIDINPUT** is such a common error code, ZAPI specifications are not required to document cases when they may return it.
- If the input name element is used to refer to an existing object with that name, and there is no object with that name, then the ZAPI fails with error code **EOBJECTNOTFOUND**. Generally the ZAPI specification documents cases when it may return this error code.

A ZAPI may deviate from these general rules, for example, it may return more specific error codes. In such cases, the ZAPI specification must document its behavior.

If an input name element is used to refer to an existing object, then the ZAPI specification must specify which object type (e.g. cluster, vserver, volume etc.) is allowed. Some ZAPIs allow the object to be one of several different types. See the description of obj-full-name for examples of valid input formats.

Note that there is no requirement that all object names must be unique. However, the names for some specific types of objects are constrained such that no two objects of that type may have the same name.

[none]

Element definition: **obj-status**

[top]

A status value which can be associated with an object. This typedef is an alias for the builtin ZAPI type **string**. The severity associated with an event has this type.

Possible values are: 'normal', 'warning', 'error', 'critical'.

- normal: An object has normal status when it is working within the thresholds specified.
- warning: An object has the warning status when an event related to the object occurred that an administrator should know about. The event will not cause service disruption.
- error: An object has error status when it does not cause any service disruption, but it may affect performance.
- critical: An object has critical status when it is still performing, but service disruption may occur if corrective action is not taken immediately.

In some contexts, it is important that severities are ordered (as above). For example, an alert might be triggered if an event with a given severity "or worse" occurs. In this example, worse means "after" in the list above.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **timestamp**

[top]

Seconds since 1/1/1970 in UTC.

[none]

Element definition: **volume-efficiency-info**

[top]

Volume efficiency information. Optional fields will not be returned if deduplication or compression have never run on the volume.

Name	Range	Type	Description
compression-space-savings	**[0..2^63-1]**	integer optional	Space savings as a result of compression on the volume in bytes.
compression-space-savings-percentage	**[0..100]**	integer optional	Percentage of space savings generated by the compression.
dedupe-progress		string optional	The progress of the current deduplication operation on the volume with information as to which stage of de-duplication is currently in progress and how much data is processed for that stage. For eg: "25 MB Scanned, 20MB Searched, 40MB (20%) Done , 30MB Verified".
dedupe-schedule		string optional	Schedule for the deduplication operation on the volume. The field is returned if deduplication operation is scheduled.
dedupe-space-savings	**[0..2^63-1]**	integer optional	Space savings as a result of deduplication on the volume in bytes.
dedupe-space-savings-percentage	**[0..100]**	integer optional	Percentage of space savings generated by the shared space.
dedupe-status		string optional	Deduplication operation status of the volume. Possible values: "idle", "active", "pending", or "undoing".
dedupe-type		string optional	Type of the volume depending on whether it has a Snapvault qtree in it or not. Possible values: "regular", or "snapvault".
efficiency-policy		string optional	Policy for the deduplication operation on the volume. The field is returned if an efficiency policy is applied on the volume.
is-compression-enabled		boolean	Specifies if compression is enabled on the volume.
is-dedupe-enabled		boolean	Specifies if deduplication is enabled on the volume.
last-dedupe-begin-timestamp		timestamp optional	Start timestamp of the last deduplication operation.
last-dedupe-end-timestamp		timestamp optional	End timestamp of the last deduplication operation on the volume.
last-dedupe-error		string optional	A human readable error message of the last deduplication operation on the volume. Returned only when there is a valid error.
last-dedupe-	**[0..2^63-**	integer	The size scanned in bytes of the last deduplication operation on

scanned-size	**1]**	optional	the volume.
last-dedupe-status		string optional	Indicates the completion status of the last deduplication operation performed on the volume. Possible values: "success", or "failure".

Element definition: **volume-security-info**

[top]

Unix oriented security settings associated with this volume.

Name	Range	Type	Description
group-id		string optional	Unix group ID for the volume. The default value is 0 ('root').
permissions		string optional	Unix permission bits in octal string format. It is similar to unix style permission bits. In Data ONTAP 7-mode, the default setting of '0755' gives read/write/execute permissions to owner and read/execute to group and other users. In Clustered Data ONTAP, for security stype 'mixed' or 'unix', the default setting of '0755' gives read/write/execute permissions to owner and read/execute permissions to group and other users. For security style 'ntfs', the default setting of '0000' gives no permissions to owner, group and other users. It consists of 4 octal digits derived by adding up bits 4, 2 and 1. Omitted digits are assumed to be zeros. First digit selects the set user ID(4), set group ID (2) and sticky (1) attributes. The second digit selects permission for the owner of the file: read (4), write (2) and execute (1); The third selects permissions for other users in the same group; The fourth for other users not in the group.
user-id		string optional	Unix user ID for the volume. The default value is 0 ('root').

Element definition: **volume-size**

[top]

Collected size information about a volume.

Name	Range	Type	Description
actual-volume-size		integer optional	Actual size in bytes of the volume. For volumes which are destinations of a Volume SnapMirror relationship, the actual size of the volume may differ from the logical size (reported by the df command). The logical size for such volumes is equal to size of the source volume. For all other volume **actual-volume-size** will be same as **total** size.
afs-avail	**[0..2^63-1]**	integer optional	Number of bytes available in active file system. This will be (afs-total - afs-used) or the available space in the aggregate, whichever is lower.
afs-daily-growth-rate	**[0..100]**	integer optional	Percentage of volume used on a daily basis.
afs-total	**[0..2^63-1]**	integer optional	Total number of bytes in active file system (total volume less snapshot reserve).
afs-used	**[0..2^63-1]**	integer optional	Number of bytes used to hold active file system data. This is what "df" reports as used for the volume. It includes data, hole reserves, overwrite reserves and snapshot overflow.

afs-used-per-day	**[-2^44-1..2^44-1]**	integer	Number of bytes used per day in the active file system of the volume. This can be either positive or negative depending on the growth of used space in the volume.
is-snapshot-enabled		boolean	Specifies whether the Snapshot copies are enabled for the volume.
overwrite-reserve-avail	**[0..2^63-1]**	integer optional	Number of bytes available reserved space for data overwrites.
overwrite-reserve-total	**[0..2^63-1]**	integer optional	Total number of bytes reserved for data overwrites. This is the space reserved for overwriting LUNs and other space-reserved files when the volume has snapshots and afs-avail is zero.
overwrite-reserve-used	**[0..2^63-1]**	integer optional	Number of bytes used in overwrite reserve space.
quota-committed-space	**[0..2^63-1]**	integer optional	Number of bytes committed for quotas within the volume.
snapshot-reserve-avail	**[0..2^63-1]**	integer optional	Number of available bytes in snapshot reserve for this volume. If snapshot-reserve-used is greater than snapshot-reserve-total, this value will be zero.
snapshot-reserve-days-until-full	**[0..2^63-1]**	integer optional	Number of days until the volume snapshot reserve becomes full. This is returned only if volume snapshot reserve is defined and 'snapshot-reserve-used-per-day' is positive.
snapshot-reserve-total	**[0..2^63-1]**	integer optional	Total number of bytes of snapshot reserve.
snapshot-reserve-used	**[0..2^63-1]**	integer optional	Total number of bytes used to hold snapshot data. This can be greater than the snapshot reserve size but will not include any space used out of the overwrite reserve.
snapshot-reserve-used-per-day	**[-2^44-1..2^44-1]**	integer optional	Number of bytes used per day in the volume snapshot reserve. This can be either positive or negative depending on the growth of used space in the volume snapshot reserve. The information is based on the regression slope of the volume snapshot reserve usage history data. This is returned only if volume snapshot reserve is defined.
total	**[0..2^63-1]**	integer optional	Total number of bytes in volume. This includes the snapshot reserve.

Element definition: **timestamp**

[top]

Seconds since 1/1/1970 in UTC.

[none]

Element definition: **volume-clone-info**

[top]

Volume FlexClone information. Based on the volume type the following fields are displayed:

For a FlexClone volume, the fields displayed are "clone-space-savings", "clone-space-savings-

percentage", "is-clone-present", "is-clone", "parent-volume-resource-key" and "parent-volume-name".

For a parent volume containing FlexClone volumes, the fields displayed are "is-clone-present", "is-clone", and "clone-child-count".

For a volume that is both a FlexClone and parent volume, the fields displayed are "clone-space-savings", "clone-space-savings-percentage", "is-clone-present", "is-clone", "parent-volume-resource-key", "parent-volume-name", and "clone-child-count".

If the volume is neither a FlexClone volume nor a parent volume, then the "is-clone-present" and "is-clone" fields are displayed.

Name	Range	Type	Description
clone-child-count	**[1..2^32-1]**	integer optional	Number of FlexClone volumes for which this volume is parent.
clone-space-savings	**[0..2^63-1]**	integer optional	For a FlexClone volume this contains space savings as a result of sharing space with its FlexClone parent. For a Parent volume, the space savings is the sum of space savings realised by all its FlexClone volumes.
clone-space-savings-percentage	**[0..100]**	integer optional	Percentage of space savings generated by the shared FlexClone space.
is-clone		boolean	Specifies if the volume is a FlexClone volume.
is-clone-present		boolean	Specifies if the volume has one or more FlexClone volumes.
parent-volume-name		string optional	Name of the FlexClone parent volume.
parent-volume-resource-key		resource-key optional	Resource key of the FlexClone parent volume.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

vserver-iter

[top]

Iterates through list of Vservers

Input Name	Range	Type	Description
max-records		integer optional	The maximum number of records per return batch the caller wants to receive. The server may return smaller batch sizes based on performance constraints. If this field is not provided, then the server will return default number of records based on server performance.
resource-filter		resource-key optional	Resource by which to filter the result set. Any resource-filters specified in the first call must be included in subsequent calls. The allowed object types for this argument are: • cluster • vserver If resource-filter identifies a vserver, that single vserver will be returned. If resource-filter resolves to more than one vserver, all of them will be returned. If no resource-filter is provided, all vservers will be listed.
tag		string optional	Specify the tag from the last call. It is not specified for the first call. For subsequent calls, copy values from the 'next-tag' obtained from the previous call. Any resource-filters specified in the first call must be included in subsequent calls.

Output Name	Range	Type	Description
next-tag		string optional	Tag for the next call. Not present when there are no more objects to return.
num-records		integer	The number of records returned in this call.
records		vserver-info[] optional	The list of records.

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **vserver-info**

[top]

Vserver's information.

Name	Range	Type	Description
aggr-list		resource-key[]	The list of aggregates assigned for volume operations. These aggregates could be shared for use with other Vservers.
cifs-domain-info		cifs-domain-info optional	CIFS domain information of the Vserver.
cluster-name		string	Name of the cluster where the Vserver is present.
cluster-resource-key		resource-key	Resource key of the cluster where the Vserver is present.
dns-domains		dns-domain-name[] optional	An array of dns domain names for the Vserver.
is-dns-enabled		boolean optional	Specifies if dns service is enabled for the Vserver.
is-kerberos-enabled		boolean optional	Specifies if Kerberos security is enabled for the Vserver.
is-ldap-client-enabled		boolean optional	Specifies if LDAP client configuration is enabled for the Vserver.
is-nis-enabled		boolean optional	Specifies if nis service is enabled for the Vserver.
is-repository		boolean	Specifies if the Vserver contains Infinite Volume.
nis-domain-name		string optional	Active nis domain name for the Vserver.
root-volume-name		string	Name of the Vserver root volume.
service-statuses		vserver-service-status[]	List of services and their statuses.
vserver-name		string	Name of the Vserver. Length: [1..255]
vserver-resource-key		resource-key	Resource key of the Vserver.
vserver-state		string	Current state of the Vserver. Possible values: "running", "stopped", "starting" or "stopping".
vserver-status		obj-status	Current status of the Vserver based on all events.

Element definition: **cifs-domain-info**

[top]

Details of the cifs domain.

Name	Range	Type	Description
cifs-dns-domain-name		string	Specifies the CIFS DNS domain name. Maximum length: 255 characters. */
cifs-domain-name		string	Specifies the NETBIOS name of the cifs domain. Maximum length: 255 characters.

Element definition: **dns-domain-name**

[top]

DNS domain name for the Vserver.

[none]

Element definition: **obj-status**

[top]

A status value which can be associated with an object. This typedef is an alias for the builtin ZAPI type **string**. The severity associated with an event has this type.

Possible values are: 'normal', 'warning', 'error', 'critical'.

- normal: An object has normal status when it is working within the thresholds specified.
- warning: An object has the warning status when an event related to the object occurred that an administrator should know about. The event will not cause service disruption.
- error: An object has error status when it does not cause any service disruption, but it may affect performance.
- critical: An object has critical status when it is still performing, but service disruption may occur if corrective action is not taken immediately.

In some contexts, it is important that severities are ordered (as above). For example, an alert might be triggered if an event with a given severity "or worse" occurs. In this example, worse means "after" in the list above.

[none]

Element definition: **resource-key**

[top]

A self-describing string identifier for a managed resource.

[none]

Element definition: **vserver-service-status**

[top]

Status indicating whether a service is up or down.

Name	Range	Type	Description
		string	

service		optional	Name of the service. Possible values: "nfs", "cifs", "iscsi" or "fcp".
status		string optional	Indicates whether the service is up or not. Possible values: "up", "down".

Copyright Notice

Trademark Information

Support

You can use the following support channels to obtain more information about and help with the NetApp Manageability SDK:

The NetApp Manageability SDK Discussion Forum:

Go to http://developer.netapp.com to post query in the discussion forum.

- This forum is the preferred way to obtain support.
- Use this forum for non-confidential queries and issues regarding the SDK.

The NetApp Manageability SDK mailing list:

dl-sdk-answers@netapp.com

Use this mailing list to post any confidential information.

Customer Success Services (CSS):

Technical Support
mysupport.netapp.com
888.4.NETAPP (US and Canada)
(888.463.8277) 00.800.44.NETAPP (EMEA/Europe)
+800.800.80.800 (Asia/Pacific)

- NetApp Manageability SDK customers and end customers of partner products can contact CSS to log issues related to the SDK.
- However, CSS does not offer support for SDK usage during solution development.

Machine Translation

See important information about localized content at netapp.com.

http://www.netapp.com/us/legal/netapptmlist.aspx