# NetApp

# Common objects

## Element Software

NetApp
November 19, 2025

# Table of Contents

# Common objects

The Element software API uses JSON objects to represent organized data concepts. Many of these API methods make use of these objects for data input and output. This section documents these commonly used objects; objects that are only used within a single method are documented with that method instead of in this section.

- account
- authSessionInfo
- bulkVolumeJob
- binding (virtual volumes)
- certificateDetails
- cluster
- clusterAdmin
- clusterCapacity
- clusterConfig
- clusterInfo
- clusterPair
- clusterStats
- clusterStructure
- drive
- driveStats
- error
- event
- fault
- fibreChannelPort
- fipsErrorNodeReport
- fipsNodeReport
- fipsReport
- groupSnapshot
- hardwareInfo
- host (virtual volumes)
- idpConfigInfo
- initiator
- ISCSIAuthentication
- keyProviderKmip
- keyServerKmip
- ldapConfiguration

- snapMirrorVolumeInfo
- snapMirrorVserver
- snapMirrorVserverAggregateInfo
- snapshot
- snmpTrapRecipient
- storageContainer
- syncJob
- task (virtual volumes)
- usmUser
- virtualNetwork
- virtualVolume
- volume
- volumeAccessGroup
- volumePair
- volumeStats

# Find more information

- SolidFire and Element Software Documentation
- Documentation for earlier versions of NetApp SolidFire and Element products

# account

The `account` object contains information about an account. This object includes only "configured" information about the account, not any runtime or usage information.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| `accountID` | The unique account ID for the account. | integer |
| `attributes` | List of name-value pairs in JSON object format. | JSON object |
| `enableChap` | Specifies whether CHAP account credentials can be used by an initiator to access volumes. | boolean |
| `initiatorSecret` | The initiator CHAP secret. | string |

| Name | Description | Type |
|------|-------------|------|
| status | The current status of the account. Possible values:<br><br>• active: An active account.<br>• locked: A locked account.<br>• removed: An account that has been deleted and purged. | string |
| storageContainerID | The unique ID of the virtual volume storage container associated with this account. | UUID |
| targetSecret | The target CHAP secret. | string |
| username | The username for the account. | string |
| volumes | A list of volume IDs for volumes owned by this account. | integer array |

## Find more information

- [AddAccount](#)
- [GetAccountByID](#)
- [GetAccountByName](#)
- [ListAccounts](#)

# authSessionInfo

The `authSessionInfo` object contains information about an auth session.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| accessGroupList | List of access groups for the user. | string array |

| Name | Description | Type |
|---|---|---|
| authMethod | The type of authorization the cluster admin user has. Possible values:<br><br>• LDAP - authenticated via LDAP.<br><br>• Cluster - authenticated via a username and password stored in the cluster database.<br><br>• IdP - authenticated via a third-party Identity Provider. | string |
| clusterAdminIDs | List of cluster AdminID(s) associated with this session. For sessions related to LDAP or a third-party Identity Provider (IdP), this will be an aggregate list of matching Cluster AdminIDs associated with this session. | integer array |
| finalTimeout | Time at which the session becomes invalid.This is set when the session is created and cannot be changed. | string |
| idpConfigVersion | IdP configuration version when the session was created. | integer |
| lastAccessTimeout | Time at which the session becomes invalid due to inactivity.It is set to a new value when the session is accessed for use, up to the time where the session becomes invalid due to finalTimeout being reached. | string |
| sessionCreationTime | Time at which the session is created. | string |
| sessionID | UUID for this session. | UUID |

| Name | Description | Type |
|------|-------------|------|
| username | Username associated with this session. For sessions related to LDAP, this will be the user's LDAP DN. For sessions related to a third-party IdP, this will be an arbitrary name-value pair that will be used for auditing operations within the session. It will not necessarily match a cluster admin name on the cluster. For example, a SAML Subject NameID,but this will be dictated by the configuration of the IdP and the resultant content of the SAML assertion. | string |

# bulkVolumeJob

The `bulkVolumeJob` object contains information about bulk volume read or write operations, such as cloning or snapshot creation.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| attributes | JSON attribute of the bulk volume job. | JSON object |
| bulkVolumeID | The internal bulk volume job ID. | integer |
| createTime | Timestamp created for the bulk volume job in UTC+0 format. | ISO 8601 date string |
| elapsedTime | The number of seconds since the job began. | string |
| format | The format of the bulk volume operation. Possible values:<br><br>• native<br>• uncompressed | string |
| key | The unique key created by the bulk volume session. | string |

| Name | Description | Type |
|---|---|---|
| percentComplete | The completed percentage reported by the operation. | integer |
| remainingTime | The estimated time remaining in seconds. | integer |
| srcVolumeID | The source volume ID. | integer |
| status | The status of the operation. Possible values:<br><br>• preparing<br>• running<br>• complete<br>• failed | string |
| script | The name of the script if one is provided. | string |
| snapshotID | The ID of the snapshot if a snapshot is in the source of the bulk volume job. | integer |
| type | The type of bulk operation. Possible values:<br><br>• read<br>• write | string |

# binding (virtual volumes)

The binding object contains information about the binding for a virtual volume. You can retrieve a list of this information for all virtual volumes using the `ListVirtualVolumeBindings` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| protocolEndpointID | The unique ID of the protocol endpoint. | UUID |

| Name | Description | Type |
|---|---|---|
| `protocolEndpointInBandID` | The scsiNAADeviceID of the protocol endpoint. | string |
| `protocolEndpointType` | The type of protocol endpoint. SCSI is the only value returned for the protocol endpoint type. | string |
| `virtualVolumeBindingID` | The unique ID of the virtual volume binding object. | integer |
| `virtualVolumeHostID` | The unique ID of the virtual volume host. | UUID |
| `virtualVolumeID` | The unique ID of the virtual volume. | UUID |
| `virtualVolumeSecondaryID` | The secondary ID of the virtual volume. | string |

## Find more information

- ListVirtualVolumeBindings
- protocolEndpoint

# certificateDetails

The `certificateDetails` object contains the decoded information about a security certificate.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| `issuer` | The name of the issuer. | string |
| `modulus` | The modulus of the public key. | string |
| `notAfter` | The expiry date of the certificate. | ISO 8601 string |
| `notBefore` | The start date of the certificate. | ISO 8601 string |
| `serial` | The certificate serial number. | string |

| Name | Description | Type |
|---|---|---|
| sha1Fingerprint | The digest of the DER-encoded version of the certificate. | string |
| subject | The subject name. | string |

# cluster

The cluster object contains information that the node uses to communicate with the cluster. You can retrieve this information with the GetClusterConfig API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| cipi | Network interface used for cluster communication. | string |
| cluster | Unique cluster name. | string |
| encryptionCapable | Indicates whether the node supports drive encryption. | boolean |
| ensemble | The nodes that are participating in the cluster. | string array |
| fipsDriveConfiguration | Indicates whether the node supports FIPS 140-2 certified drives. | boolean |
| mipi | The network interface used for node management. | string |
| name | The cluster name. | string |
| nodeID | The node ID of the node in the cluster. | string |
| pendingNodeID | The ID of the pending node in the cluster. | integer |
| role | Identifies the role of the node. | integer |

| Name | Description | Type |
|---|---|---|
| sipi | The network interface used for storage traffic. | string |
| state | The current state of the node. Possible values:<br><br>• Available: The node has not been configured with a cluster name.<br><br>• Pending: The node is pending for a specific named cluster and can be added.<br><br>• Active: The node is an active member of a cluster and cannot be added to another cluster.<br><br>• PendingActive: The node is currently being returned to the factory software image, and is not yet an active member of a cluster. When complete, it will transition to the Active state. | string |
| version | The version of software running on the node. | string |

## Member modifiability and node states

This table indicates whether or not the object parameters can be modified at each possible node state.

| Parameter name | Available state | Pending state | Active state |
|---|---|---|---|
| cipi | No | No | No |
| cluster | Yes | Yes | No |
| encryptionCapable | No | No | No |
| ensemble | No | No | No |
| mipi | Yes | Yes | No |
| name | Yes | Yes | Yes |
| nodeID | No | No | No |

| | | | |
|---|---|---|---|
| pendingNodeID | No | No | No |
| role | No | No | No |
| sipi | No | No | No |
| state | No | No | No |
| version | No | No | No |

## Find more information

GetClusterConfig

# clusterAdmin

The clusterAdmin object contains information about the current cluster administrator user. You can retrieve admin user information with the GetCurrentClusterAdmin API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| access | The methods this cluster admin can use. | string array |
| authMethod | The type of authorization the cluster admin user has. Possible values:<br><br>&bull; LDAP<br><br>&bull; Cluster<br><br>&bull; Local | string |
| attributes | List of name-value pairs in JSON object format. | JSON object |
| clusterAdminID | The cluster administrator ID for this cluster admin user. | integer |
| username | User name for this cluster admin. | string |

# clusterCapacity

The clusterCapacity object contains high-level capacity measurements for the cluster. You can get cluster capacity information with the GetClusterCapacity API method. Space measurements in the object members are calculated in bytes.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| activeBlockSpace | The amount of space on the block drives. This includes additional information such as metadata entries and space which can be cleaned up. | integer |
| activeSessions | The number of active iSCSI sessions communicating with the cluster. | integer |
| averageIOPS | The average IOPS for the cluster since midnight Coordinated Universal Time (UTC). | integer |
| clusterRecentIOSize | The average size of IOPS to all volumes in the cluster. | integer |
| currentIOPS | The average IOPS for all volumes in the cluster over the last 5 seconds. | integer |
| maxIOPS | The estimated maximum IOPS capability of the current cluster. | integer |

| Name | Description | Type |
|---|---|---|
| maxOverProvisionableSpace | The maximum amount of provisionable space. This is a computed value. You cannot create new volumes if the current provisioned space plus the new volume size would exceed this number. The value is calculated as follows: `maxOverProvisionableSpace = maxProvisionedSpace * maxMetadataOverProvisionFactor` | integer |
| maxProvisionedSpace | The total amount of provisionable space if all volumes are 100% filled (no thin provisioned metadata). | integer |
| maxUsedMetadataSpace | The number of bytes on volume drives used to store metadata. | integer |
| maxUsedSpace | The total amount of space on all active block drives. | integer |
| nonZeroBlock | The total number of 4KiB blocks that contain data after the last garbage collection operation has completed. | integer |
| peakActiveSessions | The peak number of iSCSI connections since midnight UTC. | integer |
| peakIOPS | The highest value for currentIOPS since midnight UTC. | integer |
| provisionedSpace | The total amount of space provisioned in all volumes on the cluster. | integer |
| timestamp | The date and time, in UTC+0 format, that this cluster capacity sample was taken. | ISO 8601 string |
| totalOps | The total number of I/O operations performed throughout the lifetime of the cluster. | integer |

| Name | Description | Type |
|---|---|---|
| uniqueBlocks | The total number of blocks stored on the block drives. The value includes replicated blocks. | integer |
| uniqueBlocksUsedSpace | The total amount of data the uniqueBlocks take up on the block drives. See the GetclusterCapacity method for information about how this number relates to the uniqueBlocks value. | integer |
| usedMetadataSpace | The total number of bytes on volume drives used to store metadata. | integer |
| usedMetadataSpaceInSnapshots | The number of bytes on volume drives used for storing unique data in snapshots. This number provides an estimate of how much metadata space would be regained by deleting all snapshots on the system. | integer |
| usedSpace | The total amount of space used by all block drives in the system. | integer |
| zeroBlocks | The total number of empty 4KiB blocks without data after the last round of garbage collection operation has completed. | integer |

## Find more information

GetClusterCapacity

# clusterConfig

The `clusterConfig` object returns information the node uses to communicate with the cluster.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| `cipi` | Network interface used for cluster communication. | string |
| `cluster` | Unique name of the cluster. | string |
| `encryptionCapable` | Specifies whether the node supports encryption. | boolean |
| `ensemble` | Nodes that are participating in the cluster. | string array |
| `fipsDriveConfiguration` | Specifies whether the node supports FIPS 140-2 certified drives. | boolean |
| `hasLocalAdmin` | Specifies whether the cluster has a local administrator. | boolean |
| `mipi` | Network interface used for node management. | string |
| `name` | Unique identifier for the cluster. | string |
| `nodeID` | Unique identifier for the node. | integer |
| `pendingNodeID` | Unique identifier for the pending node. | integer |
| `role` | Identifies the role of the node. | string |
| `sipi` | Network interface used for storage. | string |
| `state` | Indicates the state of the node. | string |
| `version` | Indicates the version of the node. | string |

# clusterInfo

The clusterInfo object contains information that the node uses to communicate with the cluster. You can get this information with the GetClusterInfo API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| attributes | List of name-value pairs in JSON object format. | JSON object |
| defaultProtectionScheme | The protection scheme used by default for new volumes, unless a protection scheme is provided with the CreateVolume method call. This protection scheme must always be in the set of enabled protection schemes. | string |
| enabledProtectionSchemes | A list of all protection schemes that have been enabled on this storage cluster. | string array |
| encryptionAtRestState | The state of the Encryption at Rest feature. Possible values:<br><br>• Enabling: Encryption at rest is being enabled.<br>• Enabled: Encryption at rest is enabled.<br>• Disabling: Encryption at rest is being disabled.<br>• Disabled: Encryption at rest is disabled. | string |
| ensemble | The nodes that are participating in the cluster. | string array |
| mvip | The floating (virtual) IP address for the cluster on the management network. | string |
| mvipInterface | The physical interface associated with the MVIP address. | string |
| mvipNodeID | The node that holds the master MVIP address. | integer |
| mvipVlanTag | The VLAN identifier for the MVIP address. | string |

| Name | Description | Type |
|------|-------------|------|
| name | The unique cluster name. | string |
| repCount | The number of replicas of each piece of data to store in the cluster. The valid value is "2". | integer |
| softwareEncryptionAtRestState | Software-based encryption-at-rest state. | string |
| supportedProtectionSchemes | A list of all protection schemes that are supported on this storage cluster. | string array |
| svip | The floating (virtual) IP address for the cluster on the storage (iSCSI) network. | string |
| svipInterface | The physical interface associated with the master SVIP address. | string |
| svipNodeID | The node holding the master SVIP address. | integer |
| svipVlanTag | The VLAN identifier for the master SVIP address. | string |
| uniqueID | The unique ID for the cluster. | string |
| uuid | The unique identifier for the cluster. | UUID |

## Find more information

- GetClusterInfo
- SolidFire and Element Software Documentationentation]
- Documentation for earlier versions of NetApp SolidFire and Element products

# clusterPair

The clusterPair object contains information about clusters paired with the local cluster. You can retrieve a list of clusterPair objects for the local cluster with the ListClusterPairs method.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| clusterName | The name of the other cluster in the pair. | string |
| clusterPairID | A unique ID given to each cluster in the pair. | integer |
| clusterPairUUID | The universally unique identifier for the cluster pair. | string |
| UUID | Unique identifier for the remote cluster in the cluster pair. | integer |
| latency | The latency, in milliseconds, between clusters. | integer |
| mvip | The IP address of the management connection for paired clusters. | string |
| status | The status of the connection between the paired clusters. Possible values:<br><br>• Unconfigured<br><br>• Connected<br><br>• Misconfigured<br><br>• Disconnected | string |
| version | The Element version of the other cluster in the pair. | string |

## Find more information

ListClusterPairs

# clusterStats

The clusterStats object contains statistical data for a cluster. Many of the volume-related statistics contained in the object are averaged for all volumes in the cluster. You can use the GetClusterStats method to retrieve this information for a cluster.

## Object members

This object contains the following members:

| Name | Description | Calculation | Type |
|------|-------------|-------------|------|
| actualIOPS | Current actual IOPS for the entire cluster in the last 500 milliseconds. | Point in time | integer |
| averageIOPSize | Average size in bytes of recent I/O to the cluster in the last 500 milliseconds. | Point in time | integer |
| clientQueueDepth | The number of outstanding read and write operations to the cluster. | N/A | integer |
| clusterUtilization | The percentage of the cluster's max IOPS currently being utilized. This is computed as clusterUtilization = normalizedIOPS / maxIOPS (from GetClusterCapacity). | N/A | float |
| latencyUSec | The average time, in microseconds, to complete operations to a cluster in the last 500 milliseconds. | Point in time | integer |
| normalizedIOPS | Average number of IOPS for the entire cluster in the last 500 milliseconds. | Point in time | integer |
| readBytes | The total cumulative bytes read from the cluster since the creation of the cluster. | Monotonically increasing | integer |
| readBytesLastSample | The total number of bytes read from the cluster during the last sample period. | Point in time | integer |
| readLatencyUSec | The average time, in microseconds, to complete read operations to the cluster in the last 500 milliseconds. | Point in time | integer |

| Name | Description | Calculation | Type |
|---|---|---|---|
| readLatencyUSecTotal | The total time spent performing read operations since the creation of the cluster. | Monotonically increasing | integer |
| readOps | The total cumulative read operations to the cluster since the creation of the cluster. | Monotonically increasing | integer |
| readOpsLastSample | The total number of read operations during the last sample period. | Point in time | integer |
| samplePeriodMSec | The length of the sample period, in milliseconds. | N/A | integer |
| servicesCount | The number of services running on the cluster. If equal to the servicesTotal, this indicates that valid statistics were collected from all nodes. | Point in time | integer |
| servicesTotal | The total number of expected services running on the cluster. | N/A | integer |
| timestamp | The current time in UTC+0 format. | N/A | ISO 8601 date string |
| unalignedReads | The total cumulative unaligned read operations to a cluster since the creation of the cluster. | Monotonically increasing | integer |
| unalignedWrites | The total cumulative unaligned write operations to a cluster since the creation of the cluster. | Monotonically increasing | integer |
| writeBytes | The total cumulative bytes written to the cluster since the creation of the cluster. | Monotonically increasing | integer |

| Name | Description | Calculation | Type |
|------|-------------|-------------|------|
| writeBytesLastSample | The total number of bytes written to the cluster during the last sample period. | Monotonically increasing | integer |
| writeLatencyUSec | The average time, in microseconds, to complete write operations to a cluster in the last 500 milliseconds. | Point in time | integer |
| writeLatencyUSecTotal | The total time spent performing write operations since the creation of the cluster. | Monotonically increasing | integer |
| writeOps | The total cumulative write operations to the cluster since the creation of the cluster. | Monotonically increasing | integer |
| writeOpsLastSample | The total number of write operations during the last sample period. | Point in time | integer |

## Find more information

[GetClusterStats](#)

# clusterStructure

The clusterStructure object holds cluster configuration backup information created by the GetClusterStructure method. You can use the SetClusterStructure method to restore this information to a storage cluster you are rebuilding.

## Object members

This object contains the combined return information from the following methods:

- [GetClusterInfo](#)
- [ListAccounts](#)
- [ListInitiators](#)
- [ListVolumes](#) (with includeVirtualVolumes=false)
- [ListVolumeAccessGroups](#)
- [ListStorageContainers](#)

- ListQoSPolicies
- GetSnmpInfo
- GetNtpInfo
- ListVirtualNetworks
- ListClusterAdmins
- ListSchedules
- ListSnapMirrorEndpoints
- GetFeatureStatus
- GetLdapConfiguration
- GetRemoteLoggingHosts
- GetDefaultQoS
- GetVolumeAccessGroupLunAssignments

## Find more information

- GetClusterStructure
- SetClusterStructure

# drive

The drive object contains information about individual drives in the cluster's active nodes. This object contains details on drives that have been added as volume metadata or block drives, as well as drives that have not yet been added and are available. You can retrieve this information with the `ListDrives` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
| --- | --- | --- |
| attributes | List of name-value pairs in JSON object format. This object is always null and is not modifiable. | JSON object |
| capacity | The total capacity of the drive, in bytes. | integer |
| chassisSlot | For HCI platforms, this value is the node letter and slot number in the server chassis where this drive is located. For storage platforms, the slot number is a string representation of the "slot" integer. | string |

| Name | Description | Type |
|---|---|---|
| driveFailureDetail | If a drive's status is "Failed", this field provides more detail on why the drive was marked failed. | string |
| driveID | The ID of this drive. | integer |
| driveSecurityFaultReason | If enabling or disabling drive security failed, the reason why it failed. If the value is "none", there was no failure. | string |
| keyID | The keyID used by the key provider to acquire the authentication key for unlocking this drive. | UUID |
| keyProviderID | Identifies the provider of the authentication key for unlocking this drive. | integer |
| nodeID | The ID of the node containing this drive. | integer |
| segmentFileSize | The segment file size of the drive, in bytes. | integer |
| serial | The drive serial number. | string |
| slot | The slot number in the server chassis where this drive is located, or -1 if a SATADimm device is used for the internal metadata drive. | integer |

| Name | Description | Type |
|------|-------------|------|
| status | The status of the drive. Possible values:<br><br>• available: An available drive.<br><br>• active: An active drive.<br><br>• erasing: A drive is in the process of being secure erased. Any data on that drive is permanently removed.<br><br>• failed: A drive that has failed. Any data that was previously on the drive has been migrated to other drives in the cluster.<br><br>• removing: A drive is in the process of being removed. Any data previously on the drive is being migrated to other drives in the cluster. | string |
| type | The type of drive. Possible values:<br><br>• volume: Stores volume metadata.<br><br>• block: Stores block data.<br><br>• unknown: Drive type not yet active and is yet to be determined. | string |
| usableCapacity | The usable capacity of the drive, in bytes. | integer |

## Find more information

ListDrives

# driveStats

The driveStats object contains high-level activity measurements for a single drive. You can retrieve measurement information with the API method `GetDriveStats`.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| activeSessions | Number of iSCSI sessions currently using this drive (only present for metadata drives). | integer |
| driveID | Unique ID of the drive in the cluster. | integer |
| failedDieCount | Number of failed drive hardware elements. | integer |
| iosInProgress | The number of I/Os to this drive that are in progress. | integer |
| lifeRemainingPercent | Drive media wear out indicator. | integer |
| lifetimeReadBytes | Total bytes read from this drive for the lifetime of the drive. | integer |
| lifetimeWriteBytes | Total bytes written to this drive for the lifetime of the drive. | integer |
| powerOnHours | Number of hours this drive has been powered on. | integer |
| reads | The number of read() calls per second to this drive. | integer |
| readBytes | Total bytes read from the drive due to client operations. | integer |
| readsCombined | The number of read() calls to adjacent sectors that could be combined into a larger read. | integer |
| readMsec | The number of milliseconds spent reading. | integer |
| readOps | Total read operations on the drive due to client operations. | integer |
| reallocatedSectors | Number of bad sectors replaced in this drive. | integer |
| reserveCapacityPercent | The available reserve capacity of the drive. | integer |

| Name | Description | Type |
|------|-------------|------|
| timestamp | The current time in UTC+0 format. | ISO 8601 date string |
| totalCapacity | Total capacity of the drive, in bytes. | integer |
| uncorrectableErrors | The Reported Uncorrectable Errors value from the Self-Monitoring, Analysis and Reporting Technology (SMART) monitoring system in the drive. | integer |
| usedCapacity | Used capacity of the drive, in bytes. | integer |
| usedMemory | Amount of memory currently used by the node hosting this drive. | integer |
| writes | The number of write() calls per second to this drive. | integer |
| writeBytes | Total bytes written to the drive due to client activity. | integer |
| writesCombined | The number of write() calls to adjacent sectors that could be combined into a larger write. | integer |
| writeMsec | The number of milliseconds spent writing. | integer |
| writeOps | Total write operations to the drive due to client activity. | integer |

## Find more information

[GetDriveStats](#)

# error

The error object contains an error code and message if an error occurs during a method call. All system-generated errors have an error code of 500.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| code | The numeric code used to identify the error. All system-generated errors return a code of 500. | integer |
| name | The unique identifier for the specific error that occurred. Each method returns a documented set of errors, although you should be prepared to handle unrecognized errors as well. | string |
| message | A description of the error, possibly with additional details. | string |

# event

The event object contains details of events that occur during an API method call or while the system is performing an operation.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| details | Extra information about the event. | JSON object |
| driveID | The driveID of the drive reporting the failure. 0 if not applicable. | integer |
| driveIDs | A list of the driveIDs of the drives reporting the failure. An empty list if not applicable. | integer array |
| eventID | Unique ID associated with each event. | integer |
| eventInfoType | The type of fault. | string |
| message | A string description of the event that occurred. | string |
| nodeID | The nodeID of the node reporting the failure. 0 if not applicable. | integer |

| Name | Description | Type |
|---|---|---|
| serviceID | The serviceID of the service reporting the failure. 0 if not applicable. | integer |
| severity | Severity the event is reporting. | integer |
| timeOfPublish | The time at which the cluster's event log received the event, in UTC+0 format. | ISO 8601 date string |
| timeOfReport | The time at which the event occurred on the cluster, in UTC+0 format. | ISO 8601 date string |

**Note:** There might be a slight difference between timeOfReport and timeOfPublish if the event occurred and was not able to be immediately published.

## Event types

The following list describes the possible event types that the eventInfoType member can contain:

- apiEvent: Events initiated through the API or web UI that modify settings.
- binAssignmentsEvent: Events related to the assignment of data to internal containers.
- binSyncEvent: Events related to a reassignment of data among block services.
- bsCheckEvent: Events related to block service checks.
- bsKillEvent: Events related to block service terminations.
- bulkOpEvent: Events that operate on an entire volume, such as a volume backup, restore, snapshot, or clone.
- cloneEvent: Events related to volume cloning.
- clusterMasterEvent: Cluster configuration change events such as adding or removing nodes.
- dataEvent: Events related to reading and writing data.
- dbEvent: Events related to the ensemble node database.
- driveEvent: Events related to drive operations.
- encryptionAtRestEvent: Events related to stored data encryption.
- ensembleEvent: Events related to ensemble size increase or decrease.
- fibreChannelEvent: Events related to Fibre Channel node configuration or connections.
- gcEvent: Events related to garbage collection. These processes run every 60 minutes to reclaim storage on block drives.
- ieEvent: Events related to internal system errors.
- installEvent: Eevnts related to automatic software installation on pending storage nodes.
- iSCSIEvent: Events related to iSCSI connection or configuration issues.

- limitEvent: Events related to the number of volumes or virtual volumes in an account or in the cluster nearing the maximum allowed.
- networkEvent: Events related to virtual networking.
- platformHardwareEvent: Events related to issues detected on hardware devices.
- remoteClusterEvent: Events related to remote cluster pairing.
- schedulerEvent: Events related to scheduled snapshots.
- serviceEvent: Events related to system service status.
- statEvent: Events related to system statistics.
- sliceEvent: Events related to metadata storage.
- snmpTrapEvent: Events related to SNMP traps.
- tsEvent: System transport service events.
- unexpectedException: Events related to unexpected errors.
- vasaProviderEvent: Events related to a VMware VASA provider.

## Find more information

[ListEvents](ListEvents)

# fault

The fault object contains information about faults that are detected in the cluster. The `ListClusterFaults` method returns cluster fault information.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| blocksUpgrade | The fault blocks an upgrade. Possible values:<br><br>• true: The fault blocks an upgrade.<br><br>• false: The fault does not block an upgrade. | boolean |
| clusterFaultID | The unique ID associated with each cluster fault. | integer |
| code | The fault code for the specific fault that was detected. For further details, see Cluster Fault Codes. | string |
| data | Additional fault-specific information. | JSON object |

| Name | Description | Type |
|---|---|---|
| date | The current time in UTC+0 format. | ISO 8601 string |
| details | The description of the fault with additional details. | string |
| driveID | The first drive ID in the driveIDs list. If the driveIDs list is empty (which means that no faults were returned that deal with drives), this value is 0. | integer |
| driveIDs | A list of driveID values for the drives that this fault refers to. Included for faults dealing with drives. If none, this is an empty array. | integer array |
| nodeHardwareFaultID | The identifier assigned to a hardware fault on the cluster. | integer |
| nodeID | The node ID for the node that this fault refers to. Included for node and drive faults, otherwise set to 0. | integer |
| resolved | The resolved status of the fault. Possible values:<br><br>• true: The fault is no longer detected.<br>• false: The fault is still present. | boolean |
| resolvedDate | The date and time the fault was resolved. | ISO 8601 string |
| serviceID | The service associated with the fault. This value is "0" (zero) if the fault is not associated with a service. | integer |

| Name | Description | Type |
|------|-------------|------|
| severity | The severity of the fault. Possible values:<br><br>• warning: A minor issue. The cluster is functioning and upgrades are allowed at this severity level.<br><br>• error: A failure that generally should not affect service (except possible performance degradation or loss of HA). Some features might be disabled.<br><br>• critical: A serious failure that is affecting service. The system is unable to serve API requests or client I/O and is at risk of data loss.<br><br>• bestPractice: Faults triggered by sub-optimal system configuration. | string |
| type | The type of fault. Possible values:<br><br>• node: A fault affecting an entire node.<br><br>• drive: A fault affecting an individual drive.<br><br>• cluster: A fault affecting the entire cluster.<br><br>• service: A fault affecting a service on the cluster.<br><br>• volume: A fault affecting an individual volume. | string |

## Find more information

- ListClusterFaults
- Cluster fault codes

# fibreChannelPort

The fibreChannelPort object contains information about individual ports on a node, or for an entire node in the cluster. You can retrieve this information using the `ListNodeFibreChannelPortInfo` method.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| firmware | The version of the firmware installed on the Fibre Channel port. | integer |
| hbaPort | The ID of the individual host bus adapter (HBA) port. | integer |
| model | Model of the HBA on the port. | string |
| nPortID | The unique port node ID. | string |
| pciSlot | The slot containing the PCI card in the Fibre Channel node chassis. | integer |
| serial | The serial number on the Fibre Channel port. | string |
| speed | The speed of the HBA on the port. | string |
| state | Possible values:<br><br>• Unknown<br>• NotPresent<br>• Online<br>• Offline<br>• Blocked<br>• Bypassed<br>• Diagnostics<br>• Linkdown<br>• Error<br>• Loopback<br>• Deleted | string |
| switchWwn | The World Wide Name of the Fibre Channel switch port. | string |
| wwnn | World Wide Node Name of the HBA node. | string |

| Name | Description | Type |
|------|-------------|------|
| wwpn | World Wide Port Name assigned to the physical port of the HBA. | string |

## Find more information

# fipsErrorNodeReport

The fipsErrorNodeReport object contains error information for each node that does not respond with information about FIPS 140-2 support when you query it with the `GetFipsReport` method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| nodeID | The ID of the node that did not respond. | integer |
| error | A JSON object containing error information. | JSON object |

# fipsNodeReport

The fipsNodeReport object contains information about FIPS 140-2 support for a single node in the storage cluster. You can retrieve this information using the `GetFipsReport` method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| nodeID | The ID of the node reporting the information. | integer |

| Name | Description | Type |
|------|-------------|------|
| fipsDrives | Whether or not FIPS 140-2 drive encryption is enabled for this node. Possible values:<br><br>• None: This node is not capable of FIPS drive encryption.<br><br>• Partial: Node is capable of FIPS drive encryption but not all drives present are FIPS-capable drives.<br><br>• Ready: Node is capable of FIPS drive encryption and either all drives present are FIPS-capable drives, or there are no drives present. | FipsDrivesStatusType |
| httpsEnabled | Whether or not FIPS 140-2 HTTPS encryption is enabled for this node. Possible values:<br><br>• true: enabled<br><br>• false: disabled | boolean |

# fipsReport

The fipsReport object contains information about FIPS 140-2 support for all nodes in the storage cluster. You can retrieve this information using the `GetFipsReport` method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| nodes | A report on FIPS 140-2 support status for each node in the storage cluster. | fipsNodeReport |
| errorNodes | Error information for each node that did not respond with FIPS 140-2 support status. | fipsErrorNodeReport |

# groupSnapshot

The groupSnapshot object contains information about a snapshot for a group of volumes. You can use the `ListGroupSnapshots` API method to retrieve group snapshot

information.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| attributes | List of name-value pairs in JSON object format. | JSON object |
| createTime | The UTC+0 formatted day and time on which the group snapshot was created. | ISO 8601 date string |
| enableRemoteReplication | Identifies if the snapshot is enabled for remote replication. | boolean |
| groupSnapshotID | The unique ID of the group snapshot. | integer |
| groupSnapshotUUID | The UUID of the group snapshot. | string |
| members | An array of objects containing information about each member of the group snapshot. | snapshot array |
| name | The name of the group snapshot, or, if none was given, the UTC formatted day and time on which the snapshot was created. | string or ISO 8601 date string |
| remoteStatuses | An array containing the universal identifier and replication status of each remote snapshot on the target cluster as seen from the source cluster. | remoteClusterSnapshotStatus array |

| Name | Description | Type |
|------|-------------|------|
| status | Current status of the snapshot. Possible values:<br><br>• Unknown: There was an error obtaining the status of the snapshot.<br>• Preparing: This snapshot is being prepared for use and is not yet writable.<br>• RemoteSyncing: This snapshot is being replicated from a remote cluster.<br>• Done: This snapshot has finished preparation or replication and is now usable.<br>• Active: This snapshot is the active branch.<br>• Cloning: This snapshot is involved in a CopyVolume operation. | string |

## Find more information

ListGroupSnapshots

# hardwareInfo

The hardwareInfo object contains detailed information about the hardware and status of each node in the cluster. You can retrieve this information with the `GetHardwareInfo` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| boardSerial | The DMI board serial number. | string |
| bus | Motherboard media bus information. | JSON object |
| chassisSerial | The serial number of the chassis. | string |

| Name | Description | Type |
|------|-------------|------|
| driveHardware | A list of information for each drive in the node. | JSON object array |
| fibreChannelPorts | A list of Fibre Channel ports on the node. | integer array |
| hardwareConfig | Motherboard peripheral configuration information. | JSON object |
| kernelCrashDumpState | The crash dump configuration of the operating system kernel. | string |
| memory | Firmware and system memory hardware information. | JSON object |
| network | Descriptions of the hardware of each of the node's network interfaces. | JSON object |
| networkInterfaces | The status of the node's network interfaces. | JSON object |
| nodeSlot | For HCI platforms, the letter corresponding to the chassis slot this node is in ("A", "B", "C", or "D"). For storage platforms, this value is null. | string |
| nvram | NVRAM statistics for the node. | JSON object |
| origin | The vendor of the motherboard. | string |
| platform | A description of the chassis platform. | JSON object |
| serial | The serial number of the product. | string |
| storage | Storage controller information. | JSON object |
| systemMemory | Operating system memory usage and performance information. | JSON object |
| system | The type of node chassis. | JSON object |
| uuid | The unique ID of the node. | UUID |

## Find more information

# host (virtual volumes)

The host object contains information about a virtual volume host. You can use the `ListVirtualVolumeHosts` method to get this information for all virtual volume hosts.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| bindings | A list of objects describing the bindings for the virtual volume host. | integer array |
| clusterID | The unique ID of the cluster this host is associated with. | UUID |
| hostAddress | The IP address or DNS name of the virtual volume host. | string |
| initiatorNames | A list of initiator IQNs for the virtual volume host. | string array |
| virtualVolumeHostID | The unique ID of this virtual volume host. | UUID |
| visibleProtocolEndpointIDs | A list of IDs of protocol endpoints visible on this host. | UUID array |

## Find more information

# idpConfigInfo

The idpConfigInfo object contains configuration and integration details regarding a third-party Identity Provider (IdP).

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| enabled | Specifies whether this third party IdPconfiguration is enabled. | boolean |
| idpConfigurationID | UUID for the third-party IdP configuration. | UUID |
| idpMetadata | Metadata for configuration and integration details for SAML 2.0 single sign-on. | string |
| idpName | Name for retrieving IdP provider for SAML 2.0 single sign-on. | string |
| serviceProviderCertificate | A PEM format Base64 encoded PKCS#10 X.509 certificate to be used for communication with this IdP. | string |
| spMetadataUrl | URL for retrieving Service Provider (SP) Metadata from the Cluster to provide to the IdP for establish a trust relationship. | string |

# initiator

The initiator object contains information about an iSCSI or Fibre Channel initiator. An initiator object can contain IQN or WWPN identifiers. You can use the `ListInitiators` method to get a list of all initiators known on the system. You use initiator objects to configure SCSI initiator access to a set of volumes through volume access groups. An initiator can only be a member of one volume access group at a time. You can restrict initiator access to one or more VLANs by specifying one or more virtualNetworkIDs using the `CreateInitiators` and `ModifyInitiators` methods. If you don't specify any virtual networks, the initiator can access all networks.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| alias | The friendly name assigned to the initiator, if any. | string |

| Name | Description | Type |
|------|-------------|------|
| attributes | A set of JSON attributes assigned to this initiator. Empty if no attributes are assigned. | JSON object |
| chapUsername | The unique CHAP username for this initiator. | string |
| initiatorID | The numeric identifier for the initiator. | integer |
| initiatorName | The initiator name, in IQN or WWPN format. | string |
| initiatorSecret | The CHAP secret used to authenticate the initiator. | string |
| requireChap | True if CHAP is required for this initiator. | boolean |
| targetSecret | The CHAP secret used to authenticate the target (when using mutual CHAP authentication). | string |
| virtualNetworkIDs | The list of virtual network identifiers associated with this initiator. If one or more are defined, this initiator will only be able to login to the specified virtual networks. If no virtual networks are defined this initiator can login to all networks. | integer |
| volumeAccessGroups | A list of volume access group IDs that this initiator belongs to. | integer array |

## Find more information

ListInitiators

# ISCSIAuthentication

The ISCSIAuthentication object contains authentication information about an ISCSI session.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| authMethod | The authentication method used during iSCSI session login, for example, CHAP or None. | string |
| chapAlgorithm | The CHAP algorithm being used, for example, MD5, SHA1*, SHA-256*, or SHA3-256* | string |
| chapUsername | The CHAP username specified by the initiator during an iSCSI session login. | string |
| direction | The authentication direction, for example, one-way (initiator only) or two-way (both initiator and target). | string |

- Available beginning with Element 12.7.

# keyProviderKmip

The keyProviderKmip object describes a Key Management Interoperability Protocol (KMIP) key provider. A key provider is both a mechanism and a location for retrieving authentication keys for use with cluster features such as Encryption at Rest.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| keyProviderID | The ID of the KMIP key provider. This is a unique value assigned by the cluster during key provider creation which cannot be changed. | integer |
| keyProviderIsActive | True if the KMIP key provider is active. A provider is considered active if there are outstanding keys which were created but not yet deleted and therefore assumed to still be in use. | boolean |
| keyProviderName | The name of the KMIP key provider. | string |

| Name | Description | Type |
|---|---|---|
| keyServerIDs | A key server ID that is associated with this provider. The server must be added before this provider can become active. The server cannot be removed while this provider is active. Only one server ID is supported for each provider. | integer array |
| kmipCapabilities | The capabilities of this KMIP key provider including details about the underlying library, FIPS compliance, SSL provider, etc. | string |

# keyServerKmip

The keyServerKmip object describes a Key Management Interoperability Protocol (KMIP) key server, which is a location for retrieving authentication keys for use with cluster features such as Encryption at Rest.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| keyProviderID | If this KMIP key server is assigned to a provider, this member contains the ID of the KMIP key provider it is assigned to. Otherwise this member is null. | integer |
| keyServerID | The ID of the KMIP key server. This is a unique value assigned by the cluster during key server creation. This value cannot be changed. | integer |
| kmipAssignedProviderIsActive | If this KMIP key server is assigned to a provider (keyProviderID is not null), this member indicates whether that provider is active (providing keys which are currently in use). Otherwise, this member is null. | boolean |

| Name | Description | Type |
|------|-------------|------|
| kmipCaCertificate | The public key certificate of the external key server's root CA. This is used to verify the certificate presented by the external key server in the TLS communication. For key server clusters where individual servers use different CAs, this member contains a concatenated string of the root certificates of all the CAs. | string |
| kmipClientCertificate | A PEM format Base64 encoded PKCS#10 X.509 certificate used by the Element storage KMIP client. | string |
| kmipKeyServerHostnames | The hostnames or IP addresses associated with this KMIP key server. | string array |
| kmipKeyServerName | The name of the KMIP key server. This name is only used for display purposes and does not need to be unique. | string |
| kmipKeyServerPort | The port number associated with this KMIP key server (typically 5696). | integer |

# ldapConfiguration

The ldapConfiguration object contains information about the LDAP configuration on the storage system. You can retrieve LDAP information with the `GetLdapConfiguration` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| authType | Identifies which user authentication method to use. Possible values:<br><br>• DirectBind<br><br>• SearchAndBind | string |

| Name | Description | Type |
|------|-------------|------|
| enabled | Identifies whether or not the system is configured for LDAP. Possible values:<br><br>• true<br><br>• false | boolean |
| groupSearchBaseDN | The base DN of the tree to start the group search (the system will perform a subtree search from here). | string |
| groupSearchCustomFilter | The custom search filter used. | string |
| groupSearchType | Controls the default group search filter used. Possible values:<br><br>• NoGroups: No group support.<br><br>• ActiveDirectory: Nested membership of all of a user's AD groups.<br><br>• MemberDN: MemberDN style groups (single-level). | string |
| searchBindDN | A fully qualified DN to log in with to perform an LDAP search for the user (needs read access to the LDAP directory). | string |
| serverURIs | A comma-separated list of LDAP server URIs (for example, `ldap://1.2.3.4` and `ldaps://1.2.3.4:123`.) | string |
| userDNTemplate | A string that is used to form a fully qualified user DN. | string |
| userSearchBaseDN | The base DN of the tree used to start the search (will do a subtree search from here). | string |
| userSearchFilter | The LDAP filter used. | string |

# Find more information

GetLdapConfiguration

# loggingServer

The loggingServer object contains information about any logging hosts configured for the storage cluster. You can use `GetRemoteLoggingHosts` to determine what the current logging hosts are and then use `SetRemoteLoggingHosts` to set the desired list of current and new logging hosts.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| host | IP address of the log server. | string |
| port | Port number used to communicate with the log server. | integer |

# network (bonded interfaces)

The network (bonded interfaces) object contains configuration information for bonded network interfaces on a storage node. You can use the `GetConfig` and `GetNetworkConfig` methods to obtain this information for a storage node.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| address | The IPv4 address assigned to this interface on the node. | string |
| addressV6 | The IPv6 management address assigned to the Bond1G interface on the node. | string |
| bond-downdelay | Time to wait, in milliseconds, before disabling a slave after a link failure has been detected. | string |
| bond-fail_over_mac | The configuration of the MAC address of the network interface. | string |
| bond-miimon | The frequency, in milliseconds, at which the MII link state is inspected for link failures. | string |

| bond-mode | The bonding mode. Possible values:<br><br>• ActivePassive (Default)<br>• ALB<br>• LACP (Recommended) | string |
|---|---|---|
| bond-primary_reselect | Specifies when the primary bond slave is chosen as the active slave. Possible values:<br><br>• Always<br>• Better<br>• Failure | string |
| bond-slaves | The list of slave interfaces for the bond. | string |
| bond-lacp_rate | When Bond Mode is LACP, the rate may change to one of the following:<br><br>• LACP Fast (Default)<br>• LACP Slow | string |
| bond-updelay | The time, in milliseconds, to wait before enabling a slave after a link is detected. | string |
| dns-nameservers | A list of addresses used for domain name services, separated by comma or space. | string |
| dns-search | A space or comma separated list of DNS search domains. | string |
| family | Address family that the interface is configured to use. Currently "inet" for IPv4 is supported. | string |
| gateway | The IPv4 router network address used to send traffic from the local network. | string |
| gatewayV6 | The IPv6 router network address used to send traffic from the local Bond1G network. | string |

| ipV6PrefixLength | The subnet prefix length for static routes of type "net" for IPv6 traffic on the Bond1G network. | string |
|---|---|---|
| macAddress | The actual MAC address assigned to the interface and observed by the network. | string |
| macAddressPermanent | The immutable MAC address assigned by the manufacturer to the interface. | string |
| method | The method used to configure the interface. Possible values:<br><br>• Loopback: Used to define the IPv4 loopback interface.<br><br>• manual: Used to define interfaces that are not configured automatically.<br><br>• dhcp: Can be used to obtain an IP address via DHCP.<br><br>• static: Used to define Ethernet interfaces with statically allocated IPv4 addresses. | string |
| mtu | The largest packet size (in bytes) that the interface can transmit. Must be greater than or equal to 1500; up to 9000 is supported. | string |
| netmask | The bitmask that specifies the subnet for the interface. | string |
| network | Indicates where the IP address range begins based on the netmask. | string |
| routes | Comma separated array of route strings to apply to the routing table. | string array |

| status | The state of the interface. Possible values:<br><br>• Down: The interface is inactive.<br>• Up: The interface is ready, but has no link.<br>• UpAndRunning: The interface is ready and a link is established. | string |
|---|---|---|
| symmetricRouteRules | The symmetric routing rules configured on the node. | string array |
| upAndRunning | Indicates if the interface is ready and has a link. | boolean |
| virtualNetworkTag | The virtual network identifier of the interface (VLAN tag). | string |

## Member modifiability and node states

This table indicates whether or not the object parameters can be modified at each possible node state.

| Member name | Available state | Pending state | Active state |
|---|---|---|---|
| address | Yes | Yes | No |
| addressV6 | Yes | Yes | No |
| bond-downdelay | Configured by the system | N/A | N/A |
| bond-fail_over_mac | Configured by the system | N/A | N/A |
| bond-miimon | Configured by the system | N/A | N/A |
| bond-mode | Yes | Yes | Yes |
| bond-primary_reselect | Configured by the system | N/A | N/A |
| bond-slaves | Configured by the system | N/A | N/A |
| bond-lacp_rate | Yes | Yes | Yes |
| bond-updelay | Configured by the system | N/A | N/A |
| dns-nameservers | Yes | Yes | Yes |

| | | | |
|---|---|---|---|
| dns-search | Yes | Yes | Yes |
| family | No | No | No |
| gateway | Yes | Yes | Yes |
| gatewayV6 | Yes | Yes | Yes |
| ipV6PrefixLength | Yes | Yes | Yes |
| macAddress | Configured by the system | N/A | N/A |
| macAddressPermanent | Configured by the system | N/A | N/A |
| method | No | No | No |
| mtu | Yes | Yes | Yes |
| netmask | Yes | Yes | Yes |
| network | No | No | No |
| routes | Yes | Yes | Yes |
| status | Yes | Yes | Yes |
| symmetricRouteRules | Configured by the system | N/A | N/A |
| upAndRunning | Configured by the system | N/A | N/A |
| virtualNetworkTag | Yes | Yes | Yes |

**Find more information**

- GetConfig
- GetNetworkConfig

# network (all interfaces)

The network (all interfaces) object collects information about network interface configuration for a storage node. You can use the `GetConfig` and `GetNetworkConfig` methods to obtain this information for a storage node.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| Bond10G | Configuration information for the Bond10G bonded interface. | network (bonded interfaces) |
| Bond1G | Configuration information for the Bond1G bonded interface. | network (bonded interfaces) |
| eth0-5 | One object for each Ethernet interface in the storage node, describing configuration information for the interface. These objects are numbered 0 through 5 to match the interface name. | network (Ethernet interfaces) |
| lo | Configuration information for the loopback interface. | network (local interfaces) |

## Find more information

- GetConfig
- GetNetworkConfig

# network (Ethernet interfaces)

The network (Ethernet interfaces) object contains configuration information for individual Ethernet interfaces. You can use the `GetConfig` and `GetNetworkConfig` methods to obtain this information for a storage node.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| bond-master | Specifies which bonded interface this physical interface has joined as a bond slave. | string |
| family | Address family that the interface is configured to use. Currently "inet" for IPv4 is supported. | string |

| | | |
|---|---|---|
| macAddress | The actual MAC address assigned to the interface and observed by the network. | string |
| macAddressPermanent | The immutable MAC address assigned by the manufacturer to the interface. | string |
| method | The method used to configure the interface. Possible values:<br><br>• loopback: Used to define the IPv4 loopback interface.<br>• manual: Used to define interfaces that are not configured automatically.<br>• dhcp: Can be used to obtain an IP address via DHCP.<br>• static: Used to define Ethernet interfaces with statically allocated IPv4 addresses. | string |
| status | The state of the interface. Possible values:<br><br>• Down: The interface is inactive.<br>• Up: The interface is ready, but has no link.<br>• UpAndRunning: The interface is ready and a link is established. | string |
| upAndRunning | Indicates if the interface is ready and has a link. | boolean |

## Member modifiability and node states

This table indicates whether or not the object parameters can be modified at each possible node state.

| Parameter name | Available state | Pending state | Active state |
|---|---|---|---|
| bond-master | No | No | No |
| family | No | No | No |
| macAddress | Configured by system | N/A | N/A |

| macAddressPermanent | Configured by system | N/A | N/A |
|---|---|---|---|
| method | No | No | No |
| status | Yes | Yes | Yes |
| upAndRunning | Configured by system | N/A | N/A |

**Find more information**

- GetConfig
- GetNetworkConfig

# network (local interfaces)

The network (local interfaces) object contains configuration information for local network interfaces, such as the loopback interface, on a storage node. You can use the `GetConfig` and `GetNetworkConfig` methods to obtain this information for a storage node.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| family | Address family that the interface is configured to use. Currently "inet" for IPv4 is supported. | string |
| macAddress | The actual MAC address assigned to the interface and observed by the network. | string |
| macAddressPermanent | The immutable MAC address assigned by the manufacturer to the interface. | string |

| method | The method used to configure the interface. Possible values:<br><br>• loopback: Used to define the IPv4 loopback interface.<br><br>• manual: Used to define interfaces that are not configured automatically.<br><br>• dhcp: Can be used to obtain an IP address via DHCP.<br><br>• static: Used to define Ethernet interfaces with statically allocated IPv4 addresses. | string |
|---|---|---|
| status | The state of the interface. Possible values:<br><br>• Down: The interface is inactive.<br><br>• Up: The interface is ready, but has no link.<br><br>• UpAndRunning: The interface is ready and a link is established. | string |
| upAndRunning | Indicates if the interface is ready and has a link. | boolean |

## Member modifiability and node states

This table indicates whether or not the object parameters can be modified at each possible node state.

| Parameter name | Available state | Pending state | Active state |
|---|---|---|---|
| family | No | No | No |
| macAddress | Configured by system | N/A | N/A |
| macAddressPermanent | Configured by system | N/A | N/A |
| method | No | No | No |
| status | Yes | Yes | Yes |
| upAndRunning | Configured by system | N/A | N/A |

# network (SNMP)

The SNMP network object contains information about SNMP v3 configuration for the cluster nodes.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| access | The type of access allowed for SNMP information requests. Possible values:<br><br>• ro: Read-only access.<br><br>• rw: Read-write access.<br><br>• rosys: Read-only access to a restricted set of system information. | string |
| cidr | A CIDR network mask. This network mask must be an integer greater than or equal to 0, and less than or equal to 32. It must also not be equal to 31. | integer |
| community | The SNMP community string. | string |
| network | This member, along with the cidr member, controls which network the access and community string apply to. The special value of "default" is used to specify an entry that applies to all networks. The CIDR mask is ignored when this member is either a host name or "default". | string |

**Find more information**

GetSnmpInfo

# networkInterface

The networkInterface object contains configuration information for individual network interfaces on a storage node.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| address | The IPv4 management address of the interface. | string |
| addressV6 | The IPv6 management address of the interface. | string |
| broadcast | The broadcast address of the interface. | string |
| macAddress | The MAC address of the interface. | string |
| mtu | The Maximum Transfer Unit, in bytes, of the interface. | integer |
| name | The name of the interface. | string |
| namespace | Whether or not this interface is assigned a virtual network namespace. | boolean |
| netmask | The subnet mask of the interface. | string |
| status | The operational status of the interface. | string |
| type | The type of interface (bond master, bond slave, etc). | string |
| virtualNetworkTag | The VLAN ID assigned to the interface on the virtual network. | integer |

# networkInterfaceStats

The networkInterfaceStats object contains network statistics, the total number of transmitted and received packets, and error information for individual network interfaces on a storage node. You can use the `ListNetworkInterfaceStats` API method to list

this information for the network interfaces on a storage node.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| collisions | The number of collisions detected. | integer |
| name | Name of the network interface. | string |
| rxBytes | The total number of bytes received. | integer |
| rxCrcErrors | The number of received packets that had a CRC error. | integer |
| rxDropped | The number of received packets that were dropped. | integer |
| rxErrors | The number of bad or malformed packets received. | integer |
| rxFifoErrors | The number of FIFO overrun errors in the received data. | integer |
| rxFrameErrors | The number of received packets with frame alignment errors. | integer |
| rxLengthErrors | The number of received packets with a length error. | integer |
| rxMissedErrors | The number of packets missed by the receiver. | integer |
| rxOverErrors | The number of receiver ring buffer overflow errors for this interface. | integer |
| rxPackets | The total number of packets received. | integer |
| txBytes | The total number of bytes transmitted. | integer |
| txCarrierErrors | The number of carrier errors for the transmit side. | integer |
| txErrors | The number of packet transmission errors. | integer |
| txFifoErrors | The number of FIFO overrun errors on the transmit side. | integer |
| txPackets | The total number of packets transmitted. | integer |

# node

The node object contains information about each node in the cluster. You can retrieve this

information using the `ListActiveNodes` and `ListAllNodes` methods.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| associatedFServiceID | The Fibre Channel service ID for the node. "0" if the node is not a Fibre Channel node. | integer |
| associatedMasterServiceID | Master service ID for the node. | integer |
| attributes | List of name-value pairs in JSON object format. | JSON object |
| chassisName | Uniquely identifies a chassis; identical for all nodes in a single chassis. | string |
| cip | The cluster IP address assigned to the node. | string |
| cipi | Network interface used for cluster communication. | string |
| customProtectionDomainName | Uniquely identifies a custom protection domain. This name is identical for all storage nodes within all chassis in a given custom protection domain. | string |
| fibreChannelTargetPortGroup | The target group associated with this node. "null" if the node is not a Fibre Channel node. | integer |
| maintenanceMode | Indicates which mode a node is in for maintenance. | n/a |
| mip | The IP address used for node management. | string |
| mipi | The network interface used for node management. | string |
| name | Host name for the node. | string |

| Name | Description | Type |
|------|-------------|------|
| nodeID | NodeID for this node. | integer |
| nodeSlot | For HCI platforms, the letter corresponding to the chassis slot this node is in ("A", "B", "C", or "D"). For storage platforms, this value is null. | string |
| platformInfo | Hardware information for the node. Members:<br><br>• chassisType: The hardware platform of the node.<br><br>• cpuModel: The CPU model of the hardware platform.<br><br>• nodeMemoryGB: The amount of memory installed in the physical platform in GB.<br><br>• nodeType: The node model name.<br><br>• platformConfigVersion: The version of software configured for this node hardware. | JSON object |
| role | The node's role in the cluster. Possible values:<br><br>• Management<br><br>• Storage<br><br>• Compute<br><br>• Witness | |
| sip | The storage IP address assigned to the node. | string |
| sipi | The network interface used for storage traffic. | string |
| softwareVersion | Returns the current version of Element software running on the node. | string |
| uuid | The universally unique identifier associated with this node. | string |

| Name | Description | Type |
|------|-------------|------|
| virtualNetworks | Object containing virtual network IP addresses and IDs. | virtualNetwork array |

## Find more information

- ListActiveNodes
- ListAllNodes

# nodeProtectionDomains

The nodeProtectionDomains object contains information on the identify of a node and the protection domains associated with that node.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| nodeID | Unique identifier for the node. | integer |
| protectionDomains | List of protection domains of which the node is a member. | protectionDomain |

# nodeStats

The nodeStats object contains high-level activity measurements for a node. You can use the `GetNodeStats` and `ListNodeStats` API methods to get some or all of the nodeStats objects.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| count | The number of total samples in the nodeStats object. | integer |
| cpu | CPU usage, in %. | integer |
| cpuTotal | Monotonically increasing value of cpu utilization. | integer |

| Name | Description | Type |
|---|---|---|
| cBytesIn | Bytes in on the cluster interface. | integer |
| cBytesOut | Bytes out on the cluster interface. | integer |
| sBytesIn | Bytes in on the storage interface. | integer |
| sBytesOut | Bytes out on the storage interface. | integer |
| mBytesIn | Bytes in on the management interface. | integer |
| mBytesOut | Bytes out on the management interface. | integer |
| networkUtilizationCluster | Network interface utilization (in %) for the cluster network interface. | integer |
| networkUtilizationStorage | Network interface utilization (in %) for the storage network interface. | integer |
| readLatencyUSecTotal | Monotonically increasing value of total time spent performing read operations to the node. | integer |
| readOps | Monotonically increasing value of total read operations to a node. | integer |
| ssLoadHistogram | Histogram data illustrating slice service load over time. | JSON object |
| timestamp | The current time in UTC+0 format. | ISO 8601 date string |
| usedMemory | Total memory usage in bytes. | integer |
| writeLatencyUSecTotal | Monotonically increasing value of total time spent performing write operations to the node. | integer |
| writeOps | Monotonically increasing value of total write operations to a node. | integer |

## Find more information

- GetNodeStats
- ListNodeStats

# ontapVersionInfo

The ontapVersionInfo object contains information about the API version of the ONTAP cluster in a SnapMirror relationship. The Element web UI uses the `GetOntapVersionInfo` API method to get this information.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| clientAPIMajorVesion | The ONTAP API major version in use by the Element API client. | string |
| clientAPIMinorVesion | The ONTAP API minor version in use by the Element API client. | string |
| ontapAPIMajorVersion | The current API major version supported by the ONTAP system. | string |
| ontapAPIMinorVesion | The current API minor version supported by the ONTAP system. | string |
| ontapVersion | The current software version running on the ONTAP cluster. | string |

# pendingActiveNode

The pendingActiveNode object contains information about a node that is currently in the pendingActive state, between the pending and active states. These are nodes that are currently being returned to the factory software image. Use the `ListPendingActiveNodes` API method to return a list of this information for all pendingActive nodes.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| activeNodeKey | A unique key that allows the node to join the cluster automatically after a successful installation of software. | string |
| assignedNodeID | The assigned node ID for the node. | string |
| asyncHandle | The asynchronous method handle that you can use to query the status of the operation. | integer |
| cip | The cluster IP address assigned to the node. | string |
| mip | The management IP address assigned to the node. | string |
| nodeSlot | For HCI platforms, the letter corresponding to the chassis slot this node is in ("A", "B", "C", or "D"). For storage platforms, this value is null. | string |
| pendingActiveNodeID | The pending node ID of the node. | integer |
| platformInfo | Hardware information for the node. Members:<br><br>• chassisType: The hardware platform of the node.<br>• cpuModel: The CPU model of the hardware platform.<br>• nodeMemoryGB: The amount of memory installed in the physical platform in GB.<br>• nodeType: The node model name.<br>• platformConfigVersion: The version of software configured for this node hardware. | JSON object |

| Name | Description | Type |
|------|-------------|------|
| role | The node's role in the cluster. Possible values:<br><br>&bull; Management<br><br>&bull; Storage<br><br>&bull; Compute<br><br>&bull; Witness | |
| sip | The storage (iSCSI) IP address assigned to the node. | string |
| softwareVersion | The current version of Element software running on the node. | string |

## Find more information

# pendingNode

The pendingNode object contains information about a node that can be added to a cluster. Use the `ListPendingNodes` API method to return a list of this information for all pending nodes. You can add any of the listed nodes to a cluster using the `AddNodes` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| cipi | The cluster IP address assigned to the node. | string |
| activeNodeKey | A unique key that allows the node to join the cluster automatically after a successful installation of software. | string |
| assignedNodeID | The assigned node ID for the node. | string |
| asyncHandle | The asynchronous method handle that you can use to query the status of the operation. | integer |

| Name | Description | Type |
|---|---|---|
| chassisName | Uniquely identifies a chassis; identical for all nodes in a single chassis. | string |
| cip | The cluster IP address assigned to the node. | string |
| mip | The management IP address assigned to the node. | string |
| nodeSlot | For HCI platforms, the letter corresponding to the chassis slot this node is in ("A", "B", "C", or "D"). For storage platforms, this value is null. | string |
| pendingActiveNodeID | The pending node ID of the node. | integer |
| platformInfo | Hardware information for the node. Members:<br><br>• chassisType: The hardware platform of the node.<br><br>• cpuModel: The CPU model of the hardware platform.<br><br>• nodeMemoryGB: The amount of memory installed in the physical platform in GB.<br><br>• nodeType: The node model name.<br><br>• platformConfigVersion: The version of software configured for this node hardware. | JSON object |
| role | The node's role in the cluster. Possible values:<br><br>• Management<br><br>• Storage<br><br>• Compute<br><br>• Witness | |
| sip | The storage (iSCSI) IP address assigned to the node. | string |

| Name | Description | Type |
|------|-------------|------|
| softwareVersion | The current version of Element software running on the node. | string |

## Find more information

- [AddNodes](#)
- [ListPendingNodes](#)

# protectionDomain

The protectionDomain object contains the name and type details for a protection domain.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| protectionDomainName | The name of the protection domain. | string |
| protectionDomainType | The type of the protection domain. Possible values:<br><br>• chassis: All storage nodes in a single chassis.<br><br>• custom: All storage nodes in a single customer-defined protection domain. | string |

# protectionDomainLevel

The protectionDomainLevel object contains information about the storage cluster's current tolerance and resiliency levels. Tolerance levels indicate the cluster's ability to continue reading and writing data in the event of a failure, and resiliency levels indicate the cluster's ability to automatically heal itself from one or more failures within its associated type of protection domain.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| protectionDomainType | The type of the protection domain which has the associated tolerance and resiliency. Possible values:<br><br>• node: Any individual node.<br><br>• chassis: Any individual node or all storage nodes in a single chassis.<br><br>• custom: All storage nodes in a single customer-defined protection domain. | string |
| resiliency | The current resiliency of this cluster from the perspective of this protection domain type. | protectionDomainResiliency |
| tolerance | The current tolerance of this cluster from the perspective of this protection domain type. | protectionDomainTolerance |

# protectionDomainResiliency

The protectionDomainResiliency object contains the resiliency status of this storage cluster. Resiliency indicates the storage cluster's ability to automatically heal itself from one or more failures all within a single protection domain of its associated protection domain type. A storage cluster is considered healed when it can continue reading and writing data through the failure of any single storage node (a state known as node tolerance).

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| protectionSchemeResiliencies | A list of objects (one for each protection scheme) containing failure resiliency information for the associated type of protection domain. | protectionSchemeResiliency array |
| singleFailureThresholdBytesForBlockData | The maximum number of bytes that can be stored on the storage cluster before losing the ability to automatically heal to a state of node tolerance. | integer |

| Name | Description | Type |
|---|---|---|
| sustainableFailuresForEnsemble | The predicted number of simultaneous failures that can occur without losing the ability to automatically heal to a state of node tolerance for the ensemble quorum. | integer |

# protectionDomainTolerance

The protectionDomainTolerance object contains information about the ability of the storage cluster to continue reading and writing data in the event of one or more failures all within a single protection domain of its associated protection domain type.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| protectionSchemeTolerances | A list of objects (one for each protection scheme) containing failure tolerance information for the associated type of protection domain. | protectionSchemeTolerance array |
| sustainableFailuresForEnsemble | The number of simultaneous failures within the associated type of protection domain that can occur without losing the ensemble quorum. | integer |

# protectionSchemeResiliency

The protectionSchemeResiliency object contains information about whether a storage cluster, for a specific protection scheme, can automatically heal itself from one or more failures within its associated protectionDomainType. A storage cluster is considered healed when it can continue reading and writing data through the failure of any single storage node (a state known as node tolerance).

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| protectionScheme | The current protection scheme of this storage cluster. The only possible value is doubleHelix. | string |
| sustainableFailuresForBlockData | The predicted number of simultaneous failures which can occur without losing the ability to automatically heal to a state of node tolerance for data. | integer |
| sustainableFailuresForMetadata | The predicted number of simultaneous failures which can occur without losing the ability to automatically heal to a state of node tolerance for metadata. | integer |

# protectionSchemeTolerance

The protectionSchemeTolerance object contains information about whether a storage cluster, for a specific protection scheme, can continue to read and write data after failures.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| protectionScheme | The current protection scheme of this storage cluster. The only possible value is doubleHelix. | string |
| sustainableFailuresForBlockData | The current number of simultaneous failures which can occur without losing block data availability for the associated protection scheme. | integer |
| sustainableFailuresForMetadata | The current number of simultaneous failures which can occur without losing metadata availability for the associated protection scheme. | integer |

# protocolEndpoint

The protocolEndpoint object contains the attributes of a protocol endpoint. You can retrieve this information for all protocol endpoints in the cluster using the `ListProtocolEndpoints` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| primaryProviderID | The ID of the primary protocol endpoint provider object for the protocol endpoint. | integer |
| protocolEndpointID | The unique ID of the protocol endpoint. | UUID |
| protocolEndpointState | The status of the protocol endpoint. Possible values:<br><br>• Active: The protocol endpoint is in use.<br><br>• Start: The protocol endpoint is starting.<br><br>• Failover: The protocol endpoint has failed over.<br><br>• Reserved: The protocol endpoint is reserved. | string |
| providerType | The type of the protocol endpoint's provider. Possible values:<br><br>• Primary<br><br>• Secondary | string |
| scsiNAADeviceID | The globally unique SCSI device identifier for the protocol endpoint in NAA IEEE Registered Extended Format. | string |
| secondaryProviderID | The ID of the secondary protocol endpoint provider object for the protocol endpoint. | integer |

**Find more information**

# QoS

The QoS object contains information about Quality of Service (QoS) settings for volumes. Volumes created without specified QoS values are created using the default values. You can find default values using the `GetDefaultQoS` method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| burstIOPS | Maximum "peak" 4KB IOPS allowed for short periods of time. Allows for bursts of I/O activity over the normal maxIOPS value. | integer |
| burstTime | The length of time burstIOPS is allowed. The value returned is represented in seconds. This value is calculated by the system based on IOPS set for QoS. | integer |
| curve | The curve is a set of key-value pairs. The keys are I/O sizes in bytes. The values represent the cost of performing one IOP at a specific I/O size. The curve is calculated relative to a 4096 byte operation set at 100 IOPS. | JSON object |
| maxIOPS | The desired maximum 4KB IOPS allowed over an extended period of time. | integer |
| minIOPS | The desired minimum 4KB IOPS to guarantee. The allowed IOPS will only drop below this level if all volumes have been capped at their minIOPS value and there is still insufficient performance capacity. | integer |

## Find more information

# QoSPolicy

The QoSPolicy object contains information about a QoS policy on a storage cluster running Element software.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| qosPolicyID | A unique integer identifier for the QoSPolicy automatically assigned by the storage cluster. | integer |
| name | The name of the QoS policy. For example: gold, platinum, or silver. | string |
| qos | The QoS settings that this policy represents. | QoS |
| volumeIDs | A list of volumes associated with this policy. | integer array |

## Find more information

GetQoSPolicy

# remoteClusterSnapshotStatus

The `remoteClusterSnapshotStatus` object contains the UUID and status of a snapshot stored on a remote storage cluster. You can get this information with the `ListSnapshots` or `ListGroupSnapshots` API methods.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| remoteStatus | The replication status of the remote snapshot on the target cluster as seen from the source cluster. Possible values:<br><br>• Present: The snapshot exists on a remote cluster.<br>• NotPresent: The snapshot does not exist on a remote cluster.<br>• Syncing: This is a target cluster and it is currently replicating the snapshot.<br>• Deleted: This is a target cluster. The snapshot has been deleted, and it still exists on the source. | string |
| volumePairUUID | The universal identifier of the volume pair. | UUID |

# schedule

The schedule object contains information about a schedule created to autonomously make a snapshot of a volume. You can retrieve schedule information for all schedules with the `ListSchedules` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| attributes | Indicates the frequency of the schedule occurrence. Possible values:<br><br>• Day of Week<br>• Day of Month<br>• Time Interval | JSON object |

| Name | Description | Type |
|------|-------------|------|
| hasError | Indicates whether or not the schedule has errors. Possible values:<br><br>• true<br><br>• false | boolean |
| hours | Shows the hours that will elapse before the next snapshot is created. Possible values are 0 through 24. | integer |
| lastRunStatus | Indicates the status of the last scheduled snapshot. Possible values:<br><br>• Success<br><br>• Failed | string |
| lastRunTimeStart | Indicates the last time the schedule started. | ISO 8601 date string |
| minutes | Shows the minutes that will elapse before the next snapshot is created. Possible values are 0 through 59. | integer |
| monthdays | Indicates the days of the month that a snapshot will be made. | array |
| paused | Indicates whether or not the schedule is paused. Possible values:<br><br>• true<br><br>• false | boolean |
| recurring | Indicates whether or not the schedule is recurring. Possible values:<br><br>• true<br><br>• false | boolean |

| Name | Description | Type |
|------|-------------|------|
| runNextInterval | Indicates whether or not the schedule will run the next time the scheduler is active. When true, the schedule will run the next time the scheduler is active and then this value is set back to false. Possible values:<br><br>    • true<br><br>    • false | boolean |
| scheduleID | The unique ID of the schedule. | integer |

| Name | Description | Type |
|------|-------------|------|
| scheduleInfo | Includes the unique name given to the schedule, the retention period for the snapshot that was created, and the volume ID of the volume from which the snapshot was created. Valid values:<br><br>• `enableRemoteReplication`: Indicates if the snapshot should be included in remote replication. (boolean)<br><br>• `ensureSerialCreation`: Specifies whether a new snapshot creation should be allowed if a previous snapshot replication is in progress. (boolean)<br><br>• `name`: The snapshot name to be used. (string)<br><br>• `retention`: The amount of time the snapshot is retained. Depending on the time, it displays in one of the following formats:<br><br>  ◦ `fifo`: The snapshot is retained on a First-In-First-Out (FIFO) basis. If empty, the snapshot is retained forever. (string)<br><br>  ◦ "HH:mm:ss"<br><br>• `volumeID`: The ID of the volume to be included in the snapshot. (integer)<br><br>• `volumes`: A list of volume IDs to be included in the group snapshot. (integer array) | JSON object |
| scheduleName | The unique name assigned to the schedule. | string |
| scheduleType | Only schedule types of snapshot are supported at this time. | string |

| Name | Description | Type |
|---|---|---|
| snapMirrorLabel | The snapMirrorLabel to be applied to the created Snapshot or Group Snapshot, contained in the scheduleInfo. If not set, this value is null. | string |
| startingDate | Indicates the date the first time the schedule began or will begin; formatted in UTC time. | ISO 8601 date string |
| toBeDeleted | Indicates if the schedule is marked for deletion. Possible values:<br><br>• true<br><br>• false | boolean |
| weekdays | Indicates the days of the week that a snapshot will be made. | array |

### Find more information

[ListSchedules](#)

## session (Fibre Channel)

The session object contains information about each Fibre Channel session that is visible to the cluster and what target ports it is visible on. You can retrieve this information with the `ListFibreChannelSessions` API method.

### Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| initiatorWWPN | The World Wide Port Name (WWPN) of the initiator that is logged into the target port. | string |
| nodeID | The node that owns the Fibre Channel session. | integer |

| Name | Description | Type |
|---|---|---|
| initiator | Information about this Fibre Channel session's server initiator. Members:<br><br>• alias: The friendly name assigned to the initiator.<br><br>• attributes: The attributes of this initiator.<br><br>• initiatorID: The ID of this initiator.<br><br>• initiatorName: The name of this initiator.<br><br>• volumeAccessGroups: A list of volume access groups associated with this initiator. | JSON object |
| serviceID | The service ID of the target port involved in this session. | integer |
| targetWWPN | The WWPN of the target port involved in this session. | string |
| volumeAccessGroupID | The ID of the volume access group to which the initiatorWWPN belongs. If not in a volume access group, this value is null. | integer |

## Find more information

[ListFibreChannelSessions](#)

# session (iSCSI)

The session (iSCSI) object contains detailed information about each volume's iSCSI session. You can retrieve iSCSI session information with the `ListISCSISessions` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| accountID | The account ID of the account used for CHAP authentication, if any. | integer |

| Name | Description | Type |
|------|-------------|------|
| accountName | The name of the account used for CHAP authentication, if any. | string |
| authentication | Authentication information for this iSCSI session. | ISCSIAuthentication |
| createTime | The time of the creation of the iSCSI session, in UTC+0 format. | ISO 8601 date string |
| driveID | The driveID associated with the transport service hosting the session. | integer |
| driveIDs | A list of the driveIDs of the drives reporting the failure. An empty list if not applicable. | integer array |
| initiator | Information about this iSCSI session's server initiator. Members:<br><br>• alias: The friendly name assigned to the initiator.<br><br>• attributes: The attributes of this initiator.<br><br>• initiatorID: The ID of this initiator.<br><br>• initiatorName: The name of this initiator.<br><br>• volumeAccessGroups: A list of volume access groups associated with this initiator. | JSON object |
| initiatorIP | The IP address and port number of the iSCSI server initiator. | string |
| initiatorName | The iSCSI Qualified Name (IQN) of the iSCSI server initiator. | string |
| initiatorPortName | The initiatorName combined with the initiatorSessionID; identifies the initiator port. | string |
| initiatorSessionID | A 48-bit ID provided by the initiator that identifies the iSCSI session as belonging to that initiator. | integer |

| Name | Description | Type |
|---|---|---|
| msSinceLastIscsiPDU | The time, in milliseconds, since the last iSCSI PDU was received for this session. | integer |
| msSinceLastScsiCommand | The time, in milliseconds, since the last SCSI command was received for this session. | integer |
| nodeID | The nodeID associated with the transport service hosting the session. | integer |
| serviceID | The serviceID of the transport service hosting the session. | integer |
| sessionID | The iSCSI session ID. | integer |
| targetIP | The IP address and port number of the iSCSI storage target. | string |
| targetName | The IQN of the iSCSI target. | string |
| targetPortName | The targetName combined with the target portal group tag; identifies the target port. | string |
| virtualNetworkID | The virtual network ID associated with the session. | integer |
| volumeID | The volumeID of the volume associated with the session, if any. | integer |
| volumeInstance | Identifies the volume object associated with the iSCSI session, if any. | integer |

## Find more information

ListISCSISessions

# snapMirrorAggregate

The snapMirrorAggregate object contains information about the available ONTAP aggregates, which are collections of disks made available to volumes as storage. You can get this information using the ListSnapMirrorAggregates API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| aggregateName | The name of the aggregate. | string |
| nodeName | The name of the ONTAP node that owns this aggregate. | string |
| sizeAvailable | The number of available bytes remaining in the aggregate. | integer |
| sizeTotal | The total size (in bytes) of the aggregate. | integer |
| percentUsedCapacity | The percentage of disk space currently in use. | integer |
| volumeCount | The number of volumes in the aggregate. | integer |

# snapMirrorClusterIdentity

The snapMirrorClusterIdentity object contains identification information about the remote ONTAP cluster in a SnapMirror relationship.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| clusterName | The name of the destination ONTAP cluster. | string |
| clusterUUID | The 128-bit universally-unique identifier of the destination ONTAP cluster. | string |

| Name | Description | Type |
|------|-------------|------|
| clusterSerialNumber | The serial number of the destination ONTAP cluster. | string |

# snapMirrorEndpoint

The snapMirrorEndpoint object contains information about the remote SnapMirror storage systems communicating with the Element storage cluster. You can retrieve this information with the ListSnapMirrorEndpoints API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| snapMirrorEndpointID | The unique identifier for the object in the local cluster. | integer |
| managementIP | The cluster management IP address of the endpoint. | string |
| clusterName | The ONTAP cluster name. This value is automatically populated with the value of "clusterName" from the snapMirrorClusterIdentity object. | string |
| username | The management user name for the ONTAP system. | string |
| ipAddresses | List of the inter-cluster storage IP addresses for all nodes in the cluster. You can get these IP addresses with the ListSnapMirrorNetworkInterfaces method. | string array |
| isConnected | The connectivity status of the control link to the ONTAP cluster. | boolean |

# snapMirrorJobScheduleCronInfo

The snapMirrorJobScheduleCronInfo object contains information about a cron job schedule on the ONTAP system.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| jobScheduleName | The name of the job schedule. | string |
| jobScheduleDescription | An automatically-generated human-readable summary of the schedule. | string |

# snapMirrorLunInfo

The snapMirrorLunInfo object contains information about the ONTAP LUN object.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| creationTimestamp | The creation time of the LUN. | ISO 8601 date string |
| lunName | The name of the LUN. | string |
| path | The path of the LUN. | string |
| size | The size of the LUN in bytes. | integer |
| sizeUsed | The number of bytes used by the LUN. | integer |

| Name | Description | Type |
|---|---|---|
| state | The current access state of the LUN. Possible values:<br><br>• online<br><br>• offline<br><br>• foreign_lun_error<br><br>• nvfail<br><br>• space_error | string |
| volume | The name of the volume that contains the LUN. | string |
| vserver | The Vserver that contains the LUN. | string |

# snapMirrorNetworkInterface

The snapMirrorNetworkInterface object contains information about the intercluster Logical Interfaces (LIFs).

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| administrativeStatus | Whether the logical interface (LIF) is administratively enabled or disabled. Possible values:<br><br>• up<br><br>• down | string |
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| interfaceName | The LIF name. | string |
| networkAddress | The IP address of the LIF. | string |
| networkMask | The network mask of the LIF. | string |

| Name | Description | Type |
|---|---|---|
| interfaceRole | The role of the LIF. Possible values:<br><br>• undef<br>• cluster<br>• data<br>• node_mgmt<br>• intercluster<br>• cluster_mgmt | string |
| operationalStatus | The operational state of the LIF (whether or not it has formed a successful connection). This status can differ from the administrative status if there is a network problem that prevents the interface from functioning. Possible values:<br><br>• up<br>• down | string |
| vserverName | The name of the Vserver. | string |

# snapMirrorNode

The snapMirrorNode object contains information about the nodes of the destination ONTAP cluster in a SnapMirror relationship.

## Object members

This object contains the following members:

| Name | Description | Type |
|---|---|---|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| name | The name of the ONTAP node. | string |
| model | The model of the ONTAP node. | string |
| serialNumber | The serial number of the ONTAP node. | string |

| Name | Description | Type |
|------|-------------|------|
| productVersion | The ONTAP product version. | string |
| isNodeHealthy | The health of a node in the ONTAP cluster. Possible values:<br><br>• true<br><br>• false | string |
| isNodeEligible | Whether or not the node is eligible to participate in an ONTAP cluster. Possible values:<br><br>• true<br><br>• false | string |

# snapMirrorPolicy

The snapMirrorPolicy object contains information about a SnapMirror policy that is stored on an ONTAP system.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| policyName | The unique name assigned to the policy. | string |
| policyType | The type of policy. Possible values:<br><br>• async_mirror<br><br>• mirror_vault | string |
| comment | A human-readable description associated with the SnapMirror policy. | string |

| Name | Description | Type |
|------|-------------|------|
| transferPriority | The priority at which a SnapMirror transfer runs. Possible values:<br><br>• normal: The default priority. These transfers are scheduled before most low priority transfers.<br><br>• low: These transfers have the lowest priority and are scheduled after most normal priority transfers. | string |
| policyRules | A list of objects describing the policy rules. | snapMirrorPolicyRule array |
| totalKeepCount | The total retention count for all rules in the policy. | integer |
| totalRules | The total number of rules in the policy. | integer |
| vserverName | The name of the Vserver for the SnapMirror policy. | string |

# snapMirrorPolicyRule

The snapMirrorPolicyRule object contains information about the rules in a SnapMirror policy.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| snapMirrorLabel | The snapshot copy label, used for snapshot copy selection in extended data protection relationships. | string |
| keepCount | Specifies the maximum number of snapshot copies that are retained on the SnapMirror destination volume for a rule. | integer |

# snapMirrorRelationship

The snapMirrorRelationship object contains information about a SnapMirror relationship between a Element volume and an ONTAP volume.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| snapMirrorRelationshipID | The unique identifier for each snapMirrorRelationship object in an array as would be returned in ListSnapMirrorRelationships. This UUID is created and returned from the ONTAP system. | string |
| sourceVolume | An object describing the source volume. | snapMirrorVolumeInfo |
| destinationVolume | An object describing the destination volume. | snapMirrorVolumeInfo |
| currentMaxTransferRate | The current maximum transfer rate between the source and destination volumes, in kilobytes per second. | integer |
| isHealthy | Whether the relationship is healthy or not. Possible values:<br><br>• true: The relationship is healthy.<br><br>• false: The relationship is not healthy. This can be caused by a manual or scheduled update failing or being aborted, or by the last scheduled update being delayed. | boolean |
| lagtime | The amount of time in seconds by which the data on the destination volume lags behind the data on the source volume. | integer |

| Name | Description | Type |
|------|-------------|------|
| lastTransferDuration | The amount of time in seconds it took for the last transfer to complete. | integer |
| lastTransferError | A message describing the cause of the last transfer failure. | string |
| lastTransferSize | The total number of bytes transferred during the last transfer. | integer |
| lastTransferEndTimestamp | The timestamp of the end of the last transfer. | ISO 8601 date string |
| lastTransferType | The type of the previous transfer in the relationship. | string |
| maxTransferRate | Specifies the maximum data transfer rate between the volumes in kilobytes per second. The default value, 0, is unlimited and permits the SnapMirror relationship to fully utilize the available network bandwidth. | integer |
| mirrorState | The mirror state of the SnapMirror relationship. Possible values:<br><br>• uninitialized: The destination volume has not been initialized.<br><br>• snapmirrored: The destination volume has been initialized and is ready to receive SnapMirror updates.<br><br>• broken-off: The destination volume is read-write and snapshots are present. | string |
| newestSnapshot | The name of the newest Snapshot copy on the destination volume. | string |
| policyName | Specifies the name of the ONTAP SnapMirror policy for the relationship. A list of available policies can be retrieved with ListSnapMirrorPolicies. Example values are "MirrorLatest" and "MirrorAndVault". | string |

| Name | Description | Type |
|------|-------------|------|
| policyType | The type of the ONTAP SnapMirror policy for the relationship. See ListSnapMirrorPolicies. Examples are: "async_mirror" or "mirror_vault". | string |
| relationshipProgress | 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 "relationshipStatus" member indicates that an activity is in progress. | integer |
| relationshipStatus | The status of the SnapMirror relationship. Possible values:<br><br>• idle<br>• transferring<br>• checking<br>• quiescing<br>• quiesced<br>• queued<br>• preparing<br>• finalizing<br>• aborting<br>• breaking | string |
| relationshipType | The type of the SnapMirror relationship. On storage clusters running Element software, this value is always "extended_data_protection". | string |
| scheduleName | The name of the pre-existing cron schedule on the ONTAP system that is used to update the SnapMirror relationship. A list of available schedules can be retrieved with ListSnapMirrorSchedules. | string |
| unhealthyReason | The reason the relationship is not healthy. | string |

# snapMirrorVolume

The snapMirrorVolume object contains information about an ONTAP volume.

**Object members**

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| name | The name of the volume. | string |
| type | The type of volume. Possible values:<br><br>• rw: Read-write volume<br>• ls: Load-sharing volume<br>• dp: Data protection volume | string |
| vserver | The name of the Vserver that owns this volume. | string |
| aggrName | The containing aggregate name. | string |
| state | The state of volume. Possible values:<br><br>• online<br>• restricted<br>• offline<br>• mixed | string |
| size | The total filesystem size (in bytes) of the volume. | string |
| availSize | The size (in bytes) of the available space in the volume. | string |

# snapMirrorVolumeInfo

The snapMirrorVolumeInfo object contains information about a volume location in a SnapMirror relationship, such as its name and type.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| type | The type of volume. Possible values:<br><br>• solidfire: The volume resides on a storage cluster running Element software.<br><br>• ontap: The volume resides on a remote ONTAP cluster. | string |
| volumeID | The ID of the volume. Only valid if "type" is solidfire. | integer |
| vserver | The name of the Vserver that owns this volume. Only valid if "type" is ontap. | string |
| name | The name of the volume. | string |

# snapMirrorVserver

The snapMirrorVserver object contains information about the Storage Virtual Machines (or Vservers) at the destination ONTAP cluster.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| snapMirrorEndpointID | The ID of the destination ONTAP system. | integer |
| vserverName | The name of the Vserver. | string |
| vserverType | The type of Vserver. Possible values:<br><br>• data<br><br>• admin<br><br>• system<br><br>• node | string |

| Name | Description | Type |
|---|---|---|
| vserverSubtype | The subtype of the Vserver. Possible values:<br><br>• default<br>• dp_destination<br>• data<br>• sync_source<br>• sync_destination | string |
| rootVolume | The root volume of the Vserver. | string |
| rootVolumeAggregate | The aggregate on which the root volume will be created. | string |
| vserverAggregateInfo | An array of snapMirrorVserverAggregateInfo objects. | JSON object |
| adminState | The detailed administrative state of the Vserver. Possible values:<br><br>• running<br>• stopped<br>• starting<br>• stopping<br>• initializing<br>• deleting | string |
| operationalState | The basic operational state of the Vserver. Possible values:<br><br>• running<br>• stopped | string |

# snapMirrorVserverAggregateInfo

The snapMirrorVserverAggregateInfo object contains information about the available data Storage Virtual Machines (also called Vservers) at the destination ONTAP cluster.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| aggrName | The name of the aggregate assigned to a Vserver. | string |
| aggrAvailSize | The assigned aggregate's available size. | integer |

# snapshot

The snapshot object contains information about a snapshot made for a volume. You can use the `ListSnapshots` API method to retrieve a list of snapshot information for a volume or for all volumes. The object includes information about the active snapshot as well as each snapshot created for a volume.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| attributes | List of name-value pairs in JSON object format. | JSON object |
| checksum | A small string representation of the data in the stored snapshot. This checksum can be used later to compare other snapshots to detect errors in the data. | string |
| createTime | The UTC+0 formatted time the snapshot was created. | ISO 8601 date string |
| enableRemoteReplication | Identifies if snapshot is enabled for remote replication. | boolean |
| expirationReason | Indicates how the snapshot expiration is set. Possible values:<br><br>• Api: The expiration time is set by using the API.<br><br>• None: No expiration time is set.<br><br>• Test: The expiration time is set for testing.<br><br>• fifo: Expiration occurs on a first-in-first-out basis. | string |

| Name | Description | Type |
|------|-------------|------|
| expirationTime | The time at which this snapshot will expire and be purged from the cluster. | ISO 8601 date string |
| groupID | The group ID if the snapshot is a member of a group snapshot. | integer |
| groupsnapshotUUID | Contains information about each snapshot in the group. Each of these members will have a UUID parameter for the snapshot's UUID. | string |
| instanceCreateTime | The time that the snapshot was created on the local cluster. | ISO 8601 date string |
| instanceSnapshotUUID | The universally unique ID of the snapshot on the local cluster. This ID does not get replicated to other clusters. | string |
| name | The unique name assigned to the snapshot. If no name is specified, the name is the UTC+0 formatted timestamp of when the snapshot was created. | string |
| remoteStatuses | An array containing the universal identifier and replication status of each remote snapshot on the target cluster as seen from the source cluster. | remoteClusterSnapshotStatus array |
| snapMirrorLabel | The label used by SnapMirror software to specify snapshot retention policy on SnapMirror endpoints. If not set, this value is null. | string |
| snapshotID | The unique ID of an existing snapshot. | string |
| snapshotUUID | The universally unique ID of an existing snapshot. When the snapshot is replicated across clusters, this ID is replicated along with it and is used to identify the snapshot across clusters. | string |

| Name | Description | Type |
|------|-------------|------|
| status | Current status of the snapshot. Possible values:<br><br>• Unknown: There was an error obtaining the status of the snapshot.<br><br>• Preparing: This snapshot is being prepared for use and is not yet writable.<br><br>• RemoteSyncing: This snapshot is being replicated from a remote cluster.<br><br>• Done: This snapshot has finished preparation or replication and is now usable.<br><br>• Active: This snapshot is the active branch.<br><br>• Cloning: This snapshot is involved in a CopyVolume operation. | string |
| totalSize | The total size in bytes of the snapshot. | integer |
| virtualVolumeID | The ID of the virtual volume associated with this snapshot. | UUID |
| volumeID | The ID of the volume the snapshot was created from. | integer |
| volumeName | The name of the volume at the time the snapshot was created. | string |

## Find more information

# snmpTrapRecipient

The snmpTrapRecipient object contains information about a host that is configured to receive SNMP traps generated by the storage cluster. You can use the `GetSnmpTrapInfo` API method to get a list of hosts configured to receive SNMP traps.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| host | The IP address or host name of the target host. | string |
| port | The UDP port number on the host where the trap should be sent. Valid range is 1 through 65535. 0 (zero) is not a valid port number. The default port is 162. | integer |
| community | SNMP community string. | string |

# storageContainer

The storageContainer object contains the attributes of a virtual volume storage container. You can retrieve this information for each storage container in the cluster using the `ListStorageContainers` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| accountID | The ID of the storage system account associated with the storage container. | integer |
| initiatorSecret | The CHAP authentication secret for the initiator associated with the storage container. | string |
| name | The name of the storage container. | string |
| protocolEndpointType | The storage container's protocol endpoint type. SCSI is the only valid value. | string |

| Name | Description | Type |
|------|-------------|------|
| status | The status of the storage container. Possible values:<br><br>• Active: The storage container is in use.<br><br>• Locked: The storage container is locked. | string |
| storageContainerID | The unique ID of the storage container. | UUID |
| targetSecret | The CHAP authentication secret for the target associated with the storage container. | string |
| virtualVolumes | A list of IDs of the virtual volumes associated with the storage container. | UUID array |

## Find more information

ListStorageContainers

# syncJob

The syncJob object contains information about clone, remote replication, or slice synchronization jobs that are running on a cluster.

You can retrieve synchronization information with the `ListSyncJobs` API method.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| blocksPerSecond | The number of data blocks being transferred per second from the source cluster to the target cluster. Present only if the type member is set to remote. | integer |

| Name | Description | Type |
|------|-------------|------|
| branchType | Returned for remote replication sync jobs only. Possible values:<br><br>• snapshot<br><br>• volume | string |
| bytesPerSecond | The number of bytes the clone is processing per second. Present only if the type member is set to clone or slice. | float |
| cloneID | The identifier of the clone operation that is in progress. Present only if the type member is set to clone. | integer |
| currentBytes | The number of bytes the clone has processed in the source volume. Present only if the type member is set to clone or slice. | integer |
| dstServiceID | The service identifier hosting the primary replica for the volume. Present only if the type member is set to remote. | integer |
| dstVolumeID | The destination volume ID. Present only if the type member is set to clone or remote. | integer |
| elapsedTime | The time elapsed, in seconds, since the sync job started. | float or integer depending on the type of sync operation |
| groupCloneID | The ID of the group clone operation that is in progress. | integer |
| nodeID | Specifies the node the clone is occurring on. Present only if the type member is set to clone. | integer |
| percentComplete | The percentage of sync job completion. | float or integer depending on the type of sync operation |
| remainingTime | The estimated time, in seconds, to complete the operation. | float |
| sliceID | The ID of the slice drive being synced. | integer |

| Name | Description | Type |
|------|-------------|------|
| stage | Present only if the type member is set to remote or clone. Possible values:<br><br>• metadata: Replication is in the process of determining what data needs to be transferred to the remote cluster. Status is not reported for this stage of the replication process.<br><br>• data: Replication is in the process of transferring the bulk of the data to the remote cluster.<br><br>• whole: Indicates backward compatibility of the slice for slice sync jobs. | string |
| snapshotID | The ID of the snapshot the clone was created from. Present only if the type member is set to clone. | integer |
| srcServiceID | The source service ID. | integer |
| srcVolumeID | The source volume ID. | integer |
| totalBytes | The total number of bytes of the clone. Present only if the type member is set to clone or slice. | integer |
| type | The type of sync operation. Possible values:<br><br>• clone<br>• slice<br>• block<br>• remote | string |

**Find more information**

ListSyncJobs

# task (virtual volumes)

The task object contains information about a currently running or finished virtual volume task in the system. You can use the `ListVirtualVolumeTasks` method to retrieve this

information for all virtual volume tasks.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| cancelled | Indicates whether or not the task was cancelled. Possible values:<br><br>• true<br><br>• false | boolean |
| cloneVirtualVolumeID | The unique virtual volume ID of the virtual volume being cloned (for clone tasks). | UUID |
| parentMetadata | An object containing metadata of the parent for tasks which clone or create snapshots of a virtual volume. | JSON object |
| parentTotalSize | The total space available (in bytes) on the parent for clone or snapshot tasks. | integer |
| parentUsedSize | The used space of the parent (in bytes) for clone or snapshot tasks. | integer |

| Name | Description | Type |
|---|---|---|
| operation | The type of operation the task is performing. Possible values:<br><br>• unknown: The task operation is unknown.<br>• prepare: The task is preparing a virtual volume.<br>• snapshot: The task is creating a snapshot of a virtual volume.<br>• rollback: The task is rolling back a virtual volume to a snapshot.<br>• clone: The task is creating a clone of the virtual volume.<br>• fastClone: The task is creating a fast clone of a virtual volume.<br>• copyDiffs: The task is copying differing blocks to a virtual volume. | string |
| status | The current status of the virtual volume task. Possible values:<br><br>• Error: The task has failed and returned an error.<br>• Queued: The task is waiting to be run.<br>• Running: The task is currently running.<br>• Success: The task has completed successfully. | string |
| virtualVolumeHostID | The unique ID of the host that started the task. | UUID |
| virtualVolumeID | The new, unique virtual volume ID (for tasks that create a new virtual volume). | UUID |
| virtualVolumeTaskID | The unique ID of the task. | UUID |

# Find more information

[ListVirtualVolumeTasks](#)

# usmUser

You can use the SNMP usmUser object with the `SetSnmpInfo` API method to configure SNMP on the storage cluster.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| access | The type of SNMP access for this user. Possible values:<br><br>• rouser: Read-only access.<br><br>• rwuser: Read-write access. All Element software MIB objects are read-only. | string |
| name | The name of the user. | string |
| password | The password of the user. | string |
| passphrase | The passphrase of the user. | string |
| secLevel | The type of credentials required for this user. Possible values:<br><br>• noauth: No password or passphrase is required.<br><br>• auth: A password is required for user access.<br><br>• priv: A password and passphrase are required for user access. | string |

## Find more information

SetSnmpInfo

# virtualNetwork

The virtualNetwork object contains information about a specific virtual network. You can use the `ListVirtualNetworks` API method to retrieve a list of this information for all virtual networks in the system.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| addressBlocks | The range of address blocks currently assigned to the virtual network. Members:<br><br>• available: Binary string in "1"s and "0"s. "1" denotes that the IP address is available, and "0" denotes that the IP is not available. The string is read from right to left with the digit to the far right being the first IP address in the list of address blocks.<br><br>• size: The size of this block of addresses.<br><br>• start: The first IP address in the block. | JSON object array |
| attributes | List of name-value pairs in JSON object format. | JSON object |
| name | The name assigned to the virtual network. | string |
| netmask | The IP address of the netmask for the virtual network. | string |
| svip | The storage IP address for the virtual network. | string |
| gateway | The gateway used for the virtual network. | string |
| virtualNetworkID | The unique identifier for a virtual network. | integer |
| virtualNetworkTag | The VLAN tag identifier. | integer |

## Find more information

ListVirtualNetworks

# virtualVolume

The virtualVolume object contains configuration information about a virtual volume as well as information about snapshots of the virtual volume. It does not include runtime or usage information. You can use the `ListVirtualVolumes` method to retrieve this information for a cluster.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| bindings | A list of binding IDs for this virtual volume. | UUID array |
| children | A list of virtual volume UUIDs that are children of this virtual volume. | UUID array |
| descendants | When you pass recursive: true to the ListVirtualVolumes method, contains a list of virtual volume UUIDs that are descendants of this virtual volume. | UUID array |
| metadata | Key-value pairs of the virtual volume's metadata, such as virtual volume type, guest OS type, and so on. | JSON object |
| parentVirtualVolumeID | The virtual volume ID of the parent virtual volume. If the ID is all zeros, this is an independent virtual volume with no link to a parent. | UUID |
| snapshotID | The ID of the underlying volume snapshot. This value is "0" if the virtual volume does not represent a snapshot. | integer |
| snapshotInfo | The snapshot object for the associated snapshot (null if nonixestent). | snapshot |

| Name | Description | Type |
|------|-------------|------|
| status | Current status of the virtual volume. Possible values:<br><br>• cloning: The virtual volume is being processed in response to a clone or snapshot operation.<br><br>• waiting: The virtual volume is waiting for a snapshot operation to complete.<br><br>• ready: The virtual volume is ready for general purpose use. | string |
| storageContainer | An object describing the storage container that owns this virtual volume. | storageContainer |
| virtualVolumeID | The unique ID of the virtual volume. | UUID |
| virtualVolumeType | The type of the virtual volume. | string |
| volumeID | The ID of the underlying volume. | integer |
| volumeInfo | When you pass details: true to the ListVirtualVolumes method, this member is an object describing the volume. | volume |

## Find more information

- ListVirtualVolumes
- snapshot
- storageContainer
- volume

# volume

The volume object contains configuration information about unpaired or paired volumes. It does not include runtime or usage information, and does not contain information about virtual volumes.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| access | The type of access allowed for the volume. Possible values:<br><br>• `readOnly`: Only read operations are allowed.<br><br>• `readWrite`: Reads and writes are allowed.<br><br>• `locked`: No reads or writes are allowed.<br><br>• `replicationTarget`: Designated as a target volume in a replicated volume pair. | string |
| accountID | The accountID of the account containing the volume. | integer |
| attributes | List of name-value pairs in JSON object format. | JSON object |
| blockSize | The size of blocks on the volume. | integer |
| createTime | The UTC+0 formatted time the volume was created. | ISO 8601 string |
| currentProtectionScheme | The protection scheme that is being used for this volume. If a volume is converting from one protection scheme to another, this member reflects the protection scheme to which the volume is converting. | string |
| deleteTime | The UTC+0 formatted time the volume was deleted. | ISO 8601 string |
| enable512e | If set to true, the volume provides 512 byte sector emulation. | boolean |
| enableSnapMirrorReplication | Whether or not the volume can be used for replication with SnapMirror endpoints. | boolean |
| fifoSize | Specifies the maximum number of snapshots of the volume to be maintained simultaneously if using the First-In-First-Out (FIFO) snapshot retention mode. | integer |

| Name | Description | Type |
|------|-------------|------|
| iqn | The iSCSI Qualified Name of the volume. | string |
| lastAccessTime | The last time any access (including I/O) to the volume occurred (formatted as UTC+0). If the last access time is not known, this value is null. | ISO 8601 string |
| lastAccessTimeIO | The last time any I/O to the volume occurred (formatted as UTC+0). If the last access time is not known, this value is null. | ISO 8601 string |
| minFifoSize | Specifies the minimum number of First-In-First-Out (FIFO) snapshot slots reserved simultaneously by the volume if using the First-In-First-Out (FIFO) snapshot retention mode. | integer |
| name | The name of the volume as provided at creation time. | string |
| previousProtectionScheme | If a volume is converting from one protection scheme to another, this member reflects the protection scheme from which the volume is converting. This member does not change until a conversion is started. If a volume has never been converted, this member is null. | string |
| purgeTime | The UTC+0 formatted time the volume was purged from the system. | ISO 8601 string |
| qos | The quality of service settings for this volume. | QoS |
| qosPolicyID | The QoS policy ID associated with the volume. The value is null if the volume is not associated with a policy. | integer |
| scsiEUIDeviceID | Globally unique SCSI device identifier for the volume in EUI-64 based 16-byte format. | string |

| Name | Description | Type |
|------|-------------|------|
| scsiNAADeviceID | Globally unique SCSI device identifier for the volume in NAA IEEE Registered Extended format. | string |
| sliceCount | The number of slices on the volume. This value is always "1". | integer |
| status | The current status of the volume. Possible values:<br><br>• init: A volume that is being initialized and is not ready for connections.<br><br>• active: An active volume ready for connections.<br><br>• deleted: A volume that has been marked for deletion, but not yet purged. | string |
| totalSize | The total bytes of provisioned capacity. | integer |
| virtualVolumeID | The unique virtual volume ID associated with the volume, if any. | UUID |
| volumeAccessGroups | List of IDs pf volume access groups to which a volume belongs. This value is an empty list if a volume does not belong to any volume access groups. | integer array |
| volumeConsistencyGroupUUID | The universally unique ID of the volume consistency group of which the volume is a member. | UUID |
| volumeID | The unique volumeID for the volume. | integer |
| volumePairs | Information about a paired volume. Visible only if a volume is paired. This value is an empty list if the volume is not paired. | volumePair array |
| volumeUUID | The universally unique ID of the volume. | UUID |

## Find more information

-
-
-
-
-

# volumeAccessGroup

The volumeAccessGroup object contains information about a specific volume access group. You can retrieve a list of this information for all access groups with the API method `ListVolumeAccessGroups`.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| attributes | List of name-value pairs in JSON object format. | JSON object |
| deletedVolumes | Array of volumes that have been deleted from the volume access group that have not yet been purged from the system. | integer array |
| initiatorIDs | A list of IDs of initiators that are mapped to the volume access group. | integer array |
| initiators | Array of unique IQN/WWPN initiators that are mapped to the volume access group. | string array |
| name | Name of the volume access group. | string |
| volumeAccessGroupID | Unique VolumeAccessGroupID identifier for the volume access group. | integer |
| volumes | A list of VolumeIDs belonging to the volume access group. | integer array |

## Find more information

ListVolumeAccessGroups

# volumePair

The volumePair object contains information about a volume that is paired with another volume on a different cluster. If the volume is not paired, this object is empty. You can use the `ListActivePairedVolumes` and `ListActiveVolumes` API methods to return information about paired volumes.

## Object members

This object contains the following members:

| Name | Description | Type |
|------|-------------|------|
| clusterPairID | The cluster on which the volume is paired. | integer |
| remoteReplication | Details on volume replication. Members:<br><br>• mode: (string) One of "Async", "Sync", or "SnapshotsOnly".<br>• pauseLimit: (integer) Internal use only.<br>• remoteServiceID: (integer) The remote slice service ID.<br>• resumeDetails: (string) Reserved for future use.<br>• snapshotReplication (JSON object)<br>  ◦ state: (string) The state of the ongoing snapshot replication, if one is in progress.<br>  ◦ stateDetails: (string) Reserved for future use.<br>• state: (string) The state of the volume replication.<br>• stateDetails: (string) Reserved for future use. | JSON object |
| remoteSliceID | The cluster-defined slice ID on the remote cluster. | integer |

| Name | Description | Type |
|------|-------------|------|
| remoteVolumeID | The ID of the volume on the remote cluster that the local volume is paired with. | integer |
| remoteVolumeName | The name of the remote volume. | string |
| volumePairUUID | A universally unique, cluster-defined identifier for this pairing in a canonical format. | string |

## Find more information

- ListActivePairedVolumes
- ListActiveVolumes

# volumeStats

The volumeStats object contains statistical data for an individual volume.

## Object members

You can use the following methods to get volumeStats objects for some or all volumes:

- GetVolumeStats
- ListVolumeStatsByAccount
- ListVolumeStatsByVolume
- ListVolumeStatsByVolumeAccessGroup

This object contains the following members:

| Name | Description | Calculation | Type |
|------|-------------|-------------|------|
| accountID | The ID of the account of the volume owner. | N/A | integer |
| actualIOPS | The current actual IOPS to the volume in the last 500 milliseconds. | Point in time | integer |

| Name | Description | Calculation | Type |
|------|-------------|-------------|------|
| asyncDelay | The length of time since the volume was last synced with the remote cluster. If the volume is not paired, this is null. **Note:** A target volume in an active replication state always has an asyncDelay of 0 (zero). Target volumes are system-aware during replication and assume asyncDelay is accurate at all times. | N/A | ISO 8601 duration string or null |
| averageIOPSize | The average size in bytes of recent I/O to the volume in the last 500 milliseconds. | Point in time | integer |
| burstIOPSCredit | The total number of IOP credits available to the user. When volumes are not using up to the configured maxIOPS, credits are accrued. | N/A | integer |
| clientQueueDepth | The number of outstanding read and write operations to the volume. | N/A | integer |
| desiredMetadataHosts | The metadata (slice) services being migrated to if the volume metadata is being migrated between metadata services. A "null" value means the volume is not migrating. | N/A | JSON object |
| latencyUSec | The average time, in microseconds, to complete operations to the volume in the last 500 milliseconds. A "0" (zero) value means there is no I/O to the volume. | Point in time | integer |

| Name | Description | Calculation | Type |
|---|---|---|---|
| metadataHosts | The metadata (slice) services on which the volume metadata resides. Possible values:<br><br>• primary: The primary metadata services hosting the volume.<br><br>• liveSecondaries: Secondary metadata services that are currently in a "live" state.<br><br>• deadSecondaries: Secondary metadata services that are in a dead state. | N/A | JSON object |
| normalizedIOPS | Average number of IOPS for the entire cluster in the last 500 milliseconds. | Point in time | integer |
| nonZeroBlocks | The total number of 4KiB blocks that contain data after the last garbage collection operation has completed. | N/A | integer |
| readBytes | The total cumulative bytes read from the volume since the creation of the volume. | Monotonically increasing | integer |
| readBytesLastSample | The total number of bytes read from the volume during the last sample period. | Point in time | integer |
| readLatencyUSec | The average time, in microseconds, to complete read operations to the volume in the last 500 milliseconds. | Point in time | integer |
| readLatencyUSecTotal | The total time spent performing read operations from the volume. | Monotonically increasing | integer |

| Name | Description | Calculation | Type |
|------|-------------|-------------|------|
| readOps | The total read operations to the volume since the creation of the volume. | Monotonically increasing | integer |
| readOpsLastSample | The total number of read operations during the last sample period. | Point in time | integer |
| samplePeriodMSec | The length of the sample period, in milliseconds. | N/A | integer |
| throttle | A floating value between 0 and 1 that represents how much the system is throttling clients below their maxIOPS because of re-replication of data, transient errors, and snapshots taken. | N/A | float |
| timestamp | The current time in UTC+0 format. | N/A | ISO 8601 date string |
| unalignedReads | The total cumulative unaligned read operations to a volume since the creation of the volume. | Monotonically increasing | integer |
| unalignedWrites | The total cumulative unaligned write operations to a volume since the creation of the volume. | Monotonically increasing | integer |
| volumeAccessGroups | The list of IDs of volume access group(s) to which a volume belongs. | N/A | integer array |
| volumeID | The ID of the volume. | N/A | integer |
| volumeSize | Total provisioned capacity in bytes. | N/A | integer |

| Name | Description | Calculation | Type |
|------|-------------|-------------|------|
| volumeUtilization | A floating point value that describes how fully the client is using the volume's input / output capabilities in comparison with the maxIOPS QoS setting for that volume. Possible values:<br><br>• 0: The client is not using the volume.<br><br>• 0.01 to 0.99: The client is not fully utilizing the volume's IOPS capabilities.<br><br>• 1.00: The client is fully utilizing the volume up to the IOPS limit set by the maxIOPS setting.<br><br>• > 1.00: The client is utilizing more than the limit set by maxIOPS. This is possible when the burstIOPS QoS setting is set higher than maxIOPS. For example, if maxIOPS is set to 1000 and burstIOPS is set to 2000, the `volumeUtilization` value would be 2.00 if the client fully utilizes the volume. | N/A | float |
| writeBytes | The total cumulative bytes written to the volume since the creation of the volume. | Monotonically increasing | integer |
| writeBytesLastSample | The total number of bytes written to the volume during the last sample period. | Monotonically increasing | integer |

| Name | Description | Calculation | Type |
|------|-------------|-------------|------|
| writeLatencyUSec | The average time, in microseconds, to complete write operations to a volume in the last 500 milliseconds. | Point in time | integer |
| writeLatencyUSecTotal | The total time spent performing write operations to the volume. | Monotonically increasing | integer |
| writeOps | The total cumulative write operations to the volume since the creation of the volume. | Monotonically increasing | integer |
| writeOpsLastSample | The total number of write operations during the last sample period. | Point in time | integer |
| zeroBlocks | The total number of empty 4KiB blocks without data after the last round of garbage collection operation has completed. | Point in time | integer |