

Storage APIs

PersistentVolumeClaim [v1]

[PersistentVolumeClaim \[v1\]](#)

PersistentVolume [v1]

[PersistentVolume \[v1\]](#)

StorageClass [storage.k8s.io/v1]

[StorageClass \[storage.k8s.io/v1\]](#)

Ceph Storage APIs

CephCluster [cephclusters.ceph

CephFilesystem [cephfilesystems.ceph

CephBlockl

CephObjectStore [cephobjectstores.ceph

CephObjectStoreUser [cephobjectstoreusers.ceph

PersistentVolumeClaim [v1]

Description

PersistentVolumeClaim is a user's request for and claim to a persistent volume

Type

object

Specification

Property	Type	Description
apiVersion	string	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
kind	string	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds

Property	Type	Description
metadata	ObjectMeta	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
spec	object	PersistentVolumeClaimSpec describes the common attributes of storage devices and allows a Source for provider-specific attributes
status	object	PersistentVolumeClaimStatus is the current status of a persistent volume claim.

.spec

Description

PersistentVolumeClaimSpec describes the common attributes of storage devices and allows a Source for provider-specific attributes

Type

object

Property	Type	Description
accessModes	array	accessModes contains the desired access modes volume should have. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1
dataSource	object	TypedLocalObjectReference contains enough information to let you locate the typed referenced

Property	Type	Description
		object inside the same namespace.
<code>dataSourceRef</code>	<code>object</code>	TypedObjectReference contains enough information to let you locate the typed referenced object
<code>resources</code>	<code>object</code>	VolumeResourceRequirements describes the storage resource requirements for a volume.
<code>selector</code>	<code>object</code>	A label selector is a label query over a set of resources. The result of matchLabels and matchExpressions are ANDed. An empty label selector matches all objects. A null label selector matches no objects.
<code>storageClassName</code>	<code>string</code>	storageClassName is the name of the StorageClass required by the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#class-1
<code>volumeAttributesClassName</code>	<code>string</code>	volumeAttributesClassName may be used to set the VolumeAttributesClass used by this claim. If specified, the CSI driver will create or update the volume with attributes defined in the corresponding VolumeAttributesClass. This has a different purpose than storageClassName, it can be changed after the claim is created. An empty string value means that VolumeAttributesClass will be applied to the claim; it's not allowed to reset this field to empty string on

Property	Type	Description
		is set. If unspecified and the PersistentVolumeClaim is unbound, the default VolumeAttributesClass will be used by the persistentvolume controller if it exists. If the resource referred to by volumeAttributesClass does not exist, this PersistentVolumeClaim will be set to Pending state, as reflected by the modifyVolumeStatus field, until such as a resource exists. More info: https://kubernetes.io/docs/concepts/storage/volume-attributes-classes/ (Beta) Using this field requires the VolumeAttributesClass feature gate to be enabled (off by default).
<code>volumeMode</code>	<code>string</code>	volumeMode defines what type of volume is required by the claim. Value of Filesystem is implied when not included in claim spec. Possible enum values: <code>"Block"</code> means the volume will not be formatted with a filesystem and will remain a raw block device. <code>"Filesystem"</code> means the volume will be or is formatted with a filesystem.
<code>volumeName</code>	<code>string</code>	volumeName is the binding reference to the PersistentVolume backing this claim.

.spec.accessModes

Description

accessModes contains the desired access modes the volume should have. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1>

Type

array

.spec.accessModes[]

Type

string

.spec.dataSource

Description

TypedLocalObjectReference contains enough information to let you locate the typed referenced object inside the same namespace.

Type

object

Required

kind

name

Property	Type	Description
<code>apiGroup</code>	<code>string</code>	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
<code>kind</code>	<code>string</code>	Kind is the type of resource being referenced
<code>name</code>	<code>string</code>	Name is the name of resource being referenced

.spec.dataSourceRef

Description

TypedObjectReference contains enough information to let you locate the typed referenced object

Type

object

Required

kind name

Property	Type	Description
apiGroup	string	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	string	Kind is the type of resource being referenced
name	string	Name is the name of resource being referenced
namespace	string	Namespace is the namespace of resource being referenced Note that when a namespace is specified, a gateway.networking.k8s.io/ReferenceGrant object is required in the referent namespace to allow that namespace's owner to accept the reference. See the ReferenceGrant documentation for details. (Alpha) This field requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

.spec.resources

Description

VolumeResourceRequirements describes the storage resource requirements for a volume.

Type

object

Property	Type	Description
limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

.spec.resources.limits

Description

Limits describes the maximum amount of compute resources allowed. More info:
<https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

.spec.resources.requests

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

.spec.selector

Description

A label selector is a label query over a set of resources. The result of matchLabels and matchExpressions are ANDed. An empty label selector matches all objects. A null label selector matches no objects.

Type

object

Property	Type	Description
matchExpressions	array	matchExpressions is a list of label selector requirements. The requirements are ANDed.
matchLabels	object	matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

.spec.selector.matchExpressions

Description

matchExpressions is a list of label selector requirements. The requirements are ANDed.

Type

array

.spec.selector.matchExpressions[]

Description

A label selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

object

Required

key operator

Property	Type	Description
key	string	key is the label key that the selector applies to.
operator	string	operator represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists and DoesNotExist.
values	array	values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

.spec.selector.matchExpressions[].values

Description

values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

Type

array

.spec.selector.matchExpressions[].values[]

Type

string

.spec.selector.matchLabels

Description

matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

Type

object

.status

Description

PersistentVolumeClaimStatus is the current status of a persistent volume claim.

Type

object

Property	Type	Description
accessModes	array	accessModes contains the actual access modes the underlying storage engine supports. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1
allocatedResourceStatuses	object	allocatedResourceStatuses stores status of resources being resized for the given PVC. Key names must follow standard Kubernetes label syntax. Valid values are either: * Un-prefixed keys: - storage - the controller that manages the volume. * Custom resources must use implementation-defined prefixed names such as "example.com/my-custom-resource" Apart from the above values - keys that are unprefixed or have kubernetes.io prefix are considered reserved and hence may not be used. ClaimResourceStatus can be in any of following states: - ControllerResizeInProgress: State set when resize controller starts resizing the volume plane. - ControllerResizeFailed: State set when resize controller has failed in resize controller with a termination error. - NodeResizePending: State set when resize controller has finished resizing the volume but further action is needed on the node. - NodeResizeInProgress: State set when kubelet is resizing the volume. - NodeResizeFailed: State set when resizing has failed in kubelet with a termination error. Transient errors don't set NodeResizeFailed. Example: if expanding a PVC for more capacity, the pvc.status.allocatedResourceStatus['storage.kubernetes.io/resizing'] field can be one of the following states: - ControllerResizeInProgress - ControllerResizeFailed - NodeResizePending - NodeResizeInProgress - NodeResizeFailed pvc.status.allocatedResourceStatus['storage.kubernetes.io/resizing']

Property	Type	Description
		"ControllerResizeFailed" - pvc.status.allocatedResourceStatus["storage"] "NodeResizePending" - pvc.status.allocatedResourceStatus["storage"] "NodeResizeInProgress" - pvc.status.allocatedResourceStatus["storage"] "NodeResizeFailed" When this field is not null means that no resize operation is in progress for the given PVC. A controller that receives PVC update with unknown resourceName or ClaimResourceName should ignore the update for the purpose of resizing. For example - a controller that is responsible for resizing capacity of the volume should ignore PVC updates that change other values associated with PVC. This is an alpha field and requires enabling RecoverVolumeExpansionFailure feature.
allocatedResources	object	allocatedResources tracks the resources allocated to a PVC including its capacity. Key names follow standard Kubernetes label syntax. Valid values are either: * Un-prefixed keys: - storage - the capacity of the volume. * Custom resources must use implementation-defined prefixed names such as "example.com/my-custom-resource" Apart from these values - keys that are unprefixed or have kubernetes.io prefix are considered reserved and hence may not be used. Capacity reported here may be larger than the requested capacity when a volume expansion operation is requested. For storage quota, the larger value should be used.

Property	Type	Description
		allocatedResources and PVC.spec.resour If allocatedResources is not set, PVC.spec alone is used for quota calculation. If a vol expansion capacity request is lowered, allocatedResources is only lowered if ther expansion operations in progress and if th volume capacity is equal or lower than the capacity. A controller that receives PVC update with unknown resourceName should ignore the the purpose it was designed. For example controller that only is responsible for resizi of the volume, should ignore PVC updates change other valid resources associated v This is an alpha field and requires enablin RecoverVolumeExpansionFailure feature.
capacity	object	capacity represents the actual resources c underlying volume.
conditions	array	conditions is the current Condition of persi volume claim. If underlying persistent volu resized then the Condition will be set to 'R

Property	Type	Description
<code>currentVolumeAttributesClassName</code>	<code>string</code>	<code>currentVolumeAttributesClassName</code> is the name of the <code>VolumeAttributesClass</code> the <code>PV</code> When unset, there is no <code>VolumeAttributeC</code> to this <code>PersistentVolumeClaim</code> This is a be requires enabling <code>VolumeAttributesClass</code> f by default).
<code>modifyVolumeStatus</code>	<code>object</code>	<code>ModifyVolumeStatus</code> represents the status <code>ControllerModifyVolume</code> operation
<code>phase</code>	<code>string</code>	<code>phase</code> represents the current phase of <code>PersistentVolumeClaim</code>. Possible enum values: <code>"Bound"</code> used for <code>PersistentVolumeC</code> are bound <code>"Lost"</code> used for <code>PersistentVolumeCl</code> lost their underlying <code>PersistentVolume</code>. was bound to a <code>PersistentVolume</code> and does not exist any longer and all data c lost. <code>"Pending"</code> used for <code>PersistentVolum</code> are not yet bound

.status.accessModes

Description

accessModes contains the actual access modes the volume backing the PVC has. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1>

Type

array

.status.accessModes[]

Type

string

.status.allocatedResourceStatuses

Description

allocatedResourceStatuses stores status of resource being resized for the given PVC. Key names follow standard Kubernetes label syntax. Valid values are either: * Un-prefixed keys:

- storage - the capacity of the volume. * Custom resources must use implementation-defined prefixed names such as "example.com/my-custom-resource" Apart from above values - keys that are unprefixed or have kubernetes.io prefix are considered reserved and hence may not be used. ClaimResourceStatus can be in any of following states: -

ControllerResizeInProgress: State set when resize controller starts resizing the volume in control-plane. - ControllerResizeFailed: State set when resize has failed in resize controller with a terminal error. - NodeResizePending: State set when resize controller has finished resizing the volume but further resizing of volume is needed on the node. -

NodeResizeInProgress: State set when kubelet starts resizing the volume. -

NodeResizeFailed: State set when resizing has failed in kubelet with a terminal error.

Transient errors don't set NodeResizeFailed. For example: if expanding a PVC for more capacity - this field can be one of the following states: -

pvc.status.allocatedResourceStatus['storage'] = "ControllerResizeInProgress" -

pvc.status.allocatedResourceStatus['storage'] = "ControllerResizeFailed" -

pvc.status.allocatedResourceStatus['storage'] = "NodeResizePending" -

pvc.status.allocatedResourceStatus['storage'] = "NodeResizeInProgress" -

pvc.status.allocatedResourceStatus['storage'] = "NodeResizeFailed" When this field is not set, it means that no resize operation is in progress for the given PVC. A controller that receives PVC update with previously unknown resourceName or ClaimResourceStatus should ignore the update for the purpose it was designed. For example - a controller that only is responsible for resizing capacity of the volume, should ignore PVC updates that

change other valid resources associated with PVC. This is an alpha field and requires enabling RecoverVolumeExpansionFailure feature.

Type

object

.status.allocatedResources

Description

allocatedResources tracks the resources allocated to a PVC including its capacity. Key names follow standard Kubernetes label syntax. Valid values are either: * Un-prefixed keys: - storage - the capacity of the volume. * Custom resources must use implementation-defined prefixed names such as "example.com/my-custom-resource" Apart from above values - keys that are unprefixed or have kubernetes.io prefix are considered reserved and hence may not be used. Capacity reported here may be larger than the actual capacity when a volume expansion operation is requested. For storage quota, the larger value from allocatedResources and PVC.spec.resources is used. If allocatedResources is not set, PVC.spec.resources alone is used for quota calculation. If a volume expansion capacity request is lowered, allocatedResources is only lowered if there are no expansion operations in progress and if the actual volume capacity is equal or lower than the requested capacity. A controller that receives PVC update with previously unknown resourceName should ignore the update for the purpose it was designed. For example - a controller that only is responsible for resizing capacity of the volume, should ignore PVC updates that change other valid resources associated with PVC. This is an alpha field and requires enabling RecoverVolumeExpansionFailure feature.

Type

object

.status.capacity

Description

capacity represents the actual resources of the underlying volume.

Type

object

.status.conditions

Description

conditions is the current Condition of persistent volume claim. If underlying persistent volume is being resized then the Condition will be set to 'Resizing'.

Type

array

.status.conditions[]

Description

PersistentVolumeClaimCondition contains details about state of pvc

Type

object

Required

type status

Property	Type	Description
lastProbeTime	string	Time is a wrapper around time.Time which supports correct JSON. Wrappers are provided for many of the factory methods.
lastTransitionTime	string	Time is a wrapper around time.Time which supports correct JSON. Wrappers are provided for many of the factory methods.
message	string	message is the human-readable message indicating detail

Property	Type	Description
reason	string	reason is a unique, this should be a short, machine understandable reason for condition's last transition. If it reports "Resizing" underlying persistent volume is being resized.
status	string	Status is the status of the condition. Can be True, False, Unknown. https://kubernetes.io/docs/reference/kubernetes-api/config-interfaces/cluster-api/persistent-volume-claim-conditions/#persistentvolumeclaimcondition
type	string	Type is the type of the condition. More info: https://kubernetes.io/docs/reference/kubernetes-api/config-interfaces/cluster-api/persistent-volume-claim-conditions/#persistentvolumeclaimcondition

.status.modifyVolumeStatus

Description

ModifyVolumeStatus represents the status object of ControllerModifyVolume operation

Type

object

Required

status

Property	Type	Description
status	string	status is the status of the ControllerModifyVolume operation. It

Property	Type	Description
		can be in any of following states: • Pending Pending indicates that the PersistentVolumeClaim cannot be modified due to unmet requirements, such as the specified VolumeAttributesClass not existing. • InProgress InProgress indicates that the volume is being modified. • Infeasible Infeasible indicates that the request has been rejected as invalid by the CSI driver. To resolve the error, a valid VolumeAttributesClass needs to be specified. Note: New statuses can be added in the future. Consumers should check for unknown statuses and fail appropriately. Possible enum values: • <code>"InProgress"</code> InProgress indicates that the volume is being modified • <code>"Infeasible"</code> Infeasible indicates that the request has been rejected as invalid by the CSI driver. To resolve the error, a valid VolumeAttributesClass needs to be specified • <code>"Pending"</code> Pending indicates that the PersistentVolumeClaim cannot be modified due to unmet

Property	Type	Description
		requirements, such as the specified VolumeAttributesClass not existing
targetVolumeAttributesClassName	string	targetVolumeAttributesClassName is the name of the VolumeAttributesClass the PVC currently being reconciled

API Endpoints

The following API endpoints are available:

- `/kubernetes/{cluster}/api/v1/namespaces/{namespace}/persistentvolumeclaims`
 - `DELETE` : delete collection of PersistentVolumeClaim
 - `GET` : list objects of kind PersistentVolumeClaim
 - `POST` : create a new PersistentVolumeClaim
- `/kubernetes/{cluster}/api/v1/namespaces/{namespace}/persistentvolumeclaims/{name}`
 - `DELETE` : delete the specified PersistentVolumeClaim
 - `GET` : read the specified PersistentVolumeClaim
 - `PATCH` : partially update the specified PersistentVolumeClaim
 - `PUT` : replace the specified PersistentVolumeClaim
- `/kubernetes/{cluster}/api/v1/namespaces/{namespace}/persistentvolumeclaims/{name}/status`
 - `GET` : read status of the specified PersistentVolumeClaim
 - `PATCH` : partially update status of the specified PersistentVolumeClaim

- `PUT`: replace status of the specified PersistentVolumeClaim

/kubernetes/{cluster}/api/v1/namespaces/{namespace}/persistentvolumeclaims

HTTP method

DELETE

Description

delete collection of PersistentVolumeClaim

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

GET

Description

list objects of kind PersistentVolumeClaim

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolumeClaimList</code> schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a new PersistentVolumeClaim

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized <code>dryRun</code> directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a <code>BadRequest</code> error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>PersistentVolumeClaim</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolumeClaim</code> schema
201 - Created	<code>PersistentVolumeClaim</code> schema

HTTP code	Response body
202 - Accepted	<code>PersistentVolumeClaim</code> schema
401 - Unauthorized	Empty

/kubernetes/{cluster}/api/v1/namespaces/{namespace}/persistentvolumeclaims/{name}

HTTP method

DELETE

Description

delete the specified PersistentVolumeClaim

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
202 - Accepted	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

GET

Description

read the specified PersistentVolumeClaim

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolumeClaim</code> schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update the specified PersistentVolumeClaim

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are

Parameter	Type	Description
		present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolumeClaim</code> schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified PersistentVolumeClaim

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a

Parameter	Type	Description
		BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
body	PersistentVolumeClaim schema	application/json formatted

HTTP responses

HTTP code	Response body
200 - OK	PersistentVolumeClaim schema
201 - Created	PersistentVolumeClaim schema
401 - Unauthorized	Empty

/kubernetes/{cluster}/api/v1/namespaces/{namespace}/persistentvolumeclaims/{name}/status

HTTP method

GET

Description

read status of the specified PersistentVolumeClaim

HTTP responses

HTTP code	Response body
200 - OK	PersistentVolumeClaim schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update status of the specified PersistentVolumeClaim

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolumeClaim</code> schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace status of the specified PersistentVolumeClaim

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>PersistentVolumeClaim</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolumeClaim</code> schema
201 - Created	<code>PersistentVolumeClaim</code> schema
401 - Unauthorized	Empty

PersistentVolume [v1]

Description

PersistentVolume (PV) is a storage resource provisioned by an administrator. It is analogous to a node. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes>

Type

object

Specification

Property	Type	Description
<code>apiVersion</code>	<code>string</code>	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources

Property	Type	Description
kind	string	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
metadata	ObjectMeta	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
spec	object	PersistentVolumeSpec is the specification of a persistent volume.
status	object	PersistentVolumeStatus is the current status of a persistent volume.

.spec

Description

PersistentVolumeSpec is the specification of a persistent volume.

Type

object

Property	Type	Description
accessModes	array	accessModes contains all ways the volume can be mounted. More info:

Property	Type	Description
		https://kubernetes.io/docs/concepts/storage/persistentvolumes#access-modes ↗
<code>awsElasticBlockStore</code>	<code>object</code>	Represents a Persistent Disk resource in AWS. An AWS EBS disk must exist before mounting to a container. The disk must also be in the same . zone as the kubelet. An AWS EBS disk can only be mounted as read/write once. AWS EBS volumes support ownership management and SELinux relabeling.
<code>azureDisk</code>	<code>object</code>	AzureDisk represents an Azure Data Disk mounted to the host and bind mount to the pod.
<code>azureFile</code>	<code>object</code>	AzureFile represents an Azure File Service mounted to the host and bind mount to the pod.
<code>capacity</code>	<code>object</code>	capacity is the description of the persistent volume resources and capacity. More info: https://kubernetes.io/docs/concepts/storage/persistentvolumes#capacity ↗
<code>cephfs</code>	<code>object</code>	Represents a Ceph Filesystem mount that lasts the lifetime of a pod Cephfs volumes do not support ownership management or SELinux relabeling

Property	Type	Description
<code>cinder</code>	<code>object</code>	Represents a cinder volume resource in OpenStack. Cinder volume must exist before mounting to container. The volume must also be in the same region as the kubelet. Cinder volumes support ownership management and SELinux relabeling.
<code>claimRef</code>	<code>object</code>	ObjectReference contains enough information to let you inspect or modify the referred object.
<code>csi</code>	<code>object</code>	Represents storage that is managed by an external CSI volume driver
<code>fc</code>	<code>object</code>	Represents a Fibre Channel volume. Fibre Channel volumes can only be mounted as read/write once. Fibre Channel volumes support ownership management and SELinux relabeling.
<code>flexVolume</code>	<code>object</code>	FlexPersistentVolumeSource represents a generic persistent volume resource that is provisioned/attached using an exec based plugin.
<code>flocker</code>	<code>object</code>	Represents a Flocker volume mounted by the kubelet agent. One and only one of datasetName and datasetUUID should be set. Flocker volumes support ownership management or SELinux relabeling.

Property	Type	Description
<code>gcePersistentDisk</code>	<code>object</code>	Represents a Persistent Disk resource in Google Compute Engine. A GCE PD must exist before mounting to a container. The disk must also be in the same GCE project and zone as the kubelet. A GCE PD can only be mounted as read/write once or read-only many times. Containers do not support ownership management and SELinux relabeling.
<code>glusterfs</code>	<code>object</code>	Represents a Glusterfs mount that lasts the lifetime of a pod. Glusterfs volumes do not support ownership management or SELinux relabeling.
<code>hostPath</code>	<code>object</code>	Represents a host path mapped into a pod. HostPath volumes do not support ownership management or SELinux relabeling.
<code>iscsi</code>	<code>object</code>	ISCSIPersistentVolumeSource represents an iSCSI disk. iSCSI volumes can only be mounted as read/write once. iSCSI volumes support ownership management and SELinux relabeling.
<code>local</code>	<code>object</code>	Local represents directly-attached storage with no affinity
<code>mountOptions</code>	<code>array</code>	mountOptions is the list of mount options, e.g. ["soft"]. Not validated - mount will simply fail if

Property	Type	Description
		invalid. More info: https://kubernetes.io/docs/concepts/storage/persistentvolumes/#mount-options ↗
<code>nfs</code>	<code>object</code>	Represents an NFS mount that lasts the lifetime of a pod. NFS volumes do not support ownership management or SELinux relabeling.
<code>nodeAffinity</code>	<code>object</code>	VolumeNodeAffinity defines constraints that limit the nodes this volume can be accessed from.
<code>persistentVolumeReclaimPolicy</code>	<code>string</code>	<code>persistentVolumeReclaimPolicy</code> defines what to do to a persistent volume when released from its claim. Valid options are Retain (default for manually provisioned PersistentVolumes), Delete (default for dynamically provisioned PersistentVolumes), and Recycle (deprecated). Recycle must be supported by the volume plugin underlying this PersistentVolume. info: https://kubernetes.io/docs/concepts/storage/persistentvolumes#reclaiming ↗ Possible enum values: <code>"Delete"</code> means the volume will be deleted from Kubernetes on release from its claim. The volume plugin must support Deletion. <code>"Recycle"</code> means the volume will be recycled back into the pool of unbound persistent volumes on release from its claim. The volume plugin must support Recycling.

Property	Type	Description
		"Retain" means the volume will be left current phase (Released) for manual reclaim by the administrator. The default policy is FirstFit.
photonPersistentDisk	object	Represents a Photon Controller persistent disk resource.
portworxVolume	object	PortworxVolumeSource represents a Portworx resource.
quobyte	object	Represents a Quobyte mount that lasts the lifetime of a pod. Quobyte volumes do not support ownership management or SELinux relabeling.
rbd	object	Represents a Rados Block Device mount that lasts the lifetime of a pod. RBD volumes support ownership management and SELinux relabeling.
scaleIO	object	ScaleIOPersistentVolumeSource represents a persistent ScaleIO volume
storageClassName	string	storageClassName is the name of StorageClass which this persistent volume belongs to. Empty value means that this volume does not belong to any StorageClass.

Property	Type	Description
<code>storageos</code>	<code>object</code>	Represents a StorageOS persistent volume resource.
<code>volumeAttributesClassName</code>	<code>string</code>	Name of VolumeAttributesClass to which this persistent volume belongs. Empty value is not allowed. When this field is not set, it indicates volume does not belong to any VolumeAttributesClass. This field is mutable and can be changed by the driver after a volume has been updated successfully with a new class. For an unbound PersistentVolumeClaim, volumeAttributesClassName will be matched against unbound PersistentVolumeClaims during the process. This is a beta field and requires enable VolumeAttributesClass feature (off by default)
<code>volumeMode</code>	<code>string</code>	volumeMode defines if a volume is intended to be used with a formatted filesystem or to remain in block state. Value of Filesystem is implied when not included in spec. Possible enum values: <code>"Block"</code> means the volume will not be formatted with a filesystem and will remain a raw block device. <code>"Filesystem"</code> means the volume will be formatted with a filesystem.
<code>vsphereVolume</code>	<code>object</code>	Represents a vSphere volume resource.

.spec.accessModes

Description

accessModes contains all ways the volume can be mounted. More info:
<https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes>

Type

array

.spec.accessModes[]

Type

string

.spec.awsElasticBlockStore

Description

Represents a Persistent Disk resource in AWS. An AWS EBS disk must exist before mounting to a container. The disk must also be in the same AWS zone as the kubelet. An AWS EBS disk can only be mounted as read/write once. AWS EBS volumes support ownership management and SELinux relabeling.

Type

object

Required

volumeID

Property	Type	Description
<code>fsType</code>	<code>string</code>	fsType is the filesystem type of the volume that you want to mount. 1 Ensure that the filesystem type is supported by the host operating sy Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspe More info: https://kubernetes.io/docs/concepts/storage/volumes#awselasticblo
<code>partition</code>	<code>integer</code>	partition is the partition in the volume that you want to mount. If omit the default is to mount by volume name. Examples: For volume /dev you specify the partition as "1". Similarly, the volume partition for /de is "0" (or you can leave the property empty).
<code>readOnly</code>	<code>boolean</code>	readOnly value true will force the readOnly setting in VolumeMounts info: https://kubernetes.io/docs/concepts/storage/volumes#awselasticblo
<code>volumeID</code>	<code>string</code>	volumeID is unique ID of the persistent disk resource in AWS (Amaz EBS volume). More info: https://kubernetes.io/docs/concepts/storage/volumes#awselasticblo

.spec.azureDisk

Description

AzureDisk represents an Azure Data Disk mount on the host and bind mount to the pod.

Type

object

Required

diskName

diskURI

Property	Type	Description
cachingMode	string	cachingMode is the Host Caching mode: None, Read Only, Read Write. Possible enum values: "None" "ReadOnly" "ReadWrite"
diskName	string	diskName is the Name of the data disk in the blob storage
diskURI	string	diskURI is the URI of data disk in the blob storage
fsType	string	fsType is Filesystem type to mount. Must be a filesystem type supported by the host operating system. Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
kind	string	kind expected values are Shared: multiple blob disks per storage account Dedicated: single blob disk per storage account Managed: azure managed data disk (only in managed availability set). defaults to shared Possible enum values: "Dedicated"

Property	Type	Description
		"Managed" "Shared"
<code>readOnly</code>	<code>boolean</code>	<code>readOnly</code> Defaults to false (read/write). <code>ReadOnly</code> here will force the <code>ReadOnly</code> setting in <code>VolumeMounts</code> .

.spec.azureFile

Description

AzureFile represents an Azure File Service mount on the host and bind mount to the pod.

Type

`object`

Required

`secretName`

`shareName`

Property	Type	Description
<code>readOnly</code>	<code>boolean</code>	<code>readOnly</code> defaults to false (read/write). <code>ReadOnly</code> here will force the <code>ReadOnly</code> setting in <code>VolumeMounts</code> .
<code>secretName</code>	<code>string</code>	<code>secretName</code> is the name of secret that contains Azure Storage Account Name and Key
<code>secretNamespace</code>	<code>string</code>	<code>secretNamespace</code> is the namespace of the secret that contains Azure Storage Account Name and Key default is the same as the Pod

Property	Type	Description
<code>shareName</code>	<code>string</code>	shareName is the azure Share Name

`.spec.capacity`

Description

capacity is the description of the persistent volume's resources and capacity. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#capacity>

Type

`object`

`.spec.cephfs`

Description

Represents a Ceph Filesystem mount that lasts the lifetime of a pod Cephfs volumes do not support ownership management or SELinux relabeling.

Type

`object`

Required

`monitors`

Property	Type	Description
<code>monitors</code>	<code>array</code>	monitors is Required: Monitors is a collection of Ceph monitors More info: https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-it ↗

Property	Type	Description
<code>path</code>	<code>string</code>	<code>path</code> is Optional: Used as the mounted root, rather than the full Ceph tree, default is /
<code>readOnly</code>	<code>boolean</code>	<code>readOnly</code> is Optional: Defaults to false (read/write). <code>ReadOnly</code> here will force the <code>ReadOnly</code> setting in <code>VolumeMounts</code> . More info: https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-it ↗
<code>secretFile</code>	<code>string</code>	<code>secretFile</code> is Optional: <code>SecretFile</code> is the path to key ring for User, default is /etc/ceph/user.secret More info: https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-it ↗
<code>secretRef</code>	<code>object</code>	<code>SecretReference</code> represents a Secret Reference. It has enough information to retrieve secret in any namespace
<code>user</code>	<code>string</code>	<code>user</code> is Optional: User is the rados user name, default is admin More info: https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-it ↗

.spec.cephfs.monitors

Description

`monitors` is Required: Monitors is a collection of Ceph monitors More info:
<https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-it>

Type

array

.spec.cephfs.monitors[]

Type

string

.spec.cephfs.secretRef

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

object

Property	Type	Description
name	string	name is unique within a namespace to reference a secret resource.
namespace	string	namespace defines the space within which the secret name must be unique.

.spec.cinder

Description

Represents a cinder volume resource in Openstack. A Cinder volume must exist before mounting to a container. The volume must also be in the same region as the kubelet. Cinder volumes support ownership management and SELinux relabeling.

Type

`object`

Required

`volumeID`

Property	Type	Description
<code>fsType</code>	<code>string</code>	<code>fsType</code> Filesystem type to mount. Must be a filesystem type supported by the host operating system. Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified. More info: https://examples.k8s.io/mysql-cinder-pd/README.md ↗
<code>readOnly</code>	<code>boolean</code>	<code>readOnly</code> is Optional: Defaults to false (read/write). <code>ReadOnly</code> here will force the <code>ReadOnly</code> setting in <code>VolumeMounts</code> . More info: https://examples.k8s.io/mysql-cinder-pd/README.md ↗
<code>secretRef</code>	<code>object</code>	<code>SecretReference</code> represents a Secret Reference. It has enough information to retrieve secret in any namespace
<code>volumeID</code>	<code>string</code>	<code>volumeID</code> used to identify the volume in cinder. More info: https://examples.k8s.io/mysql-cinder-pd/README.md ↗

.spec.cinder.secretRef

Description

`SecretReference` represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

`object`

Property	Type	Description
<code>name</code>	<code>string</code>	name is unique within a namespace to reference a secret resource.
<code>namespace</code>	<code>string</code>	namespace defines the space within which the secret name must be unique.

.spec.claimRef

Description

ObjectReference contains enough information to let you inspect or modify the referred object.

Type

`object`

Property	Type	Description
<code>apiVersion</code>	<code>string</code>	API version of the referent.
<code>fieldPath</code>	<code>string</code>	If referring to a piece of an object instead of an entire object, this string should contain a valid JSON/Go field access statement, such as <code>desiredState.manifest.containers[2]</code>. For example, if the object reference is to a container within a pod, this would take on a value like: <code>"spec.containers{name}"</code> (where "name" refers to the name of the container that triggered the event) or if no container name is specified <code>"spec.containers[2]"</code> (container with index 2 in this pod).

Property	Type	Description
		This syntax is chosen only to have some well-defined way of referencing a part of an object.
<code>kind</code>	<code>string</code>	Kind of the referent. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
<code>name</code>	<code>string</code>	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names ↗
<code>namespace</code>	<code>string</code>	Namespace of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/namespaces/ ↗
<code>resourceVersion</code>	<code>string</code>	Specific resourceVersion to which this reference is made, if any. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#concurrency-control-and-consistency ↗
<code>uid</code>	<code>string</code>	UID of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#uids ↗

Description

Represents storage that is managed by an external CSI volume driver

Type

object

Required

driver

volumeHandle

Property	Type	Description
<code>controllerExpandSecretRef</code>	object	SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace
<code>controllerPublishSecretRef</code>	object	SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace
<code>driver</code>	string	driver is the name of the driver to use for this volume. Required.
<code>fsType</code>	string	fsType to mount. Must be a filesystem type supported by the host operating system. Ex. "ext4", "xfs", "ntfs".
<code>nodeExpandSecretRef</code>	object	SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Property	Type	Description
<code>nodePublishSecretRef</code>	<code>object</code>	SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace
<code>nodeStageSecretRef</code>	<code>object</code>	SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace
<code>readOnly</code>	<code>boolean</code>	readOnly value to pass to ControllerPublishVolumeRequest. Defaults to false (read/write).
<code>volumeAttributes</code>	<code>object</code>	volumeAttributes of the volume to publish.
<code>volumeHandle</code>	<code>string</code>	volumeHandle is the unique volume name returned by the CSI volume plugin's CreateVolume to refer to the volume on all subsequent calls. Required.

`.spec.csi.controllerExpandSecretRef`

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

`object`

Property	Type	Description
<code>name</code>	<code>string</code>	name is unique within a namespace to reference a secret resource.
<code>namespace</code>	<code>string</code>	namespace defines the space within which the secret name must be unique.

`.spec.csi.controllerPublishSecretRef`

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

`object`

Property	Type	Description
<code>name</code>	<code>string</code>	name is unique within a namespace to reference a secret resource.
<code>namespace</code>	<code>string</code>	namespace defines the space within which the secret name must be unique.

`.spec.csi.nodeExpandSecretRef`

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

object

Property	Type	Description
name	string	name is unique within a namespace to reference a secret resource.
namespace	string	namespace defines the space within which the secret name must be unique.

.spec.csi.nodePublishSecretRef

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

object

Property	Type	Description
name	string	name is unique within a namespace to reference a secret resource.
namespace	string	namespace defines the space within which the secret name must be unique.

.spec.csi.nodeStageSecretRef

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

object

Property	Type	Description
name	string	name is unique within a namespace to reference a secret resource.
namespace	string	namespace defines the space within which the secret name must be unique.

.spec.csi.volumeAttributes

Description

volumeAttributes of the volume to publish.

Type

object

.spec.fc

Description

Represents a Fibre Channel volume. Fibre Channel volumes can only be mounted as read/write once. Fibre Channel volumes support ownership management and SELinux relabeling.

Type

object

Property	Type	Description
<code>fsType</code>	<code>string</code>	<code>fsType</code> is the filesystem type to mount. Must be a filesystem type supported by the host operating system. Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
<code>lun</code>	<code>integer</code>	<code>lun</code> is Optional: FC target lun number
<code>readOnly</code>	<code>boolean</code>	<code>readOnly</code> is Optional: Defaults to false (read/write). <code>ReadOnly</code> here will force the <code>ReadOnly</code> setting in <code>VolumeMounts</code> .
<code>targetWWNs</code>	<code>array</code>	<code>targetWWNs</code> is Optional: FC target worldwide names (WWNs)
<code>wwids</code>	<code>array</code>	<code>wwids</code> Optional: FC volume world wide identifiers (wwids) Either <code>wwids</code> or combination of <code>targetWWNs</code> and <code>lun</code> must be set, but not both simultaneously.

`.spec.fc.targetWWNs`

Description

`targetWWNs` is Optional: FC target worldwide names (WWNs)

Type

`array`

`.spec.fc.targetWWNs[]`

Type

`string`

`.spec.fc.wwids`

Description

wwids Optional: FC volume world wide identifiers (wwids) Either wwids or combination of targetWWNs and lun must be set, but not both simultaneously.

Type

`array`

`.spec.fc.wwids[]`

Type

`string`

`.spec.flexVolume`

Description

FlexPersistentVolumeSource represents a generic persistent volume resource that is provisioned/attached using an exec based plugin.

Type

`object`

Required

`driver`

Property	Type	Description
<code>driver</code>	<code>string</code>	driver is the name of the driver to use for this volume.
<code>fsType</code>	<code>string</code>	fsType is the Filesystem type to mount. Must be a filesystem type supported by the host operating system. Ex. "ext4", "xfs",

Property	Type	Description
		"ntfs". The default filesystem depends on FlexVolume script.
<code>options</code>	<code>object</code>	options is Optional: this field holds extra command options if any.
<code>readOnly</code>	<code>boolean</code>	readOnly is Optional: defaults to false (read/write). ReadOnly here will force the ReadOnly setting in VolumeMounts.
<code>secretRef</code>	<code>object</code>	SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

`.spec.flexVolume.options`

Description

options is Optional: this field holds extra command options if any.

Type

`object`

`.spec.flexVolume.secretRef`

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

`object`

Property	Type	Description
<code>name</code>	<code>string</code>	name is unique within a namespace to reference a secret resource.
<code>namespace</code>	<code>string</code>	namespace defines the space within which the secret name must be unique.

.spec.flocker

Description

Represents a Flocker volume mounted by the Flocker agent. One and only one of `datasetName` and `datasetUUID` should be set. Flocker volumes do not support ownership management or SELinux relabeling.

Type

`object`

Property	Type	Description
<code>datasetName</code>	<code>string</code>	<code>datasetName</code> is Name of the dataset stored as metadata -> name on the dataset for Flocker should be considered as deprecated
<code>datasetUUID</code>	<code>string</code>	<code>datasetUUID</code> is the UUID of the dataset. This is unique identifier of a Flocker dataset

.spec.gcePersistentDisk

Description

Represents a Persistent Disk resource in Google Compute Engine. A GCE PD must exist before mounting to a container. The disk must also be in the same GCE project and zone as the kubelet. A GCE PD can only be mounted as read/write once or read-only many times. GCE PDs support ownership management and SELinux relabeling.

Type

object

Required

pdName

Property	Type	Description
fsType	string	fsType is filesystem type of the volume that you want to mount. Tip: Ensure that the filesystem type is supported by the host operating system. Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" unspecified. More info: https://kubernetes.io/docs/concepts/storage/volumes#gcepersistentdisk
partition	integer	partition is the partition in the volume that you want to mount. If omitted, the default is to mount by volume name. Examples: For volume /dev/sda1, you specify the partition as "1". Similarly, the volume partition for /dev/sda is "0" (or you can leave the property empty). More info: https://kubernetes.io/docs/concepts/storage/volumes#gcepersistentdisk
pdName	string	pdName is unique name of the PD resource in GCE. Used to identify the disk in GCE. More info: https://kubernetes.io/docs/concepts/storage/volumes#gcepersistentdisk

Property	Type	Description
<code>readOnly</code>	<code>boolean</code>	<code>readOnly</code> here will force the <code>ReadOnly</code> setting in <code>VolumeMounts</code>. Defaults to <code>false</code>. More info: https://kubernetes.io/docs/concepts/storage/volumes#gcepersistentvolume

`.spec.glusterfs`

Description

Represents a Glusterfs mount that lasts the lifetime of a pod. Glusterfs volumes do not support ownership management or SELinux relabeling.

Type

`object`

Required

`endpoints`

`path`

Property	Type	Description
<code>endpoints</code>	<code>string</code>	<code>endpoints</code> is the endpoint name that details Glusterfs topology. More info: https://examples.k8s.io/volumes/glusterfs/README.md#creating-a-pod
<code>endpointsNamespace</code>	<code>string</code>	<code>endpointsNamespace</code> is the namespace that contains Glusterfs endpoint. If this field is empty, the <code>EndpointNamespace</code> defaults to the same namespace as the bound PVC. More info: https://examples.k8s.io/volumes/glusterfs/README.md#creating-a-pod

Property	Type	Description
<code>path</code>	<code>string</code>	path is the Glusterfs volume path. More info: https://examples.k8s.io/volumes/glusterfs/README.md#creating-a-pod
<code>readOnly</code>	<code>boolean</code>	readOnly here will force the Glusterfs volume to be mounted with read-only permissions. Defaults to false. More info: https://examples.k8s.io/volumes/glusterfs/README.md#creating-a-pod

.spec.hostPath

Description

Represents a host path mapped into a pod. Host path volumes do not support ownership management or SELinux relabeling.

Type

`object`

Required

`path`

Property	Type	Description
<code>path</code>	<code>string</code>	path of the directory on the host. If the path is a symlink, it will follow the link to the real path. More info: https://kubernetes.io/docs/concepts/storage/volumes#hostpath
<code>type</code>	<code>string</code>	type for HostPath Volume Defaults to "" More info: https://kubernetes.io/docs/concepts/storage/volumes#hostpath

Property	Type	Description
		Possible enum values: <code>""</code> For backwards compatible, leave it empty if unset <code>"BlockDevice"</code> A block device must exist at the given path <code>"CharDevice"</code> A character device must exist at the given path <code>"Directory"</code> A directory must exist at the given path <code>"DirectoryOrCreate"</code> If nothing exists at the given path, an empty directory will be created there as needed with file mode 0755, having the same group and ownership with Kubelet. <code>"File"</code> A file must exist at the given path <code>"FileOrCreate"</code> If nothing exists at the given path, an empty file will be created there as needed with file mode 0644, having the same group and ownership with Kubelet. <code>"Socket"</code> A UNIX socket must exist at the given path

.spec.iscsi

Description

ISCSIPersistentVolumeSource represents an iSCSI disk. iSCSI volumes can only be mounted as read/write once. iSCSI volumes support ownership management and SELinux relabeling.

Type

object

Required

targetPortal

iqn

lun

Property	Type	Description
<code>chapAuthDiscovery</code>	<code>boolean</code>	chapAuthDiscovery defines whether support iSCSI Discovery CHAP authentication
<code>chapAuthSession</code>	<code>boolean</code>	chapAuthSession defines whether support iSCSI Session CHAP authentication
<code>fsType</code>	<code>string</code>	fsType is the filesystem type of the volume that you want to mount. Tip: Ensure that the filesystem type is supported by the host operating system. Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified. More info: https://kubernetes.io/docs/concepts/storage/volumes#iscsi
<code>initiatorName</code>	<code>string</code>	initiatorName is the custom iSCSI Initiator Name. If initiatorName is specified with iscsiInterface simultaneously, new iSCSI interface : will be created for the connection.
<code>iqn</code>	<code>string</code>	iqn is Target iSCSI Qualified Name.
<code>iscsiInterface</code>	<code>string</code>	iscsiInterface is the interface Name that uses an iSCSI transport. Defaults to 'default' (tcp).
<code>lun</code>	<code>integer</code>	lun is iSCSI Target Lun number.

Property	Type	Description
<code>portals</code>	<code>array</code>	portals is the iSCSI Target Portal List. The Portal is either an IP or ip_addr:port if the port is other than default (typically TCP ports 860 and 3260).
<code>readOnly</code>	<code>boolean</code>	readOnly here will force the ReadOnly setting in VolumeMounts. Defaults to false.
<code>secretRef</code>	<code>object</code>	SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace
<code>targetPortal</code>	<code>string</code>	targetPortal is iSCSI Target Portal. The Portal is either an IP or ip_addr:port if the port is other than default (typically TCP ports 860 and 3260).

`.spec.iscsi.portals`

Description

portals is the iSCSI Target Portal List. The Portal is either an IP or ip_addr:port if the port is other than default (typically TCP ports 860 and 3260).

Type

`array`

`.spec.iscsi.portals[]`

Type

`string`

.spec.iscsi.secretRef

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

object

Property	Type	Description
name	string	name is unique within a namespace to reference a secret resource.
namespace	string	namespace defines the space within which the secret name must be unique.

.spec.local

Description

Local represents directly-attached storage with node affinity

Type

object

Required

path

Property	Type	Description
fsType	string	fsType is the filesystem type to mount. It applies only when the Path is a block device. Must be a filesystem type supported by the

Property	Type	Description
		host operating system. Ex. "ext4", "xfs", "ntfs". The default value is to auto-select a filesystem if unspecified.
<code>path</code>	<code>string</code>	path of the full path to the volume on the node. It can be either a directory or block device (disk, partition, ...).

`.spec.mountOptions`

Description

mountOptions is the list of mount options, e.g. ["ro", "soft"]. Not validated - mount will simply fail if one is invalid. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes/#mount-options>

Type

`array`

`.spec.mountOptions[]`

Type

`string`

`.spec.nfs`

Description

Represents an NFS mount that lasts the lifetime of a pod. NFS volumes do not support ownership management or SELinux relabeling.

Type

`object`

Required

`server``path`

Property	Type	Description
<code>path</code>	<code>string</code>	path that is exported by the NFS server. More info: https://kubernetes.io/docs/concepts/storage/volumes#nfs
<code>readOnly</code>	<code>boolean</code>	readOnly here will force the NFS export to be mounted with read-only permissions. Defaults to false. More info: https://kubernetes.io/docs/concepts/storage/volumes#nfs
<code>server</code>	<code>string</code>	server is the hostname or IP address of the NFS server. More info: https://kubernetes.io/docs/concepts/storage/volumes#nfs

.spec.nodeAffinity

Description

VolumeNodeAffinity defines constraints that limit what nodes this volume can be accessed from.

Type

`object`

Property	Type	Description
<code>required</code>	<code>object</code>	A node selector represents the union of the results of one or more label queries over a set of nodes; that is, it represents the OR of the selectors represented by the node selector terms.

.spec.nodeAffinity.required

Description

A node selector represents the union of the results of one or more label queries over a set of nodes; that is, it represents the OR of the selectors represented by the node selector terms.

Type

object

Required

nodeSelectorTerms

Property	Type	Description
nodeSelectorTerms	array	Required. A list of node selector terms. The terms are ORed.

.spec.nodeAffinity.required.nodeSelectorTerms

Description

Required. A list of node selector terms. The terms are ORed.

Type

array

.spec.nodeAffinity.required.nodeSelectorTerms[]

Description

A null or empty node selector term matches no objects. The requirements of them are ANDed. The TopologySelectorTerm type implements a subset of the NodeSelectorTerm.

Type

object

Property	Type	Description
<code>matchExpressions</code>	<code>array</code>	A list of node selector requirements by node's labels.
<code>matchFields</code>	<code>array</code>	A list of node selector requirements by node's fields.

`.spec.nodeAffinity.required.nodeSelectorTerms[].matchExpressions`

Description

A list of node selector requirements by node's labels.

Type

`array`

`.spec.nodeAffinity.required.nodeSelectorTerms[].matchExpressions[]`

Description

A node selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

`object`

Required

`key`

`operator`

Property	Type	Description
<code>key</code>	<code>string</code>	The label key that the selector applies to.

Property	Type	Description
operator	string	Represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists, DoesNotExist. Gt, and Lt. Possible enum values: "DoesNotExist" "Exists" "Gt" "In" "Lt" "NotIn"
values	array	An array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. If the operator is Gt or Lt, the values array must have a single element, which will be interpreted as an integer. This array is replaced during a strategic merge patch.

`.spec.nodeAffinity.required.nodeSelectorTerms[].matchExpressions[].values`

Description

An array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. If the operator is Gt or Lt, the values array must have a single element, which will be interpreted as an integer. This array is replaced during a strategic merge patch.

Type

array

`.spec.nodeAffinity.required.nodeSelectorTerms[].matchExpressions[].values[]`

Type

`string`

`.spec.nodeAffinity.required.nodeSelectorTerms[].matchFields`

Description

A list of node selector requirements by node's fields.

Type

`array`

`.spec.nodeAffinity.required.nodeSelectorTerms[].matchFields[]`

Description

A node selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

`object`

Required

`key``operator`

Property	Type	Description
<code>key</code>	<code>string</code>	The label key that the selector applies to.

Property	Type	Description
<code>operator</code>	<code>string</code>	Represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists, DoesNotExist. Gt, and Lt. Possible enum values: <code>"DoesNotExist"</code> <code>"Exists"</code> <code>"Gt"</code> <code>"In"</code> <code>"Lt"</code> <code>"NotIn"</code>
<code>values</code>	<code>array</code>	An array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. If the operator is Gt or Lt, the values array must have a single element, which will be interpreted as an integer. This array is replaced during a strategic merge patch.

`.spec.nodeAffinity.required.nodeSelectorTerms[].matchFields[].values`

Description

An array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. If the operator is Gt or Lt, the values array must have a single element, which will be interpreted as an integer. This array is replaced during a strategic merge patch.

Type

`array`

`.spec.nodeAffinity.required.nodeSelectorTerms[].matchFields[].values[]`

Type

`string`

`.spec.photonPersistentDisk`

Description

Represents a Photon Controller persistent disk resource.

Type

`object`

Required

`pdID`

Property	Type	Description
<code>fsType</code>	<code>string</code>	fsType is the filesystem type to mount. Must be a filesystem type supported by the host operating system. Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
<code>pdID</code>	<code>string</code>	pdID is the ID that identifies Photon Controller persistent disk

`.spec.portworxVolume`

Description

PortworxVolumeSource represents a Portworx volume resource.

Type

`object`

Required

`volumeID`

Property	Type	Description
<code>fsType</code>	<code>string</code>	fSType represents the filesystem type to mount Must be a filesystem type supported by the host operating system. Ex. "ext4", "xfs". Implicitly inferred to be "ext4" if unspecified.
<code>readOnly</code>	<code>boolean</code>	readOnly defaults to false (read/write). ReadOnly here will force the ReadOnly setting in VolumeMounts.
<code>volumeID</code>	<code>string</code>	volumeID uniquely identifies a Portworx volume

.spec.quobyte

Description

Represents a Quobyte mount that lasts the lifetime of a pod. Quobyte volumes do not support ownership management or SELinux relabeling.

Type

`object`

Required

`registry``volume`

Property	Type	Description
<code>group</code>	<code>string</code>	group to map volume access to Default is no group

Property	Type	Description
<code>readOnly</code>	<code>boolean</code>	<code>readOnly</code> here will force the Quobyte volume to be mounted with read-only permissions. Defaults to false.
<code>registry</code>	<code>string</code>	<code>registry</code> represents a single or multiple Quobyte Registry services specified as a string as host:port pair (multiple entries are separated with commas) which acts as the central registry for volumes
<code>tenant</code>	<code>string</code>	tenant owning the given Quobyte volume in the Backend Used with dynamically provisioned Quobyte volumes, value is set by the plugin
<code>user</code>	<code>string</code>	user to map volume access to Defaults to serviceaccount user
<code>volume</code>	<code>string</code>	volume is a string that references an already created Quobyte volume by name.

.spec.rbd

Description

Represents a Rados Block Device mount that lasts the lifetime of a pod. RBD volumes support ownership management and SELinux relabeling.

Type

`object`

Required

`monitors`

`image`

Property	Type	Description
fsType	string	fsType is the filesystem type of the volume that you want to mount. Tip: Ensure that the filesystem type is supported by the host operating system. Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified. More info: https://kubernetes.io/docs/concepts/storage/volumes#rbd ↗
image	string	image is the rados image name. More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it ↗
keyring	string	keyring is the path to key ring for RBDUser. Default is /etc/ceph/keyring. More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it ↗
monitors	array	monitors is a collection of Ceph monitors. More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it ↗
pool	string	pool is the rados pool name. Default is rbd. More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it ↗

Property	Type	Description
<code>readOnly</code>	<code>boolean</code>	<code>readOnly</code> here will force the <code>ReadOnly</code> setting in <code>VolumeMounts</code> . Defaults to <code>false</code> . More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it ↗
<code>secretRef</code>	<code>object</code>	<code>SecretReference</code> represents a Secret Reference. It has enough information to retrieve secret in any namespace
<code>user</code>	<code>string</code>	<code>user</code> is the rados user name. Default is <code>admin</code> . More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it ↗

`.spec.rbd.monitors`

Description

`monitors` is a collection of Ceph monitors. More info: <https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it>

Type

`array`

`.spec.rbd.monitors[]`

Type

`string`

`.spec.rbd.secretRef`

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

object

Property	Type	Description
name	string	name is unique within a namespace to reference a secret resource.
namespace	string	namespace defines the space within which the secret name must be unique.

.spec.scaleIO

Description

ScaleIOPersistentVolumeSource represents a persistent ScaleIO volume

Type

object

Required

gateway

system

secretRef

Property	Type	Description
fsType	string	fsType is the filesystem type to mount. Must be a filesystem type supported by the host operating system. Ex. "ext4", "xfs", "ntfs". Default is "xfs"

Property	Type	Description
<code>gateway</code>	<code>string</code>	gateway is the host address of the ScaleIO API Gateway.
<code>protectionDomain</code>	<code>string</code>	protectionDomain is the name of the ScaleIO Protection Domain for the configured storage.
<code>readOnly</code>	<code>boolean</code>	readOnly defaults to false (read/write). ReadOnly here will force the ReadOnly setting in VolumeMounts.
<code>secretRef</code>	<code>object</code>	SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace
<code>sslEnabled</code>	<code>boolean</code>	sslEnabled is the flag to enable/disable SSL communication with Gateway, default false
<code>storageMode</code>	<code>string</code>	storageMode indicates whether the storage for a volume should be ThickProvisioned or ThinProvisioned. Default is ThinProvisioned.
<code>storagePool</code>	<code>string</code>	storagePool is the ScaleIO Storage Pool associated with the protection domain.

Property	Type	Description
<code>system</code>	<code>string</code>	system is the name of the storage system as configured in ScaleIO.
<code>volumeName</code>	<code>string</code>	volumeName is the name of a volume already created in the ScaleIO system that is associated with this volume source.

`.spec.scaleIO.secretRef`

Description

SecretReference represents a Secret Reference. It has enough information to retrieve secret in any namespace

Type

`object`

Property	Type	Description
<code>name</code>	<code>string</code>	name is unique within a namespace to reference a secret resource.
<code>namespace</code>	<code>string</code>	namespace defines the space within which the secret name must be unique.

`.spec.storageos`

Description

Represents a StorageOS persistent volume resource.

Type

object

Property	Type	Description
<code>fsType</code>	<code>string</code>	<code>fsType</code> is the filesystem type to mount. Must be a filesystem type supported by the host operating system. Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
<code>readOnly</code>	<code>boolean</code>	<code>readOnly</code> defaults to false (read/write). <code>ReadOnly</code> here will force the <code>ReadOnly</code> setting in <code>VolumeMounts</code> .
<code>secretRef</code>	<code>object</code>	<code>ObjectReference</code> contains enough information to let you inspect or modify the referred object.
<code>volumeName</code>	<code>string</code>	<code>volumeName</code> is the human-readable name of the StorageOS volume. Volume names are only unique within a namespace.
<code>volumeNamespace</code>	<code>string</code>	<code>volumeNamespace</code> specifies the scope of the volume within StorageOS. If no namespace is specified then the Pod's namespace will be used. This allows the Kubernetes name scoping to be mirrored within StorageOS for tighter integration. Set <code>VolumeName</code> to any name to override the default behaviour. Set to "default" if you are not using namespaces within StorageOS. Namespaces that do not pre-exist within StorageOS will be created.

.spec.storageos.secretRef

Description

ObjectReference contains enough information to let you inspect or modify the referred object.

Type

object

Property	Type	Description
<code>apiVersion</code>	<code>string</code>	API version of the referent.
<code>fieldPath</code>	<code>string</code>	If referring to a piece of an object instead of an entire object, this string should contain a valid JSON/Go field access statement, such as <code>desiredState.manifest.containers[2]</code>. For example, if the object reference is to a container within a pod, this would take on a value like: <code>"spec.containers{name}"</code> (where "name" refers to the name of the container that triggered the event) or if no container name is specified <code>"spec.containers[2]"</code> (container with index 2 in this pod). This syntax is chosen only to have some well-defined way of referencing a part of an object.
<code>kind</code>	<code>string</code>	Kind of the referent. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗

Property	Type	Description
<code>name</code>	<code>string</code>	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names ↗
<code>namespace</code>	<code>string</code>	Namespace of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/namespaces/ ↗
<code>resourceVersion</code>	<code>string</code>	Specific resourceVersion to which this reference is made, if any. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#concurrency-control-and-consistency ↗
<code>uid</code>	<code>string</code>	UID of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#uids ↗

.spec.vsphereVolume

Description

Represents a vSphere volume resource.

Type

`object`

Required

`volumePath`

Property	Type	Description
<code>fsType</code>	<code>string</code>	<code>fsType</code> is filesystem type to mount. Must be a filesystem type supported by the host operating system. Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
<code>storagePolicyID</code>	<code>string</code>	<code>storagePolicyID</code> is the storage Policy Based Management (SPBM) profile ID associated with the <code>StoragePolicyName</code> .
<code>storagePolicyName</code>	<code>string</code>	<code>storagePolicyName</code> is the storage Policy Based Management (SPBM) profile name.
<code>volumePath</code>	<code>string</code>	<code>volumePath</code> is the path that identifies vSphere volume vmdk

.status

Description

`PersistentVolumeStatus` is the current status of a persistent volume.

Type

`object`

Property	Type	Description
<code>lastPhaseTransitionTime</code>	<code>string</code>	<code>Time</code> is a wrapper around <code>time.Time</code> which supports correct marshaling to YAML and JSON. Wrappers ar

Property	Type	Description
		provided for many of the factory methods that the <code>tim</code> package offers.
<code>message</code>	<code>string</code>	<code>message</code> is a human-readable message indicating details about why the volume is in this state.
<code>phase</code>	<code>string</code>	<code>phase</code> indicates if a volume is available, bound to a claim, or released by a claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#phase ↗ Possible enum values: • <code>"Available"</code> used for PersistentVolumes that are not yet bound Available volumes are held by the binder and matched to PersistentVolumeClaims • <code>"Bound"</code> used for PersistentVolumes that are bound • <code>"Failed"</code> used for PersistentVolumes that failed to be correctly recycled or deleted after being released from a claim • <code>"Pending"</code> used for PersistentVolumes that are not available • <code>"Released"</code> used for PersistentVolumes where the bound PersistentVolumeClaim was deleted released volumes must be recycled before becoming available again this phase is used by the persistent volume claim binder to signal to another process to reclaim the resource

Property	Type	Description
<code>reason</code>	<code>string</code>	reason is a brief CamelCase string that describes an failure and is meant for machine parsing and tidy display in the CLI.

API Endpoints

The following API endpoints are available:

- `/kubernetes/{cluster}/api/v1/persistentvolumes`
 - `DELETE` : delete collection of PersistentVolume
 - `GET` : list objects of kind PersistentVolume
 - `POST` : create a new PersistentVolume
- `/kubernetes/{cluster}/api/v1/persistentvolumes/{name}`
 - `DELETE` : delete the specified PersistentVolume
 - `GET` : read the specified PersistentVolume
 - `PATCH` : partially update the specified PersistentVolume
 - `PUT` : replace the specified PersistentVolume
- `/kubernetes/{cluster}/api/v1/persistentvolumes/{name}/status`
 - `GET` : read status of the specified PersistentVolume
 - `PATCH` : partially update status of the specified PersistentVolume
 - `PUT` : replace status of the specified PersistentVolume

`/kubernetes/{cluster}/api/v1/persistentvolumes`

HTTP method

`DELETE`

Description

delete collection of PersistentVolume

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

GET

Description

list objects of kind PersistentVolume

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolumeList</code> schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a new PersistentVolume

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Parameter	Type	Description
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+. - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>PersistentVolume</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolume</code> schema
201 - Created	<code>PersistentVolume</code> schema
202 - Accepted	<code>PersistentVolume</code> schema
401 - Unauthorized	Empty

`/kubernetes/{cluster}/api/v1/persistentvolumes/{name}`

HTTP method

DELETE

Description

delete the specified PersistentVolume

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
202 - Accepted	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

GET

Description

read the specified PersistentVolume

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolume</code> schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update the specified PersistentVolume

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolume</code> schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified PersistentVolume

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>PersistentVolume</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolume</code> schema
201 - Created	<code>PersistentVolume</code> schema
401 - Unauthorized	Empty

/kubernetes/{cluster}/api/v1/persistentvolumes/{name}/status

HTTP method

`GET`

Description

read status of the specified PersistentVolume

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolume</code> schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update status of the specified PersistentVolume

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing

Parameter	Type	Description
		of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolume</code> schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace status of the specified PersistentVolume

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized <code>dryRun</code> directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a <code>BadRequest</code> error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>PersistentVolume</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>PersistentVolume</code> schema
201 - Created	<code>PersistentVolume</code> schema
401 - Unauthorized	Empty

StorageClass [storage.k8s.io/v1]

Description

StorageClass describes the parameters for a class of storage for which PersistentVolumes can be dynamically provisioned. StorageClasses are non-namespaced; the name of the storage class according to etcd is in ObjectMeta.Name.

Type

`object`

Required

`provisioner`

Specification

Property	Type	Description
<code>allowVolumeExpansion</code>	<code>boolean</code>	<code>allowVolumeExpansion</code> shows whether the storage class allow volume expand.
<code>allowedTopologies</code>	<code>array</code>	<code>allowedTopologies</code> restrict the node topologies where volumes can be dynamically provisioned. Each volume plugin defines its own supported topology specifications. An empty <code>TopologySelectorTerm</code> list means there is no topology restriction. This field is only honored by

Property	Type	Description
		servers that enable the VolumeScheduling feature.
apiVersion	string	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
kind	string	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds
metadata	ObjectMeta	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
mountOptions	array	mountOptions controls the mountOptions for dynamically provisioned PersistentVolumes of this storage class. e.g. ["ro", "soft"]. Not validated - mount of the PVs will simply fail if one is invalid.

Property	Type	Description
<code>parameters</code>	<code>object</code>	<code>parameters</code> holds the parameters for the provisioner that should create volumes of this storage class.
<code>provisioner</code>	<code>string</code>	<code>provisioner</code> indicates the type of the provisioner.
<code>reclaimPolicy</code>	<code>string</code>	<code>reclaimPolicy</code> controls the <code>reclaimPolicy</code> for dynamically provisioned <code>PersistentVolumes</code> of this storage class. Defaults to <code>Delete</code>. Possible enum values: <code>"Delete"</code> means the volume will be deleted from Kubernetes on release from its claim. The volume plugin must support <code>Deletion</code>. <code>"Recycle"</code> means the volume will be recycled back into the pool of unbound persistent volumes on release from its claim. The volume plugin must support <code>Recycling</code>. <code>"Retain"</code> means the volume will be left in its current phase (<code>Released</code>) for manual reclamation by the administrator. The default policy is <code>Retain</code>.
<code>volumeBindingMode</code>	<code>string</code>	<code>volumeBindingMode</code> indicates how <code>PersistentVolumeClaims</code> should be provisioned and bound. When unset, <code>VolumeBindingImmediate</code> is used. This field is only honored by servers that enable the <code>VolumeScheduling</code> feature.

Property	Type	Description
		Possible enum values: <code>"Immediate"</code> indicates that PersistentVolumeClaims should be immediately provisioned and bound. This is the default mode. <code>"WaitForFirstConsumer"</code> indicates that PersistentVolumeClaims should not be provisioned and bound until the first Pod is created that references the PersistentVolumeClaim. The volume provisioning and binding will occur during Pod scheduling.

.allowedTopologies

Description

allowedTopologies restrict the node topologies where volumes can be dynamically provisioned. Each volume plugin defines its own supported topology specifications. An empty TopologySelectorTerm list means there is no topology restriction. This field is only honored by servers that enable the VolumeScheduling feature.

Type

array

.allowedTopologies[]

Description

A topology selector term represents the result of label queries. A null or empty topology selector term matches no objects. The requirements of them are ANDed. It provides a subset of functionality as NodeSelectorTerm. This is an alpha feature and may change in the future.

Type

`object`

Property	Type	Description
<code>matchLabelExpressions</code>	<code>array</code>	A list of topology selector requirements by labels.

`.allowedTopologies[].matchLabelExpressions`

Description

A list of topology selector requirements by labels.

Type

`array`

`.allowedTopologies[].matchLabelExpressions[]`

Description

A topology selector requirement is a selector that matches given label. This is an alpha feature and may change in the future.

Type

`object`

Required

`key``values`

Property	Type	Description
<code>key</code>	<code>string</code>	The label key that the selector applies to.
<code>values</code>	<code>array</code>	An array of string values. One value must match the label to be selected. Each entry in Values is ORed.

.allowedTopologies[].matchLabelExpressions[].values

Description

An array of string values. One value must match the label to be selected. Each entry in Values is ORed.

Type

array

.allowedTopologies[].matchLabelExpressions[].values[]

Type

string

.mountOptions

Description

mountOptions controls the mountOptions for dynamically provisioned PersistentVolumes of this storage class. e.g. ["ro", "soft"]. Not validated - mount of the PVs will simply fail if one is invalid.

Type

array

.mountOptions[]

Type

string

.parameters

Description

parameters holds the parameters for the provisioner that should create volumes of this storage class.

Type

object

API Endpoints

The following API endpoints are available:

- `/kubernetes/{cluster}/apis/storage.k8s.io/v1/storageclasses`
 - **DELETE** : delete collection of StorageClass
 - **GET** : list objects of kind StorageClass
 - **POST** : create a new StorageClass
- `/kubernetes/{cluster}/apis/storage.k8s.io/v1/storageclasses/{name}`
 - **DELETE** : delete the specified StorageClass
 - **GET** : read the specified StorageClass
 - **PATCH** : partially update the specified StorageClass
 - **PUT** : replace the specified StorageClass

/kubernetes/{cluster}/apis/storage.k8s.io/v1/storageclasses

HTTP method

DELETE

Description

delete collection of StorageClass

HTTP responses

HTTP code	Response body
200 - OK	Status schema

HTTP code	Response body
401 - Unauthorized	Empty

HTTP method

GET

Description

list objects of kind StorageClass

HTTP responses

HTTP code	Response body
200 - OK	<code>StorageClassList</code> schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a new StorageClass

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a

Parameter	Type	Description
		warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
body	StorageClass schema	application/json formatted

HTTP responses

HTTP code	Response body
200 - OK	StorageClass schema
201 - Created	StorageClass schema
202 - Accepted	StorageClass schema
401 - Unauthorized	Empty

/kubernetes/{cluster}/apis/storage.k8s.io/v1/storageclasses/{name}

HTTP method

DELETE

Description

delete the specified StorageClass

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
202 - Accepted	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

read the specified StorageClass

HTTP responses

HTTP code	Response body
200 - OK	<code>StorageClass</code> schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update the specified StorageClass

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized <code>dryRun</code> directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>StorageClass</code> schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace the specified `StorageClass`

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized <code>dryRun</code> directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a <code>BadRequest</code> error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>StorageClass</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>StorageClass</code> schema
201 - Created	<code>StorageClass</code> schema
401 - Unauthorized	Empty

Ceph Storage APIs

[CephCluster \[cephclusters.ceph\]](#) [CephFilesystem \[cephfilesystem.ceph\]](#) [CephBlockI](#)

[CephObjectStore \[cephobjects.ceph\]](#) [CephObjectStoreUser \[cephobjectstoreusers.ceph\]](#)

CephCluster [cephclusters.ceph.rook.io/v1]

Description

CephCluster is a Ceph storage cluster

Type

object

Required

metadata

spec

Specification

Property	Type	Description
apiVersion	string	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources ↗
kind	string	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds

Property	Type	Description
		https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
metadata	ObjectMeta	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
spec	object	ClusterSpec represents the specification of Ceph Cluster
status	object	ClusterStatus represents the status of a Ceph cluster

.spec

Description

ClusterSpec represents the specification of Ceph Cluster

Type

object

Property	Type	Description
annotations	object	The annotations-related conf each Pod related object.
cephConfig	object	Ceph Config options

Property	Type	Description
<code>cephConfigFromSecret</code>	<code>object</code>	CephConfigFromSecret works like CephConfig but takes config reference.
<code>cephVersion</code>	<code>object</code>	The version information that orchestrator uses to orchestrate a particular version.
<code>cleanupPolicy</code>	<code>object</code>	Indicates user intent when deleting a cluster during orchestration and should not delete if deletion is not imminent.
<code>continueUpgradeAfterChecksEvenIfNotHealthy</code>	<code>boolean</code>	ContinueUpgradeAfterChecksEvenIfNotHealthy defines if an upgrade should continue even if the cluster is not clean.
<code>crashCollector</code>	<code>object</code>	A spec for the crash controller.
<code>csi</code>	<code>object</code>	CSI Driver Options applied to the storage.
<code>dashboard</code>	<code>object</code>	Dashboard settings.
<code>dataDirHostPath</code>	<code>string</code>	The path on the host where ceph data is persisted.

Property	Type	Description
disruptionManagement	object	A spec for configuring disruption
external	object	Whether the Ceph Cluster is Kubernetes cluster mon, mgr discover daemons will not be clusters.
healthCheck	object	Internal daemon healthcheck
labels	object	The labels-related configurat Pod related object.
logCollector	object	Logging represents loggings
mgr	object	A spec for mgr related option
mon	object	A spec for mon related option
monitoring	object	Prometheus based Monitorin
network	object	Network related configurator

Property	Type	Description
placement	object	
priorityClassNames	object	PriorityClassNames sets prio components
removeOSDsIfOutAndSafeToRemove	boolean	Remove the OSD that is out only if this option is true
resources	object	Resources set resource requ
security	object	Security represents security :
skipUpgradeChecks	boolean	SkipUpgradeChecks defines be forced even if one of the c
storage	object	A spec for available storage i it should be used
upgradeOSDRequiresHealthyPGs	boolean	UpgradeOSDRequiresHealthr upgrade requires PGs are clk OSD upgrade process won't healthy. This configuration wi skipUpgradeChecks is tr

Property	Type	Description
		WaitTimeoutForHealthyOSDInMinutes
waitTimeoutForHealthyOSDInMinutes	integer	continueUpgradeAfterChecksOnlyIfHealthyOSDsExceeds is false . If continueUpgradeAfterChecksOnlyIfHealthyOSDsExceeds is true , then operator would upgrade of an OSD even if it exceeds the timeout. This timeout works only if skipUpgradeChecks is true . If skipUpgradeChecks is false , the timeout is 10 minutes.

.spec.annotations

Description

The annotations-related configuration to add/set on each Pod related object.

Type

object

.spec.cephConfig

Description

Ceph Config options

Type

object

.spec.cephConfigFromSecret

Description

CephConfigFromSecret works exactly like CephConfig but takes config value from Secret Key reference.

Type

object

.spec.cephVersion

Description

The version information that instructs Rook to orchestrate a particular version of Ceph.

Type

object

Property	Type	Description
<code>allowUnsupported</code>	<code>boolean</code>	Whether to allow unsupported versions (do not set to true in production)
<code>image</code>	<code>string</code>	Image is the container image used to launch the ceph daemons, such as quay.io/ceph/ceph: The full list of images can be found at https://quay.io/repository/ceph/ceph?tab=tags
<code>imagePullPolicy</code>	<code>string</code>	ImagePullPolicy describes a policy for if/when to pull a container image One of Always, Never, IfNotPresent.

.spec.cleanupPolicy

Description

Indicates user intent when deleting a cluster; blocks orchestration and should not be set if cluster deletion is not imminent.

Type

object

Property	Type	Description
<code>allowUninstallWithVolumes</code>	<code>boolean</code>	AllowUninstallWithVolumes defines whether we can proceed with the uninstall if they are RBD images still present
<code>confirmation</code>	<code>string</code>	Confirmation represents the cleanup confirmation
<code>sanitizeDisks</code>	<code>object</code>	SanitizeDisks represents way we sanitize disks
<code>wipeDevicesFromOtherClusters</code>	<code>boolean</code>	WipeDevicesFromOtherClusters wipes the OSD disks belonging to other clusters. This is useful in scenarios where ceph cluster was reinstalled but OSD disk still contains the metadata from previous ceph cluster.

.spec.cleanupPolicy.sanitizeDisks

Description

SanitizeDisks represents way we sanitize disks

Type

object

Property	Type	Description
dataSource	string	DataSource is the data source to use to sanitize the disk with
iteration	integer	Iteration is the number of pass to apply the sanitizing
method	string	Method is the method we use to sanitize disks

.spec.crashCollector

Description

A spec for the crash controller

Type

object

Property	Type	Description
daysToRetain	integer	DaysToRetain represents the number of days to retain crash until they get pruned
disable	boolean	Disable determines whether we should enable the crash collector

.spec.csi

Description

CSI Driver Options applied per cluster.

Type

object

Property	Type	Description
<code>cephfs</code>	object	CephFS defines CSI Driver settings for CephFS driver.
<code>readAffinity</code>	object	ReadAffinity defines the read affinity settings for CSI driver.
<code>skipUserCreation</code>	boolean	SkipUserCreation determines whether CSI users and their associated secrets should be skipped. If set to true, the user must manually manage these secrets.

.spec.csi.cephfs

Description

CephFS defines CSI Driver settings for CephFS driver.

Type

object

Property	Type	Description
<code>fuseMountOptions</code>	string	FuseMountOptions defines the mount options for ceph fuse mounter.

Property	Type	Description
<code>kernelMountOptions</code>	<code>string</code>	KernelMountOptions defines the mount options for kernel mounter.

`.spec.csi.readAffinity`

Description

ReadAffinity defines the read affinity settings for CSI driver.

Type

`object`

Property	Type	Description
<code>crushLocationLabels</code>	<code>array</code>	CrushLocationLabels defines which node labels to use as CRUSH location. This should correspond to the values set in the CRUSH map.
<code>enabled</code>	<code>boolean</code>	Enables read affinity for CSI driver.

`.spec.csi.readAffinity.crushLocationLabels`

Description

CrushLocationLabels defines which node labels to use as CRUSH location. This should correspond to the values set in the CRUSH map.

Type

`array`

`.spec.csi.readAffinity.crushLocationLabels[]`

Type

string

.spec.dashboard

Description

Dashboard settings

Type

object

Property	Type	Description
enabled	boolean	Enabled determines whether to enable the dashboard
port	integer	Port is the dashboard webserver port
prometheusEndpoint	string	Endpoint for the Prometheus host
prometheusEndpointSSLVerify	boolean	Whether to verify the ssl endpoint for prometheus. Set to false for a self-signed cert.
ssl	boolean	SSL determines whether SSL should be used

Property	Type	Description
<code>urlPrefix</code>	<code>string</code>	URLPrefix is a prefix for all URLs to use the dashboard with a reverse proxy

.spec.disruptionManagement

Description

A spec for configuring disruption management.

Type

`object`

Property	Type	Description
<code>machineDisruptionBudgetNamespace</code>	<code>string</code>	Deprecated. Namespace to look for MDE
<code>manageMachineDisruptionBudgets</code>	<code>boolean</code>	Deprecated. This enables management of
<code>managePodBudgets</code>	<code>boolean</code>	This enables management of poddisrupti
<code>osdMaintenanceTimeout</code>	<code>integer</code>	OSDMaintenanceTimeout sets how man for drained failure domains it only works minutes
<code>pgHealthCheckTimeout</code>	<code>integer</code>	DEPRECATED: PGHealthCheckTimeout

Property	Type	Description
<code>pgHealthyRegex</code>	<code>string</code>	PgHealthyRegex is the regular expression that should be considered healthy. The default value is <code>^(active\+clean active\+clean\+scrub)\$</code>

.spec.external

Description

Whether the Ceph Cluster is running external to this Kubernetes cluster mon, mgr, osd, mds, and discover daemons will not be created for external clusters.

Type

`object`

Property	Type	Description
<code>enable</code>	<code>boolean</code>	Enable determines whether external mode is enabled or not

.spec.healthCheck

Description

Internal daemon healthchecks and liveness probe

Type

`object`

Property	Type	Description
<code>daemonHealth</code>	<code>object</code>	DaemonHealth is the health check for a given daemon

Property	Type	Description
<code>livenessProbe</code>	<code>object</code>	LivenessProbe allows changing the livenessProbe configuration for a given daemon
<code>startupProbe</code>	<code>object</code>	StartupProbe allows changing the startupProbe configuration for a given daemon

`.spec.healthCheck.daemonHealth`

Description

DaemonHealth is the health check for a given daemon

Type

`object`

Property	Type	Description
<code>mon</code>	<code>object</code>	Monitor represents the health check settings for the Ceph monitor
<code>osd</code>	<code>object</code>	ObjectStorageDaemon represents the health check settings for the Ceph OSDs
<code>status</code>	<code>object</code>	Status represents the health check settings for the Ceph health

`.spec.healthCheck.daemonHealth.mon`

Description

Monitor represents the health check settings for the Ceph monitor

Type

object

Property	Type	Description
disabled	boolean	
interval	string	Interval is the internal in second or minute for the health check to run like 60s for 60 seconds
timeout	string	

.spec.healthCheck.daemonHealth.osd

Description

ObjectStorageDaemon represents the health check settings for the Ceph OSDs

Type

object

Property	Type	Description
disabled	boolean	
interval	string	Interval is the internal in second or minute for the health check to run like 60s for 60 seconds
timeout	string	

.spec.healthCheck.daemonHealth.status

Description

Status represents the health check settings for the Ceph health

Type

object

Property	Type	Description
disabled	boolean	
interval	string	Interval is the internal in second or minute for the health check to run like 60s for 60 seconds
timeout	string	

.spec.healthCheck.livenessProbe

Description

LivenessProbe allows changing the livenessProbe configuration for a given daemon

Type

object

.spec.healthCheck.startupProbe

Description

StartupProbe allows changing the startupProbe configuration for a given daemon

Type

object

.spec.labels

Description

The labels-related configuration to add/set on each Pod related object.

Type

`object`

.spec.logCollector

Description

Logging represents loggings settings

Type

`object`

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled represents whether the log collector is enabled
<code>maxLogSize</code>		MaxLogSize is the maximum size of the log per ceph daemons. Must be at least 1M.
<code>periodicity</code>	<code>string</code>	Periodicity is the periodicity of the log rotation.

.spec.mgr

Description

A spec for mgr related options

Type

`object`

Property	Type	Description
<code>allowMultiplePerNode</code>	<code>boolean</code>	AllowMultiplePerNode allows to run multiple managers on the same node (not recommended)
<code>count</code>	<code>integer</code>	Count is the number of manager daemons to run
<code>hostNetwork</code>	<code>boolean</code>	Whether host networking is enabled for the Ceph Mgr. If not set, the network settings from CephCluster.spec.networking will be applied.
<code>modules</code>	<code>array</code>	Modules is the list of ceph manager modules to enable/disable

`.spec.mgr.modules`

Description

Modules is the list of ceph manager modules to enable/disable

Type

`array`

`.spec.mgr.modules[]`

Description

Module represents mgr modules that the user wants to enable or disable

Type

`object`

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled determines whether a module should be enabled or not
<code>name</code>	<code>string</code>	Name is the name of the ceph manager module
<code>settings</code>	<code>object</code>	Settings to further configure the module

`.spec.mgr.modules[].settings`

Description

Settings to further configure the module

Type

`object`

Property	Type	Description
<code>balancerMode</code>	<code>string</code>	BalancerMode sets the <code>balancer</code> module with different modes like <code>upmap</code> , <code>crush-compact</code> etc

`.spec.mon`

Description

A spec for mon related options

Type

`object`

Property	Type	Description
<code>allowMultiplePerNode</code>	<code>boolean</code>	AllowMultiplePerNode determines if we can run multiple monitors on the same node (not recommended)
<code>count</code>	<code>integer</code>	Count is the number of Ceph monitors
<code>externalMonIDs</code>	<code>array</code>	ExternalMonIDs - optional list of monitor IDs which are deployed externally and not managed by Rook. If set, Rook will not remove mons with given IDs from quorum. This parameter is used only for local Rook cluster running in normal mode and will be ignored if external or stretched mode is used. leading
<code>failureDomainLabel</code>	<code>string</code>	
<code>stretchCluster</code>	<code>object</code>	StretchCluster is the stretch cluster specification
<code>volumeClaimTemplate</code>	<code>object</code>	VolumeClaimTemplate is the PVC definition
<code>zones</code>	<code>array</code>	Zones are specified when we want to provide zonal awareness to mons

.spec.mon.externalMonIDs

Description

ExternalMonIDs - optional list of monitor IDs which are deployed externally and not managed by Rook. If set, Rook will not remove mons with given IDs from quorum. This parameter is used only for local Rook cluster running in normal mode and will be ignored if external or stretched mode is used. leading

Type

array

.spec.mon.externalMonIDs[]

Type

string

.spec.mon.stretchCluster

Description

StretchCluster is the stretch cluster specification

Type

object

Property	Type	Description
failureDomainLabel	string	FailureDomainLabel the failure domain name (e,g: zone)
subFailureDomain	string	SubFailureDomain is the failure domain within a zone
zones	array	Zones is the list of zones

`.spec.mon.stretchCluster.zones`

Description

Zones is the list of zones

Type

array

`.spec.mon.stretchCluster.zones[]`

Description

MonZoneSpec represents the specification of a zone in a Ceph Cluster

Type

object

Property	Type	Description
<code>arbiter</code>	<code>boolean</code>	Arbiter determines if the zone contains the arbiter used for stretch cluster mode
<code>name</code>	<code>string</code>	Name is the name of the zone
<code>volumeClaimTemplate</code>	<code>object</code>	VolumeClaimTemplate is the PVC template

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate`

Description

VolumeClaimTemplate is the PVC template

Type

object

Property	Type	Description
metadata	ObjectMeta	Standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata
spec	object	spec defines the desired characteristics of a volume requested by a pod author. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec`

Description

spec defines the desired characteristics of a volume requested by a pod author. More info:
<https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims>

Type

object

Property	Type	Description
accessModes	array	accessModes contains the desired access modes volume should have. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1
dataSource	object	dataSource field can be used to specify either:

Property	Type	Description
		• An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) • An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.
dataSourceRef	object	dataSourceRef specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a PersistentVolumeClaim object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will require the functionality of the dataSource field and as such both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in dataSourceRef, both fields (dataSource and dataSourceRef) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in dataSourceRef, dataSource isn't set to the same value and must be empty. There are three important differences between dataSource and dataSourceRef

Property	Type	Description
		- While dataSource only allows two specific type objects, dataSourceRef allows any non-core objects as well as PersistentVolumeClaim objects. - While dataSource ignores disallowed values (dropping them), dataSourceRef preserves all values, and generates an error if a disallowed value is specified. - While dataSource only allows local objects, dataSourceRef allows objects in any namespace (Beta) Using this field requires the AnyVolumeDataSource feature gate to be enabled (Alpha) Using the namespace field of dataSourceRef requires the CrossNamespaceVolumeDataSource feature gate to be enabled.
resources	object	resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources
selector	object	selector is a label query over volumes to consider for binding.

Property	Type	Description
<code>storageClassName</code>	<code>string</code>	<code>storageClassName</code> is the name of the <code>StorageClass</code> required by the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#class-1 ↗
<code>volumeAttributesClassName</code>	<code>string</code>	<code>volumeAttributesClassName</code> may be used to set the <code>VolumeAttributesClass</code> used by this claim. If specified, the CSI driver will create or update the volume with attributes defined in the corresponding <code>VolumeAttributesClass</code>. This has a different purpose than <code>storageClassName</code>, it can be changed after the claim is created. An empty string or nil value indicates that no <code>VolumeAttributesClass</code> will be applied to the claim. If the claim enters an Infeasible error state, the field can be reset to its previous value (including nil) to cancel the modification. If the resource referred to by <code>volumeAttributesClassName</code> does not exist, this <code>PersistentVolumeClaim</code> will be set to a Pending state as reflected by the <code>modifyVolumeStatus</code> field, until such as a resource exists. More info: https://kubernetes.io/docs/concepts/storage/volume-attributes-classes/ ↗
<code>volumeMode</code>	<code>string</code>	<code>volumeMode</code> defines what type of volume is required by the claim. Value of <code>Filesystem</code> is implied when not included in claim spec.
<code>volumeName</code>	<code>string</code>	<code>volumeName</code> is the binding reference to the <code>PersistentVolume</code> backing this claim.

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.accessModes`

Description

accessModes contains the desired access modes the volume should have. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1>

Type

array

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.accessModes[]`

Type

string

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.dataSource`

Description

dataSource field can be used to specify either: * An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) * An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.

Type

object

Required

kind

name

Property	Type	Description
<code>apiGroup</code>	<code>string</code>	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
<code>kind</code>	<code>string</code>	Kind is the type of resource being referenced
<code>name</code>	<code>string</code>	Name is the name of resource being referenced

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.dataSourceRef`

Description

`dataSourceRef` specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a `PersistentVolumeClaim` object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will replace the functionality of the `dataSource` field and as such if both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in `dataSourceRef`, both fields (`dataSource` and `dataSourceRef`) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in `dataSourceRef`, `dataSource` isn't set to the same value and must be empty. There are three important differences between `dataSource` and `dataSourceRef`:

- * While `dataSource` only allows two specific types of objects, `dataSourceRef` allows any non-core object, as well as `PersistentVolumeClaim` objects.
- * While `dataSource` ignores disallowed values (dropping them), `dataSourceRef` preserves all values, and generates an error if a disallowed value is specified.
- * While `dataSource` only allows local objects, `dataSourceRef` allows objects in any namespaces.

(Beta) Using this field requires the `AnyVolumeDataSource` feature gate to be enabled.
 (Alpha) Using the namespace field of `dataSourceRef` requires the `CrossNamespaceVolumeDataSource` feature gate to be enabled.

Type

object

Required

kind

name

Property	Type	Description
apiGroup	string	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	string	Kind is the type of resource being referenced
name	string	Name is the name of resource being referenced
namespace	string	Namespace is the namespace of resource being referenced Note that when a namespace is specified, a gateway.networking.k8s.io/ReferenceGrant object is required in the referent namespace to allow that namespace's owner to accept the reference. See the ReferenceGrant documentation for details. (Alpha) This field requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.resources`

Description

resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info:

<https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources>

Type

object

Property	Type	Description
limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.resources.limits`

Description

Limits describes the maximum amount of compute resources allowed. More info:
<https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.resources.requests`

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info:

<https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.selector`

Description

selector is a label query over volumes to consider for binding.

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	matchExpressions is a list of label selector requirements. The requirements are ANDed.
<code>matchLabels</code>	object	matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.selector.matchExpressions`

Description

`matchExpressions` is a list of label selector requirements. The requirements are ANDed.

Type

array

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.selector.matchExpressions[]`

Description

A label selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

object

Required

key

operator

Property	Type	Description
<code>key</code>	<code>string</code>	key is the label key that the selector applies to.
<code>operator</code>	<code>string</code>	operator represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists and DoesNotExist.

Property	Type	Description
<code>values</code>	<code>array</code>	<code>values</code> is an array of string values. If the operator is <code>In</code> or <code>NotIn</code> , the values array must be non-empty. If the operator is <code>Exists</code> or <code>DoesNotExist</code> , the values array must be empty. This array is replaced during a strategic merge patch.

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.selector.matchExpressions[].values`

Description

`values` is an array of string values. If the operator is `In` or `NotIn`, the values array must be non-empty. If the operator is `Exists` or `DoesNotExist`, the values array must be empty. This array is replaced during a strategic merge patch.

Type

`array`

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.selector.matchExpressions[].values[]`

Type

`string`

`.spec.mon.stretchCluster.zones[].volumeClaimTemplate.spec.selector.matchLabels`

Description

`matchLabels` is a map of {key,value} pairs. A single {key,value} in the `matchLabels` map is equivalent to an element of `matchExpressions`, whose key field is "key", the operator is `In`, and the values array contains only "value". The requirements are ANDed.

Type

object

.spec.mon.volumeClaimTemplate

Description

VolumeClaimTemplate is the PVC definition

Type

object

Property	Type	Description
metadata	ObjectMeta	Standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata ↗
spec	object	spec defines the desired characteristics of a volume requested by a pod author. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims ↗

.spec.mon.volumeClaimTemplate.spec

Description

spec defines the desired characteristics of a volume requested by a pod author. More info:
<https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims>

Type

object

Property	Type	Description
<code>accessModes</code>	<code>array</code>	accessModes contains the desired access modes volume should have. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1
<code>dataSource</code>	<code>object</code>	dataSource field can be used to specify either: • An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) • An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.
<code>dataSourceRef</code>	<code>object</code>	dataSourceRef specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a PersistentVolumeClaim object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will require the functionality of the dataSource field and as such both fields are non-empty, they must have the same

Property	Type	Description
		value. For backwards compatibility, when namespace isn't specified in dataSourceRef, both fields (dataSource and dataSourceRef) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in dataSourceRef, dataSource isn't set to the same value and must be empty. There are three important differences between dataSource and dataSourceRef: • While dataSource only allows two specific type objects, dataSourceRef allows any non-core object as well as PersistentVolumeClaim objects. • While dataSource ignores disallowed values (dropping them), dataSourceRef preserves all values, and generates an error if a disallowed value is specified. • While dataSource only allows local objects, dataSourceRef allows objects in any namespace (Beta) Using this field requires the AnyVolumeDataSource feature gate to be enabled (Alpha) Using the namespace field of dataSourceRef requires the CrossNamespaceVolumeDataSource feature gate to be enabled.
resources	object	resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources

Property	Type	Description
<code>selector</code>	<code>object</code>	selector is a label query over volumes to consider binding.
<code>storageClassName</code>	<code>string</code>	storageClassName is the name of the StorageClass required by the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#class-1 ↗
<code>volumeAttributesClassName</code>	<code>string</code>	volumeAttributesClassName may be used to set the VolumeAttributesClass used by this claim. If specified, the CSI driver will create or update the volume with attributes defined in the corresponding VolumeAttributesClass. This has a different purpose than storageClassName, it can be changed after the claim is created. An empty string or nil value indicates that no VolumeAttributesClass will be applied to the claim. If the claim enters an Infeasible error state, the field can be reset to its previous value (including nil) to cancel the modification. If the resource referred to by volumeAttributesClass does not exist, this PersistentVolumeClaim will be set to a Pending state as reflected by the modifyVolumeStatus field, until such a resource exists. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes/attributes-classes/ ↗
<code>volumeMode</code>	<code>string</code>	volumeMode defines what type of volume is required by the claim. Value of Filesystem is implied when not included in claim spec.

Property	Type	Description
<code>volumeName</code>	<code>string</code>	volumeName is the binding reference to the PersistentVolume backing this claim.

`.spec.mon.volumeClaimTemplate.spec.accessModes`

Description

accessModes contains the desired access modes the volume should have. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1>

Type

`array`

`.spec.mon.volumeClaimTemplate.spec.accessModes[]`

Type

`string`

`.spec.mon.volumeClaimTemplate.spec.dataSource`

Description

dataSource field can be used to specify either: * An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) * An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.

Type

`object`

Required

kind

name

Property	Type	Description
apiGroup	string	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	string	Kind is the type of resource being referenced
name	string	Name is the name of resource being referenced

.spec.mon.volumeClaimTemplate.spec.dataSourceRef

Description

dataSourceRef specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a PersistentVolumeClaim object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will replace the functionality of the dataSource field and as such if both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in dataSourceRef, both fields (dataSource and dataSourceRef) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in dataSourceRef, dataSource isn't set to the same value and must be empty. There are three important differences between dataSource and dataSourceRef: * While dataSource only allows two specific types of objects, dataSourceRef allows any non-core object, as well as PersistentVolumeClaim objects. * While dataSource ignores disallowed values (dropping them), dataSourceRef preserves all values, and generates an error if a disallowed value is specified. * While dataSource only allows local objects, dataSourceRef allows objects in any namespaces. (Beta) Using this field requires the AnyVolumeDataSource feature gate to be enabled.

(Alpha) Using the namespace field of dataSourceRef requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

Type

object

Required

kind name

Property	Type	Description
apiGroup	string	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	string	Kind is the type of resource being referenced
name	string	Name is the name of resource being referenced
namespace	string	Namespace is the namespace of resource being referenced Note that when a namespace is specified, a gateway.networking.k8s.io/ReferenceGrant object is required in the referent namespace to allow that namespace's owner to accept the reference. See the ReferenceGrant documentation for details. (Alpha) This field requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

.spec.mon.volumeClaimTemplate.spec.resources

Description

resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info:

<https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources>

Type

object

Property	Type	Description
<code>limits</code>	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
<code>requests</code>	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

`.spec.mon.volumeClaimTemplate.spec.resources.limits`

Description

Limits describes the maximum amount of compute resources allowed. More info:
<https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

.spec.mon.volumeClaimTemplate.spec.resources.request

S

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info:

<https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

.spec.mon.volumeClaimTemplate.spec.selector

Description

selector is a label query over volumes to consider for binding.

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	matchExpressions is a list of label selector requirements. The requirements are ANDed.
<code>matchLabels</code>	object	matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

.spec.mon.volumeClaimTemplate.spec.selector.matchExpressions

Description

matchExpressions is a list of label selector requirements. The requirements are ANDed.

Type

array

.spec.mon.volumeClaimTemplate.spec.selector.matchExpressions[]

Description

A label selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

object

Required

key operator

Property	Type	Description
key	string	key is the label key that the selector applies to.
operator	string	operator represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists and DoesNotExist.

Property	Type	Description
<code>values</code>	<code>array</code>	values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

`.spec.mon.volumeClaimTemplate.spec.selector.matchExpressions[].values`

Description

values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

Type

`array`

`.spec.mon.volumeClaimTemplate.spec.selector.matchExpressions[].values[]`

Type

`string`

`.spec.mon.volumeClaimTemplate.spec.selector.matchLabels`

Description

matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

Type

`object`

`.spec.mon.zones`

Description

Zones are specified when we want to provide zonal awareness to mons

Type

`array`

`.spec.mon.zones[]`

Description

MonZoneSpec represents the specification of a zone in a Ceph Cluster

Type

`object`

Property	Type	Description
<code>arbiter</code>	<code>boolean</code>	Arbiter determines if the zone contains the arbiter used for stretch cluster mode
<code>name</code>	<code>string</code>	Name is the name of the zone
<code>volumeClaimTemplate</code>	<code>object</code>	VolumeClaimTemplate is the PVC template

`.spec.mon.zones[].volumeClaimTemplate`

Description

VolumeClaimTemplate is the PVC template

Type

object

Property	Type	Description
metadata	ObjectMeta	Standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata
spec	object	spec defines the desired characteristics of a volume requested by a pod author. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims

.spec.mon.zones[].volumeClaimTemplate.spec

Description

spec defines the desired characteristics of a volume requested by a pod author. More info:
<https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims>

Type

object

Property	Type	Description
accessModes	array	accessModes contains the desired access modes volume should have. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1
dataSource	object	dataSource field can be used to specify either:

Property	Type	Description
		• An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) • An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.
dataSourceRef	object	dataSourceRef specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a PersistentVolumeClaim object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will require the functionality of the dataSource field and as such both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in dataSourceRef, both fields (dataSource and dataSourceRef) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in dataSourceRef, dataSource isn't set to the same value and must be empty. There are three important differences between dataSource and dataSourceRef

Property	Type	Description
		- While dataSource only allows two specific type objects, dataSourceRef allows any non-core object as well as PersistentVolumeClaim objects. - While dataSource ignores disallowed values (dropping them), dataSourceRef preserves all values, and generates an error if a disallowed value is specified. - While dataSource only allows local objects, dataSourceRef allows objects in any namespace (Beta) Using this field requires the AnyVolumeDataSource feature gate to be enabled (Alpha) Using the namespace field of dataSourceRef requires the CrossNamespaceVolumeDataSource feature gate to be enabled.
resources	object	resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources
selector	object	selector is a label query over volumes to consider for binding.

Property	Type	Description
<code>storageClassName</code>	<code>string</code>	<code>storageClassName</code> is the name of the <code>StorageClass</code> required by the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#class-1 ↗
<code>volumeAttributesClassName</code>	<code>string</code>	<code>volumeAttributesClassName</code> may be used to set the <code>VolumeAttributesClass</code> used by this claim. If specified, the CSI driver will create or update the volume with attributes defined in the corresponding <code>VolumeAttributesClass</code>. This has a different purpose than <code>storageClassName</code>, it can be changed after the claim is created. An empty string or nil value indicates that no <code>VolumeAttributesClass</code> will be applied to the claim. If the claim enters an Infeasible error state, the field can be reset to its previous value (including nil) to cancel the modification. If the resource referred to by <code>volumeAttributesClassName</code> does not exist, this <code>PersistentVolumeClaim</code> will be set to a Pending state as reflected by the <code>modifyVolumeStatus</code> field, until such as a resource exists. More info: https://kubernetes.io/docs/concepts/storage/volume-attributes-classes/ ↗
<code>volumeMode</code>	<code>string</code>	<code>volumeMode</code> defines what type of volume is required by the claim. Value of <code>Filesystem</code> is implied when not included in claim spec.
<code>volumeName</code>	<code>string</code>	<code>volumeName</code> is the binding reference to the <code>PersistentVolume</code> backing this claim.

`.spec.mon.zones[].volumeClaimTemplate.spec.accessModes`

Description

accessModes contains the desired access modes the volume should have. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1>

Type

array

`.spec.mon.zones[].volumeClaimTemplate.spec.accessModes[]`

Type

string

`.spec.mon.zones[].volumeClaimTemplate.spec.dataSource`

Description

dataSource field can be used to specify either: * An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) * An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.

Type

object

Required

kind

name

Property	Type	Description
<code>apiGroup</code>	<code>string</code>	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
<code>kind</code>	<code>string</code>	Kind is the type of resource being referenced
<code>name</code>	<code>string</code>	Name is the name of resource being referenced

`.spec.mon.zones[].volumeClaimTemplate.spec.dataSourceRef`

Description

`dataSourceRef` specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a `PersistentVolumeClaim` object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will replace the functionality of the `dataSource` field and as such if both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in `dataSourceRef`, both fields (`dataSource` and `dataSourceRef`) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in `dataSourceRef`, `dataSource` isn't set to the same value and must be empty. There are three important differences between `dataSource` and `dataSourceRef`:

- * While `dataSource` only allows two specific types of objects, `dataSourceRef` allows any non-core object, as well as `PersistentVolumeClaim` objects.
- * While `dataSource` ignores disallowed values (dropping them), `dataSourceRef` preserves all values, and generates an error if a disallowed value is specified.
- * While `dataSource` only allows local objects, `dataSourceRef` allows objects in any namespaces.

(Beta) Using this field requires the `AnyVolumeDataSource` feature gate to be enabled.
 (Alpha) Using the namespace field of `dataSourceRef` requires the `CrossNamespaceVolumeDataSource` feature gate to be enabled.

Type

object

Required

kind

name

Property	Type	Description
apiGroup	string	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	string	Kind is the type of resource being referenced
name	string	Name is the name of resource being referenced
namespace	string	Namespace is the namespace of resource being referenced Note that when a namespace is specified, a gateway.networking.k8s.io/ReferenceGrant object is required in the referent namespace to allow that namespace's owner to accept the reference. See the ReferenceGrant documentation for details. (Alpha) This field requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

.spec.mon.zones[].volumeClaimTemplate.spec.resources

Description

resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher

than capacity recorded in the status field of the claim. More info:

<https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources>

Type

object

Property	Type	Description
<code>limits</code>	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
<code>requests</code>	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

`.spec.mon.zones[].volumeClaimTemplate.spec.resources.limits`

Description

Limits describes the maximum amount of compute resources allowed. More info:

<https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

`.spec.mon.zones[].volumeClaimTemplate.spec.resources.requests`

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info:

<https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

`.spec.mon.zones[].volumeClaimTemplate.spec.selector`

Description

selector is a label query over volumes to consider for binding.

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	matchExpressions is a list of label selector requirements. The requirements are ANDed.
<code>matchLabels</code>	object	matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

.spec.mon.zones[].volumeClaimTemplate.spec.selector.matchExpressions

Description

matchExpressions is a list of label selector requirements. The requirements are ANDed.

Type

array

.spec.mon.zones[].volumeClaimTemplate.spec.selector.matchExpressions[]

Description

A label selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

object

Required

key operator

Property	Type	Description
key	string	key is the label key that the selector applies to.
operator	string	operator represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists and DoesNotExist.

Property	Type	Description
<code>values</code>	<code>array</code>	values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

`.spec.mon.zones[].volumeClaimTemplate.spec.selector.matchExpressions[].values`

Description

values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

Type

`array`

`.spec.mon.zones[].volumeClaimTemplate.spec.selector.matchExpressions[].values[]`

Type

`string`

`.spec.mon.zones[].volumeClaimTemplate.spec.selector.matchLabels`

Description

matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

Type

object

.spec.monitoring

Description

Prometheus based Monitoring settings

Type

object

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled determines whether to create the prometheus rules for the ceph cluster. If true, the prometheus types must exist or the creation will fail. Default is false.
<code>exporter</code>	<code>object</code>	Ceph exporter configuration
<code>externalMgrEndpoints</code>	<code>array</code>	ExternalMgrEndpoints points to an existing Ceph prometheus exporter endpoint
<code>externalMgrPrometheusPort</code>	<code>integer</code>	ExternalMgrPrometheusPort Prometheus exporter port
<code>interval</code>	<code>string</code>	Interval determines prometheus scrape interval

Property	Type	Description
<code>metricsDisabled</code>	<code>boolean</code>	Whether to disable the metrics reported by Ceph. If false, the prometheus mgr module and Ceph exporter are enabled. If true, the prometheus mgr module and Ceph exporter are both disabled. Default is false.
<code>port</code>	<code>integer</code>	Port is the prometheus server port

.spec.monitoring.exporter

Description

Ceph exporter configuration

Type

`object`

Property	Type	Description
<code>hostNetwork</code>	<code>boolean</code>	Whether host networking is enabled for CephExporter. If not set, the network settings from CephCluster.spec.networking will be applied.
<code>perfCountersPrioLimit</code>	<code>integer</code>	Only performance counters greater than or equal to this option are fetched

Property	Type	Description
statsPeriodSeconds	integer	Time to wait before sending requests again to exporter server (seconds)

.spec.monitoring.externalMgrEndpoints

Description

ExternalMgrEndpoints points to an existing Ceph prometheus exporter endpoint

Type

array

.spec.monitoring.externalMgrEndpoints[]

Description

EndpointAddress is a tuple that describes single IP address. Deprecated: This API is deprecated in v1.33+.

Type

object

Required

ip

Property	Type	Description
hostname	string	The Hostname of this endpoint
ip	string	The IP of this endpoint. May not be loopback (127.0.0.0/8 or ::1), link-local (169.254.0.0/16 or fe80::/10), or link-local multicast (224.0.0.0/24 or ff02::/16).

Property	Type	Description
<code>nodeName</code>	<code>string</code>	Optional: Node hosting this endpoint. This can be used to determine endpoints local to a node.
<code>targetRef</code>	<code>object</code>	Reference to object providing the endpoint.

`.spec.monitoring.externalMgrEndpoints[].targetRef`

Description

Reference to object providing the endpoint.

Type

`object`

Property	Type	Description
<code>apiVersion</code>	<code>string</code>	API version of the referent.
<code>fieldPath</code>	<code>string</code>	If referring to a piece of an object instead of an entire object, this string should contain a valid JSON/Go field access statement, such as <code>desiredState.manifest.containers[2]</code>. For example, if the object reference is to a container within a pod, this would take on a value like: <code>"spec.containers{name}"</code> (where "name" refers to the name of the container that triggered the event) or if no container name is specified <code>"spec.containers[2]"</code> (container with index 2 in this pod). This syntax is chosen only to have some well-defined way of referencing a part of an object.

Property	Type	Description
<code>kind</code>	<code>string</code>	Kind of the referent. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
<code>name</code>	<code>string</code>	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names ↗
<code>namespace</code>	<code>string</code>	Namespace of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/namespaces/ ↗
<code>resourceVersion</code>	<code>string</code>	Specific resourceVersion to which this reference is made, if any. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#concurrency-control-and-consistency ↗
<code>uid</code>	<code>string</code>	UID of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#uids ↗

.spec.network

Description

Network related configuration

Type

object

Property	Type	Description
addressRanges	object	AddressRanges specify a list of CIDRs that Rook will ap to Ceph's 'public_network' and/or 'cluster_network' configurations. This config section may be used for the "host" or "multus" network providers.
connections	object	Settings for network connections such as compression a encryption across the wire.
dualStack	boolean	DualStack determines whether Ceph daemons should lis on both IPv4 and IPv6
hostNetwork	boolean	HostNetwork to enable host network. If host networking enabled or disabled on a running cluster, then the opera will automatically fail over all the mons to apply the new network settings.
ipFamily	string	IPFamily is the single stack IPv6 or IPv4 protocol
multiClusterService	object	Enable multiClusterService to export the Services betwe peer clusters
provider	string	Provider is what provides network connectivity to the clu e.g. "host" or "multus". If the Provider is updated from be

Property	Type	Description
		empty to "host" on a running cluster, then the operator will automatically fail over all the mons to apply the "host" network settings.
<code>selectors</code>	<code>object</code>	Selectors define NetworkAttachmentDefinitions to be used for Ceph public and/or cluster networks when the "multus" network provider is used. This config section is not used with other network providers. Valid keys are "public" and "cluster". Refer to Ceph networking documentation for more: https://docs.ceph.com/en/latest/rados/configuration/network-config-ref/ Refer to Multus network annotation documentation for how to select values: https://github.com/k8snetworkplumbingwg/multus-cni/blob/master/docs/how-to-use.md#run-pod-with-network-annotation Rook will make a best-effort attempt to automatically detect CIDR address ranges for given network attachment definitions. Rook's methods are robust but may be imprecise for sufficiently complicated networks. Rook's auto-detection process obtains a new IP address lease for each CephCluster reconcile. If Rook fails to detect, incorrectly detects, only partially detects, or if underlying networks do not support reusing old IP addresses, it is best to use the 'addressRanges' config section to specify CIDR ranges for the Ceph cluster. As a contrived example, one can use a theoretical Kubernetes-wide network for Ceph client traffic and a theoretical Rook-only network for Ceph replication traffic

Property	Type	Description
		shown: selectors: public: "default/cluster-fast-net" cluster "rook-ceph/ceph-backend-net"

.spec.network.addressRanges

Description

AddressRanges specify a list of CIDRs that Rook will apply to Ceph's 'public_network' and/or 'cluster_network' configurations. This config section may be used for the "host" or "multus" network providers.

Type

object

Property	Type	Description
cluster	array	Cluster defines a list of CIDRs to use for Ceph cluster network communication.
public	array	Public defines a list of CIDRs to use for Ceph public network communication.

.spec.network.addressRanges.cluster

Description

Cluster defines a list of CIDRs to use for Ceph cluster network communication.

Type

array

.spec.network.addressRanges.cluster[]

Description

An IPv4 or IPv6 network CIDR. This naive kubebuilder regex provides immediate feedback for some typos and for a common problem case where the range spec is forgotten (e.g., /24). Rook does in-depth validation in code.

Type

string

.spec.network.addressRanges.public

Description

Public defines a list of CIDRs to use for Ceph public network communication.

Type

array

.spec.network.addressRanges.public[]

Description

An IPv4 or IPv6 network CIDR. This naive kubebuilder regex provides immediate feedback for some typos and for a common problem case where the range spec is forgotten (e.g., /24). Rook does in-depth validation in code.

Type

string

.spec.network.connections

Description

Settings for network connections such as compression and encryption across the wire.

Type

object

Property	Type	Description
<code>compression</code>	<code>object</code>	Compression settings for the network connections.
<code>encryption</code>	<code>object</code>	Encryption settings for the network connections.
<code>requireMsgr2</code>	<code>boolean</code>	Whether to require msgr2 (port 3300) even if compression or encryption are not enabled. If true, the msgr1 port (6789) will be disabled. Requires a kernel that supports msgr2 (kernel 5.11 or CentOS 8.4 or newer).

`.spec.network.connections.compression`

Description

Compression settings for the network connections.

Type

`object`

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Whether to compress the data in transit across the wire. The default is not set.

`.spec.network.connections.encryption`

Description

Encryption settings for the network connections.

Type

`object`

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Whether to encrypt the data in transit across the wire to prevent eavesdropping the data on the network. The default is not set. Even if encryption is not enabled, clients still establish a strong initial authentication for the connection and data integrity is still validated with a crc check. When encryption is enabled, all communication between clients and Ceph daemons, or between Ceph daemons will be encrypted.

.spec.network.multiClusterService

Description

Enable multiClusterService to export the Services between peer clusters

Type

`object`

Property	Type	Description
<code>clusterID</code>	<code>string</code>	ClusterID uniquely identifies a cluster. It is used as a prefix to nslookup exported services. For example: ...svc.clusterset.local
<code>enabled</code>	<code>boolean</code>	Enable multiClusterService to export the mon and OSD services to peer cluster. Ensure that peer clusters are connected using an MCS API compatible application, like Globalnet Submariner.

.spec.network.selectors

Description

Selectors define NetworkAttachmentDefinitions to be used for Ceph public and/or cluster networks when the "multus" network provider is used. This config section is not used for other network providers. Valid keys are "public" and "cluster". Refer to Ceph networking documentation for more: <https://docs.ceph.com/en/latest/rados/configuration/network-config-ref/> Refer to Multus network annotation documentation for help selecting values: <https://github.com/k8snetworkplumbingwg/multus-cni/blob/master/docs/how-to-use.md#run-pod-with-network-annotation> Rook will make a best-effort attempt to automatically detect CIDR address ranges for given network attachment definitions. Rook's methods are robust but may be imprecise for sufficiently complicated networks. Rook's auto-detection process obtains a new IP address lease for each CephCluster reconcile. If Rook fails to detect, incorrectly detects, only partially detects, or if underlying networks do not support reusing old IP addresses, it is best to use the 'addressRanges' config section to specify CIDR ranges for the Ceph cluster. As a contrived example, one can use a theoretical Kubernetes-wide network for Ceph client traffic and a theoretical Rook-only network for Ceph replication traffic as shown: selectors: public: "default/cluster-fast-net" cluster: "rook-ceph/ceph-backend-net"

Type

object

.spec.placement

Type

object

.spec.priorityClassNames

Description

PriorityClassNames sets priority classes on components

Type

object

.spec.resources

Description

Resources set resource requests and limits

Type

object

.spec.security

Description

Security represents security settings

Type

object

Property	Type	Description
<code>cephx</code>	object	CephX configures CephX key settings. More: https://docs.ceph.com/en/latest/dev/cephx/ ↗
<code>keyRotation</code>	object	KeyRotation defines options for rotation of OSD disk encryption keys.
<code>kms</code>	object	KeyManagementService is the main Key Management option

.spec.security.cephx

Description

CephX configures CephX key settings. More: <https://docs.ceph.com/en/latest/dev/cephx/>

Type

object

Property	Type	Description
csi	object	CSI configures CephX key rotation settings for the Ceph-CSI daemons in the current Kubernetes cluster. CSI key rotation can affect existing PV connections, so take care when exercising this option.
daemon	object	Daemon configures CephX key settings for local Ceph daemons managed by Rook and part of the Ceph cluster. Daemon CephX keys can be rotated without affecting client connections.
rbdMirrorPeer	object	RBDMirrorPeer configures CephX key settings of the <code>rbd-mirror-peer</code> user that is used for creating bootstrap peer token used connect peer clusters. Rotating the <code>rbd-mirror-peer</code> user key will update the mirror peer token. Rotation will affect any existing peers connected to this cluster, so take care when exercising this option.

.spec.security.cephx.csi

Description

CSI configures CephX key rotation settings for the Ceph-CSI daemons in the current Kubernetes cluster. CSI key rotation can affect existing PV connections, so take care when exercising this option.

Type

object

Property	Type	Description
<code>keepPriorKeyCountMax</code>	<code>integer</code>	KeepPriorKeyCountMax tells Rook how many prior keys to keep active. Generally, this would be set to 1 to allow for a migration period for applications. If desired, set this to 0 to delete prior keys after migration. This config only applies to prior keys that already exist. If PriorKeyCount is set to 2 while only a single key currently exists, only a single prior key will be kept, and the reported status will only indicate the actual number of prior keys, not necessarily a reflection of PriorKeyCount config here.
<code>keyGeneration</code>	<code>integer</code>	KeyGeneration specifies the desired CephX key generation. This is used when KeyRotationPolicy is KeyGeneration and ignored for other policies. If this is set to greater than the current key generation, relevant keys will be rotated, and the generation value will be updated to this new value (generation values are not necessarily incremental, though that is the intended use case). If this is set to less than or equal to the current key generation, keys are not rotated.
<code>keyRotationPolicy</code>	<code>string</code>	KeyRotationPolicy controls if and when CephX keys are rotated after initial creation. One of Disabled, or KeyGeneration. Default Disabled.

`.spec.security.cephx.daemon`

Description

Daemon configures CephX key settings for local Ceph daemons managed by Rook and part of the Ceph cluster. Daemon CephX keys can be rotated without affecting client connections.

Type

object

Property	Type	Description
keyGeneration	integer	KeyGeneration specifies the desired CephX key generation. This is used when KeyRotationPolicy is KeyGeneration and ignored for other policies. If this is set to greater than the current key generation, relevant keys will be rotated, and the generation value will be updated to this new value (generation values are not necessarily incremental, though that is the intended use case). If this is set to less than or equal to the current key generation, keys are not rotated.
keyRotationPolicy	string	KeyRotationPolicy controls if and when CephX keys are rotated after initial creation. One of Disabled, or KeyGeneration. Default Disabled.

.spec.security.cephx.rbdMirrorPeer

Description

RBDMirrorPeer configures CephX key settings of the `rbd-mirror-peer` user that is used for creating bootstrap peer token used connect peer clusters. Rotating the `rbd-mirror-peer` user key will update the mirror peer token. Rotation will affect any existing peers connected to this cluster, so take care when exercising this option.

Type

object

Property	Type	Description
keyGeneration	integer	KeyGeneration specifies the desired CephX key generation. This is used when KeyRotationPolicy is KeyGeneration and ignored for other policies. If this is set to greater than the current key generation, relevant keys will be rotated, and the generation value will be updated to this new value (generation values are not necessarily incremental, though that is the intended use case). If this is set to less than or equal to the current key generation, keys are not rotated.
keyRotationPolicy	string	KeyRotationPolicy controls if and when CephX keys are rotated after initial creation. One of Disabled, or KeyGeneration. Default Disabled.

.spec.security.keyRotation

Description

KeyRotation defines options for rotation of OSD disk encryption keys.

Type

object

Property	Type	Description
enabled	boolean	Enabled represents whether the key rotation is enabled.
schedule	string	Schedule represents the cron schedule for key rotation.

.spec.security.kms

Description

KeyManagementService is the main Key Management option

Type

object

Property	Type	Description
<code>connectionDetails</code>	object	ConnectionDetails contains the KMS connection details (address, port etc)
<code>tokenSecretName</code>	string	TokenSecretName is the kubernetes secret containing the KMS token

.spec.security.kms.connectionDetails

Description

ConnectionDetails contains the KMS connection details (address, port etc)

Type

object

.spec.storage

Description

A spec for available storage in the cluster and how it should be used

Type

object

Property	Type	Description
<code>allowDeviceClassUpdate</code>	<code>boolean</code>	Whether to allow updating the device class after the OSD is initially provisioned
<code>allowOsdCrushWeightUpdate</code>	<code>boolean</code>	Whether Rook will resize the OSD CRUSH weight when the OSD PVC size is increased. This allows cluster data to be rebalanced to make most effective use of new OSD space. The default is false since data rebalancing can cause temporary cluster slowdown.
<code>backfillFullRatio</code>	<code>number</code>	BackfillFullRatio is the ratio at which the cluster is too full for backfill. Backfill will be disabled if above this threshold. Default is 0.90.
<code>config</code>	<code>object</code>	
<code>deviceFilter</code>	<code>string</code>	A regular expression to allow more fine-grained selection of devices on nodes across the cluster
<code>devicePathFilter</code>	<code>string</code>	A regular expression to allow more fine-grained selection of devices with path names

Property	Type	Description
<code>devices</code>	<code>array</code>	List of devices to use as storage devices
<code>flappingRestartIntervalHours</code>	<code>integer</code>	FlappingRestartIntervalHours defines the time for which the OSD pods, that failed with zero exit code, will sleep before restarting. This is needed for OSD flapping where OSD daemons are marked down more than 5 times in 600 seconds by Ceph. Preventing the OSD pods to restart immediately in such scenarios will prevent Rook from marking OSD as <code>up</code> and thus peering of the PGs mapped to the OSD. User needs to manually restart the OSD pod if they manage to fix the underlying OSD flapping issue before the restart interval. The sleep will be disabled if this interval is set to 0.
<code>fullRatio</code>	<code>number</code>	FullRatio is the ratio at which the cluster is considered full and ceph will stop accepting writes. Default is 0.95.
<code>migration</code>	<code>object</code>	Migration handles the OSD migration
<code>nearFullRatio</code>	<code>number</code>	NearFullRatio is the ratio at which the cluster is considered nearly full and will

Property	Type	Description
		raise a ceph health warning. Default is 0.85.
<code>nodes</code>	<code>array</code>	
<code>onlyApplyOSDPlacement</code>	<code>boolean</code>	
<code>osdMaxUpdatesInParallel</code>	<code>integer</code>	The maximum number of OSDs to update in parallel.
<code>scheduleAlways</code>	<code>boolean</code>	Whether to always schedule OSDs on a node even if the node is not currently scheduleable or ready
<code>storageClassDeviceSets</code>	<code>array</code>	
<code>store</code>	<code>object</code>	OSDStore is the backend storage type used for creating the OSDs
<code>useAllDevices</code>	<code>boolean</code>	Whether to consume all the storage devices found on a machine
<code>useAllNodes</code>	<code>boolean</code>	
<code>volumeClaimTemplates</code>	<code>array</code>	PersistentVolumeClaims to use as storage

.spec.storage.config

Type

object

.spec.storage.devices

Description

List of devices to use as storage devices

Type

array

.spec.storage.devices[]

Description

Device represents a disk to use in the cluster

Type

object

Property	Type	Description
config	object	
fullpath	string	
name	string	

.spec.storage.devices[].config

Type

object

.spec.storage.migration

Description

Migration handles the OSD migration

Type

object

Property	Type	Description
confirmation	string	A user confirmation to migrate the OSDs. It destroys each OSD one at a time, cleans up the backing disk and prepares OSD with same ID on that disk

.spec.storage.nodes

Type

array

.spec.storage.nodes[]

Description

Node is a storage nodes

Type

object

Property	Type	Description
config	object	
deviceFilter	string	A regular expression to allow more fine-grained selection of devices on nodes across the cluster

Property	Type	Description
<code>devicePathFilter</code>	<code>string</code>	A regular expression to allow more fine-grained selection of devices with path names
<code>devices</code>	<code>array</code>	List of devices to use as storage devices
<code>name</code>	<code>string</code>	
<code>resources</code>	<code>object</code>	ResourceRequirements describes the compute resource requirements.
<code>useAllDevices</code>	<code>boolean</code>	Whether to consume all the storage devices found on a machine
<code>volumeClaimTemplates</code>	<code>array</code>	PersistentVolumeClaims to use as storage

`.spec.storage.nodes[].config`

Type

`object`

`.spec.storage.nodes[].devices`

Description

List of devices to use as storage devices

Type

`array`

`.spec.storage.nodes[].devices[]`

Description

Device represents a disk to use in the cluster

Type

object

Property	Type	Description
<code>config</code>	object	
<code>fullpath</code>	string	
<code>name</code>	string	

`.spec.storage.nodes[].devices[].config`

Type

object

`.spec.storage.nodes[].resources`

Description

ResourceRequirements describes the compute resource requirements.

Type

object

Property	Type	Description
<code>claims</code>	array	Claims lists the names of resources, defined in <code>spec.resourceClaims</code>, that are used by this container. This field depends on the <code>DynamicResourceAllocation</code> feature gate.

Property	Type	Description
		This field is immutable. It can only be set for containers.
limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

.spec.storage.nodes[].resources.claims

Description

Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This field depends on the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.

Type

array

.spec.storage.nodes[].resources.claims[]

Description

ResourceClaim references one entry in PodSpec.ResourceClaims.

Type

`object`

Required

`name`

Property	Type	Description
<code>name</code>	<code>string</code>	Name must match the name of one entry in pod.spec.resourceClaims of the Pod where this field is used. It makes that resource available inside a container.
<code>request</code>	<code>string</code>	Request is the name chosen for a request in the referenced claim. If empty, everything from the claim is made available, otherwise only the result of this request.

`.spec.storage.nodes[].resources.limits`

Description

Limits describes the maximum amount of compute resources allowed. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

`.spec.storage.nodes[].resources.requests`

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

`.spec.storage.nodes[].volumeClaimTemplates`

Description

PersistentVolumeClaims to use as storage

Type

array

`.spec.storage.nodes[].volumeClaimTemplates[]`

Description

VolumeClaimTemplate is a simplified version of K8s corev1's PVC. It has no type meta or status.

Type

object

Property	Type	Description
<code>metadata</code>	<code>ObjectMeta</code>	Standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata ↗
<code>spec</code>	<code>object</code>	spec defines the desired characteristics of a volume requested by a pod author. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims ↗

`.spec.storage.nodes[].volumeClaimTemplates[].spec`

Description

spec defines the desired characteristics of a volume requested by a pod author. More info:
<https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims>

Type

object

Property	Type	Description
accessModes	array	accessModes contains the desired access modes volume should have. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1
dataSource	object	dataSource field can be used to specify either: • An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) • An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.
dataSourceRef	object	dataSourceRef specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a PersistentVolumeClaim object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume

Property	Type	Description
		populator or dynamic provisioner. This field will replace the functionality of the dataSource field and as such both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in dataSourceRef, both fields (dataSource and dataSourceRef) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in dataSourceRef, dataSource isn't set to the same value and must be empty. There are three important differences between dataSource and dataSourceRef • While dataSource only allows two specific types of objects, dataSourceRef allows any non-core object as well as PersistentVolumeClaim objects. • While dataSource ignores disallowed values (dropping them), dataSourceRef preserves all values, and generates an error if a disallowed value is specified. • While dataSource only allows local objects, dataSourceRef allows objects in any namespace (Beta) Using this field requires the AnyVolumeDataSource feature gate to be enabled (Alpha) Using the namespace field of dataSourceRef requires the CrossNamespaceVolumeDataSource feature gate to be enabled.
resources	object	resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info:

Property	Type	Description
		https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources ↗
selector	object	selector is a label query over volumes to consider binding.
storageClassName	string	storageClassName is the name of the StorageClass required by the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#class-1 ↗
volumeAttributesClassName	string	volumeAttributesClassName may be used to set the VolumeAttributesClass used by this claim. If specified, the CSI driver will create or update the volume with attributes defined in the corresponding VolumeAttributesClass. This has a different purpose than storageClassName, it can be changed after the claim is created. An empty string or nil value indicates that no VolumeAttributesClass will be applied to the claim. If the claim enters an Infeasible error state, the field can be reset to its previous value (including nil) to cancel the modification. If the resource referred to by volumeAttributesClass does not exist, this PersistentVolumeClaim will be set to a Pending state as reflected by the modifyVolumeStatus field, until such a resource exists. More info: https://kubernetes.io/docs/concepts/storage/volume-attributes-classes/ ↗

Property	Type	Description
<code>volumeMode</code>	<code>string</code>	volumeMode defines what type of volume is required by the claim. Value of Filesystem is implied when not included in claim spec.
<code>volumeName</code>	<code>string</code>	volumeName is the binding reference to the PersistentVolume backing this claim.

`.spec.storage.nodes[].volumeClaimTemplates[].spec.accessModes`

Description

accessModes contains the desired access modes the volume should have. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1>

Type

`array`

`.spec.storage.nodes[].volumeClaimTemplates[].spec.accessModes[]`

Type

`string`

`.spec.storage.nodes[].volumeClaimTemplates[].spec.dataSource`

Description

dataSource field can be used to specify either: * An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) * An existing PVC (PersistentVolumeClaim) If

the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.

Type

object

Required

kind name

Property	Type	Description
apiGroup	string	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	string	Kind is the type of resource being referenced
name	string	Name is the name of resource being referenced

.spec.storage.nodes[].volumeClaimTemplates[].spec.dataSourceRef

Description

dataSourceRef specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a PersistentVolumeClaim object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will replace the functionality of the dataSource field and as

such if both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in dataSourceRef, both fields (dataSource and dataSourceRef) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in dataSourceRef, dataSource isn't set to the same value and must be empty. There are three important differences between dataSource and dataSourceRef:

- * While dataSource only allows two specific types of objects, dataSourceRef allows any non-core object, as well as PersistentVolumeClaim objects.
- * While dataSource ignores disallowed values (dropping them), dataSourceRef preserves all values, and generates an error if a disallowed value is specified.
- * While dataSource only allows local objects, dataSourceRef allows objects in any namespaces.

(Beta) Using this field requires the AnyVolumeDataSource feature gate to be enabled.

(Alpha) Using the namespace field of dataSourceRef requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

Type

object

Required

kind name

Property	Type	Description
apiGroup	string	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	string	Kind is the type of resource being referenced
name	string	Name is the name of resource being referenced
namespace	string	Namespace is the namespace of resource being referenced Note that when a namespace is specified, a

Property	Type	Description
		gateway.networking.k8s.io/ReferenceGrant object is required in the referent namespace to allow that namespace's owner to accept the reference. See the ReferenceGrant documentation for details. (Alpha) This field requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

`.spec.storage.nodes[].volumeClaimTemplates[].spec.resources`

Description

resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info:

<https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources>

Type

object

Property	Type	Description
<code>limits</code>	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
<code>requests</code>	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info:

Property	Type	Description
		https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

`.spec.storage.nodes[].volumeClaimTemplates[].spec.resources.limits`

Description

Limits describes the maximum amount of compute resources allowed. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

`.spec.storage.nodes[].volumeClaimTemplates[].spec.resources.requests`

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

`.spec.storage.nodes[].volumeClaimTemplates[].spec.selector`

Description

selector is a label query over volumes to consider for binding.

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	<code>matchExpressions</code> is a list of label selector requirements. The requirements are ANDed.
<code>matchLabels</code>	object	<code>matchLabels</code> is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

`.spec.storage.nodes[].volumeClaimTemplates[].spec.selector.matchExpressions`

Description

`matchExpressions` is a list of label selector requirements. The requirements are ANDed.

Type

array

`.spec.storage.nodes[].volumeClaimTemplates[].spec.selector.matchExpressions[]`

Description

A label selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

object

Required

key

operator

Property	Type	Description
key	string	key is the label key that the selector applies to.
operator	string	operator represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists and DoesNotExist.
values	array	values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

.spec.storage.nodes[].volumeClaimTemplates[].spec.selector.matchExpressions[].values

Description

values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

Type

array

.spec.storage.nodes[].volumeClaimTemplates[].spec.selector.matchExpressions[].values[]

Type

string

.spec.storage.nodes[].volumeClaimTemplates[].spec.selector.matchLabels

Description

matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

Type

object

.spec.storage.storageClassDeviceSets

Type

array

.spec.storage.storageClassDeviceSets[]

Description

StorageClassDeviceSet is a storage class device set

Type

object

Required

count name volumeClaimTemplates

Property	Type	Description
config	object	Provider-specific device configuration
count	integer	Count is the number of devices in this set

Property	Type	Description
encrypted	boolean	Whether to encrypt the deviceSet
name	string	Name is a unique identifier for the set
placement	object	
portable	boolean	Portable represents OSD portability across the hosts
preparePlacement	object	
resources	object	ResourceRequirements describes the compute resource requirements.
schedulerName	string	Scheduler name for OSD pod placement
tuneDeviceClass	boolean	TuneSlowDeviceClass Tune the OSD when running on a slow Device Class
tuneFastDeviceClass	boolean	TuneFastDeviceClass Tune the OSD when running on a fast Device Class
volumeClaimTemplates	array	VolumeClaimTemplates is a list of PVC templates for the underlying storage devices

.spec.storage.storageClassDeviceSets[].config

Description

Provider-specific device configuration

Type

object

.spec.storage.storageClassDeviceSets[].placement

Type

object

Property	Type	Description
nodeAffinity	object	
podAffinity	object	
podAntiAffinity	object	
tolerations	array	
topologySpreadConstraints	array	

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	object	

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

preference weight

Property	Type	Description
preference	object	
weight	integer	

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference

Type

object

Property	Type	Description
matchExpressions	array	

Property	Type	Description
<code>matchFields</code>	<code>array</code>	

`.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions`

Type

`array`

`.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[]`

Type

`object`

Required

`key` `operator`

Property	Type	Description
<code>key</code>	<code>string</code>	
<code>operator</code>	<code>string</code>	
<code>values</code>	<code>array</code>	

`.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[].values`

Type

array

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields

Type

array

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	

Property	Type	Description
values	array	

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[].values

Type

array

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution

Type

object

Required

nodeSelectorTerms

Property	Type	Description
nodeSelectorTerms	array	

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms**

Type

array

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms[]**

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchFields</code>	array	

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms[].matchExpressions**

Type

array

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms[].matchExpressions[]**

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms[].matchExpressions[].values**

Type

array

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms[].matchExpressions[].values[]**

Type

string

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms[].matchFields**

Type

array

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms[].matchFields[]**

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms[].matchFields[].values**

Type

array

**.spec.storage.storageClassDeviceSets[].placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.
nodeSelectorTerms[].matchFields[].values[]**

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	array	

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

podAffinityTerm weight

Property	Type	Description
podAffinityTerm	object	
weight	integer	

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector

Type

object

Property	Type	Description
matchExpressions	array	

Property	Type	Description
<code>matchLabels</code>	<code>object</code>	

`.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions`

Type

`array`

`.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[]`

Type

`object`

Required

`key` `operator`

Property	Type	Description
<code>key</code>	<code>string</code>	
<code>operator</code>	<code>string</code>	
<code>values</code>	<code>array</code>	

`.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values`

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[]

.podAffinityTerm.namespaceSelector.matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

`.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions`

Type

array

`.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[]`

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[]`

namespaceSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	array	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

podAffinityTerm

weight

Property	Type	Description
podAffinityTerm	object	
weight	integer	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

`.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions`

Type

array

`.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[]`

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces

Type

array

`.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces[]`

Type

string

`.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution`

Type

array

`.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[]`

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	

Property	Type	Description
<code>namespaces</code>	<code>array</code>	
<code>topologyKey</code>	<code>string</code>	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector

Type

`object`

Property	Type	Description
<code>matchExpressions</code>	<code>array</code>	
<code>matchLabels</code>	<code>object</code>	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions

Type

`array`

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[]

Type

`object`

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys[]

Type

string

`.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector`

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

`.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions`

Type

array

`.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[]`

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution

on[].namespaces

Type

array

.spec.storage.storageClassDeviceSets[].placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.tolerations

Type

array

.spec.storage.storageClassDeviceSets[].placement.tolerations[]

Type

object

Property	Type	Description
effect	string	
key	string	
operator	string	
tolerationSeconds	integer	

Property	Type	Description
<code>value</code>	<code>string</code>	

.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints

Type

`array`

.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints[]

Type

`object`

Required

`maxSkew` `topologyKey` `whenUnsatisfiable`

Property	Type	Description
<code>labelSelector</code>	<code>object</code>	
<code>matchLabelKeys</code>	<code>array</code>	
<code>maxSkew</code>	<code>integer</code>	
<code>minDomains</code>	<code>integer</code>	
<code>nodeAffinityPolicy</code>	<code>string</code>	
<code>nodeTaintsPolicy</code>	<code>string</code>	
<code>topologyKey</code>	<code>string</code>	
<code>whenUnsatisfiable</code>	<code>string</code>	

`.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints[].labelSelector`

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

`.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints[].labelSelector.matchExpressions`

Type

array

`.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints[].labelSelector.matchExpressions[]`

Type

object

Required

key

operator

Property	Type	Description
<code>key</code>	string	
<code>operator</code>	string	
<code>values</code>	array	

.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints[].labelSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints[].labelSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints[].labelSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints[].matchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].placement.topologySpreadConstraints[].matchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement

Type

object

Property	Type	Description
nodeAffinity	object	
podAffinity	object	
podAntiAffinity	object	
tolerations	array	
topologySpreadConstraints	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	object	

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.preferredDuringSchedulingIgnoredDuringEx

ecution

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.nodeAffinity.preferredDuringSchedulingIgnoredDuringEx
ecution[]**

Type

object

Required

preference weight

Property	Type	Description
preference	object	
weight	integer	

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.nodeAffinity.preferredDuringSchedulingIgnoredDuringEx
ecution[].preference**

Type

object

Property	Type	Description
matchExpressions	array	
matchFields	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[].values

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[].values[]`

Type

string

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[]`

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[].values`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[].values[]`

Type

string

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution`

Type

object

Required

nodeSelectorTerms

Property	Type	Description
nodeSelectorTerms	array	

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms`

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[]

Type

object

Property	Type	Description
matchExpressions	array	
matchFields	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions[].values`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions[].values[]`

Type

string

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringEx`

ecution.nodeSelectorTerms[].matchFields[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields[].values

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	array	

**.spec.storage.storageClassDeviceSets[].preparePlacement.
t.podAffinity.preferredDuringSchedulingIgnoredDuringExecution**

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacement.
t.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[]**

Type

object

Required

podAffinityTerm weight

Property	Type	Description
podAffinityTerm	object	
weight	integer	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector

Type

object

Property	Type	Description
matchExpressions	array	

Property	Type	Description
matchLabels	object	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution

**cution[].podAffinityTerm.labelSelector.matchExpressions[
].values**

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.podAffinity.preferredDuringSchedulingIgnoredDuringExe
cution[].podAffinityTerm.labelSelector.matchExpressions[
].values[]**

Type

string

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.podAffinity.preferredDuringSchedulingIgnoredDuringExe
cution[].podAffinityTerm.labelSelector.matchLabels**

Type

object

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.podAffinity.preferredDuringSchedulingIgnoredDuringExe
cution[].podAffinityTerm.matchLabelKeys**

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.podAffinity.preferredDuringSchedulingIgnoredDuringExe**

cution[].podAffinityTerm.matchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacement.
podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values[]**

Type

string

**.spec.storage.storageClassDeviceSets[].preparePlacement.
podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchLabels**

Type

object

**.spec.storage.storageClassDeviceSets[].preparePlacement.
podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces**

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacement.
podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces[]**

Type

string

**.spec.storage.storageClassDeviceSets[].preparePlacement.
podAffinity.requiredDuringSchedulingIgnoredDuringExecution**

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacement.
podAffinity.requiredDuringSchedulingIgnoredDuringExecution[]**

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector`

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[]`

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.storage.storageClassDeviceSets[].preparePlacement.t.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.t.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values[]`

Type

string

`.spec.storage.storageClassDeviceSets[].preparePlacement.t.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchLabels`

Type

object

`.spec.storage.storageClassDeviceSets[].preparePlacement.t.podAffinity.requiredDuringSchedulingIgnoredDuringExe`

`cution[].matchLabelKeys`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys[]`

Type

string

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys[]`

Type

string

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector`

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

**.spec.storage.storageClassDeviceSets[].preparePlacement.
podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions**

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacement.
podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[]**

Type

object

Required

key

operator

Property	Type	Description
<code>key</code>	string	
<code>operator</code>	string	
<code>values</code>	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

podAffinityTerm

weight

Property	Type	Description
podAffinityTerm	object	
weight	integer	

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.podAntiAffinity.preferredDuringSchedulingIgnoredDurin
gExecution[].podAffinityTerm**

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector`

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[]`

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values[]`

Type

string

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchLabels`

Type

object

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

.spec.storage.storageClassDeviceSets[].preparePlacement.t.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.storage.storageClassDeviceSets[].preparePlacement.t.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values[]`

Type

string

`.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuring`

Execution[].namespaceSelector.matchLabels

Type

object

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces[]

Type

string

.spec.storage.storageClassDeviceSets[].preparePlacement.tolerations

Type

array

.spec.storage.storageClassDeviceSets[].preparePlacement.tolerations[]

Type

object

Property	Type	Description
effect	string	
key	string	
operator	string	
tolerationSeconds	integer	
value	string	

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.topologySpreadConstraints**

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.topologySpreadConstraints[]**

Type

object

Required

maxSkew topologyKey whenUnsatisfiable

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
maxSkew	integer	
minDomains	integer	

Property	Type	Description
nodeAffinityPolicy	string	
nodeTaintsPolicy	string	
topologyKey	string	
whenUnsatisfiable	string	

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.topologySpreadConstraints[].labelSelector**

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.topologySpreadConstraints[].labelSelector.matchExpres
sions**

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.topologySpreadConstraints[].labelSelector.matchExpres
sions[]**

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.topologySpreadConstraints[].labelSelector.matchExpres
sions[].values**

Type

array

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.topologySpreadConstraints[].labelSelector.matchExpres
sions[].values[]**

Type

string

**.spec.storage.storageClassDeviceSets[].preparePlacemen
t.topologySpreadConstraints[].labelSelector.matchLabels**

Type

object

`.spec.storage.storageClassDeviceSets[].preparePlacement.topologySpreadConstraints[].matchLabelKeys`

Type

array

`.spec.storage.storageClassDeviceSets[].preparePlacement.topologySpreadConstraints[].matchLabelKeys[]`

Type

string

`.spec.storage.storageClassDeviceSets[].resources`

Description

ResourceRequirements describes the compute resource requirements.

Type

object

Property	Type	Description
		Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container.
<code>claims</code>	<code>array</code>	This field depends on the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
<code>limits</code>	<code>object</code>	Limits describes the maximum amount of compute resources allowed. More info:

Property	Type	Description
		https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
	requests object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

`.spec.storage.storageClassDeviceSets[].resources.claims`

Description

Claims lists the names of resources, defined in `spec.resourceClaims`, that are used by this container. This field depends on the `DynamicResourceAllocation` feature gate. This field is immutable. It can only be set for containers.

Type

array

`.spec.storage.storageClassDeviceSets[].resources.claims[]`

Description

ResourceClaim references one entry in `PodSpec.ResourceClaims`.

Type

object

Required

name

Property	Type	Description
<code>name</code>	<code>string</code>	Name must match the name of one entry in <code>pod.spec.resourceClaims</code> of the Pod where this field is used. It makes that resource available inside a container.
<code>request</code>	<code>string</code>	Request is the name chosen for a request in the referenced claim. If empty, everything from the claim is made available, otherwise only the result of this request.

`.spec.storage.storageClassDeviceSets[].resources.limits`

Description

Limits describes the maximum amount of compute resources allowed. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

`.spec.storage.storageClassDeviceSets[].resources.requests`

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

.spec.storage.storageClassDeviceSets[].volumeClaimTemplates

Description

VolumeClaimTemplates is a list of PVC templates for the underlying storage devices

Type

array

.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[]

Description

VolumeClaimTemplate is a simplified version of K8s corev1's PVC. It has no type meta or status.

Type

object

Property	Type	Description
metadata	ObjectMeta	Standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata ↗
spec	object	spec defines the desired characteristics of a volume requested by a pod author. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims ↗

.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec

Description

spec defines the desired characteristics of a volume requested by a pod author. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims>

Type

object

Property	Type	Description
accessModes	array	accessModes contains the desired access modes volume should have. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1
dataSource	object	dataSource field can be used to specify either: • An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) • An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.

Property	Type	Description
<code>dataSourceRef</code>	<code>object</code>	<code>dataSourceRef</code> specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a <code>PersistentVolumeClaim</code> object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will require the functionality of the <code>dataSource</code> field and as such both fields are non-empty, they must have the same value. For backwards compatibility, when <code>namespace</code> isn't specified in <code>dataSourceRef</code>, both fields (<code>dataSource</code> and <code>dataSourceRef</code>) will be set to the same value automatically if one of them is empty and the other is non-empty. When <code>namespace</code> is specified in <code>dataSourceRef</code>, <code>dataSource</code> isn't set to the same value and must be empty. There are three important differences between <code>dataSource</code> and <code>dataSourceRef</code>: • While <code>dataSource</code> only allows two specific type objects, <code>dataSourceRef</code> allows any non-core object as well as <code>PersistentVolumeClaim</code> objects. • While <code>dataSource</code> ignores disallowed values (dropping them), <code>dataSourceRef</code> preserves all values, and generates an error if a disallowed value is specified. • While <code>dataSource</code> only allows local objects, <code>dataSourceRef</code> allows objects in any namespace (Beta) Using this field requires the <code>AnyVolumeDataSource</code> feature gate to be enabled (Alpha) Using the <code>namespace</code> field of <code>dataSourceRef</code> requires the <code>CrossNamespaceVolumeDataSource</code> feature gate to be enabled.

Property	Type	Description
<code>resources</code>	<code>object</code>	resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info: https://kubernetes.io/docs/concepts/storage/persistentvolumes#resources
<code>selector</code>	<code>object</code>	selector is a label query over volumes to consider for binding.
<code>storageClassName</code>	<code>string</code>	storageClassName is the name of the StorageClass required by the claim. More info: https://kubernetes.io/docs/concepts/storage/persistentvolumes#class-1
<code>volumeAttributesClassName</code>	<code>string</code>	volumeAttributesClassName may be used to set the VolumeAttributesClass used by this claim. If specified, the CSI driver will create or update the volume with the attributes defined in the corresponding VolumeAttributesClass. This has a different purpose than storageClassName, it can be changed after the claim is created. An empty string or nil value indicates that no VolumeAttributesClass will be applied to the claim. If the claim enters an Infeasible error state, the field can be reset to its previous value (including nil) to cancel the modification. If the resource referred to by volumeAttributesClassName does not exist, this PersistentVolumeClaim will be set to a Pending state

Property	Type	Description
		as reflected by the modifyVolumeStatus field, until such as a resource exists. More info: https://kubernetes.io/docs/concepts/storage/volume-attributes-classes/ ↗
<code>volumeMode</code>	<code>string</code>	volumeMode defines what type of volume is required by the claim. Value of Filesystem is implied when not included in claim spec.
<code>volumeName</code>	<code>string</code>	volumeName is the binding reference to the PersistentVolume backing this claim.

`.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.accessModes`

Description

accessModes contains the desired access modes the volume should have. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1>

Type

`array`

`.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.accessModes[]`

Type

`string`

`.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.dataSource`

Description

dataSource field can be used to specify either: * An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) * An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If the namespace is specified, then dataSourceRef will not be copied to dataSource.

Type

`object`

Required

`kind``name`

Property	Type	Description
<code>apiGroup</code>	<code>string</code>	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
<code>kind</code>	<code>string</code>	Kind is the type of resource being referenced
<code>name</code>	<code>string</code>	Name is the name of resource being referenced

`.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.dataSourceRef`

Description

dataSourceRef specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a PersistentVolumeClaim object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will replace the functionality of the dataSource field and as such if both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in dataSourceRef, both fields (dataSource and dataSourceRef) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in dataSourceRef, dataSource isn't set to the same value and must be empty. There are three important differences between dataSource and dataSourceRef: * While dataSource only allows two specific types of objects, dataSourceRef allows any non-core object, as well as PersistentVolumeClaim objects. * While dataSource ignores disallowed values (dropping them), dataSourceRef preserves all values, and generates an error if a disallowed value is specified. * While dataSource only allows local objects, dataSourceRef allows objects in any namespaces. (Beta) Using this field requires the AnyVolumeDataSource feature gate to be enabled. (Alpha) Using the namespace field of dataSourceRef requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

Type

object

Required

kind name

Property	Type	Description
apiGroup	string	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	string	Kind is the type of resource being referenced

Property	Type	Description
<code>name</code>	<code>string</code>	Name is the name of resource being referenced
<code>namespace</code>	<code>string</code>	Namespace is the namespace of resource being referenced Note that when a namespace is specified, a gateway.networking.k8s.io/ReferenceGrant object is required in the referent namespace to allow that namespace's owner to accept the reference. See the ReferenceGrant documentation for details. (Alpha) This field requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

`.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.resources`

Description

resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info:

<https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources>

Type

`object`

Property	Type	Description
<code>limits</code>	<code>object</code>	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/

Property	Type	Description
<code>requests</code>	<code>object</code>	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

`.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.resources.limits`

Description

Limits describes the maximum amount of compute resources allowed. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

`.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.resources.requests`

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.selector

Description

selector is a label query over volumes to consider for binding.

Type

object

Property	Type	Description
matchExpressions	array	matchExpressions is a list of label selector requirements. The requirements are ANDed.
matchLabels	object	matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.selector.matchExpressions

Description

matchExpressions is a list of label selector requirements. The requirements are ANDed.

Type

array

.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.selector.matchExpressions[]

Description

A label selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

object

Required

key operator

Property	Type	Description
key	string	key is the label key that the selector applies to.
operator	string	operator represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists and DoesNotExist.
values	array	values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.selector.matchExpressions[].values

Description

values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

Type

array

`.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.selector.matchExpressions[].values[]`

Type

`string`

`.spec.storage.storageClassDeviceSets[].volumeClaimTemplates[].spec.selector.matchLabels`

Description

matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

Type

`object`

`.spec.storage.store`

Description

OSDStore is the backend storage type used for creating the OSDs

Type

`object`

Property	Type	Description
<code>type</code>	<code>string</code>	Type of backend storage to be used while creating OSDs. If empty, then bluestore will be used

Property	Type	Description
<code>updateStore</code>	<code>string</code>	UpdateStore updates the backend store for existing OSDs. It destroys each OSD one at a time, cleans up the backing disk and prepares same OSD on that disk

`.spec.storage.volumeClaimTemplates`

Description

PersistentVolumeClaims to use as storage

Type

`array`

`.spec.storage.volumeClaimTemplates[]`

Description

VolumeClaimTemplate is a simplified version of K8s corev1's PVC. It has no type meta or status.

Type

`object`

Property	Type	Description
<code>metadata</code>	<code>ObjectMeta</code>	Standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata ↗
<code>spec</code>	<code>object</code>	spec defines the desired characteristics of a volume requested by a pod author. More info:

Property	Type	Description
		https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims ↗

.spec.storage.volumeClaimTemplates[].spec

Description

spec defines the desired characteristics of a volume requested by a pod author. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#persistentvolumeclaims>

Type

object

Property	Type	Description
<code>accessModes</code>	array	accessModes contains the desired access modes volume should have. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1 ↗
<code>dataSource</code>	object	dataSource field can be used to specify either: • An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) • An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the AnyVolumeDataSource feature gate is enabled, dataSource contents will be copied to dataSourceRef, and dataSourceRef contents will be copied to dataSource when dataSourceRef.namespace is not specified. If t

Property	Type	Description
		namespace is specified, then dataSourceRef will not be copied to dataSource.
dataSourceRef	object	dataSourceRef specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a PersistentVolumeClaim object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will replace the functionality of the dataSource field and as such both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in dataSourceRef, both fields (dataSource and dataSourceRef) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in dataSourceRef, dataSource isn't set to the same value and must be empty. There are three important differences between dataSource and dataSourceRef: • While dataSource only allows two specific types of objects, dataSourceRef allows any non-core object as well as PersistentVolumeClaim objects. • While dataSource ignores disallowed values (dropping them), dataSourceRef preserves all values, and generates an error if a disallowed value is specified. • While dataSource only allows local objects, dataSourceRef allows objects in any namespace (Beta) Using this field requires the AnyVolumeDataSource feature gate to be enabled (Alpha) Using the namespace field of

Property	Type	Description
		dataSourceRef requires the CrossNamespaceVolumeDataSource feature gate to be enabled.
resources	object	resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources
selector	object	selector is a label query over volumes to consider for binding.
storageClassName	string	storageClassName is the name of the StorageClass required by the claim. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#class-1
volumeAttributesClassName	string	volumeAttributesClassName may be used to set the VolumeAttributesClass used by this claim. If specified, the CSI driver will create or update the volume with attributes defined in the corresponding VolumeAttributesClass. This has a different purpose than storageClassName, it can be changed after the claim is created. An empty string or nil value indicates that no VolumeAttributesClass will be applied to the claim. If the claim enters an Infeasible error state, the field can be reset to its previous value (including nil).

Property	Type	Description
		cancel the modification. If the resource referred to volumeAttributesClass does not exist, this PersistentVolumeClaim will be set to a Pending state as reflected by the modifyVolumeStatus field, until such as a resource exists. More info: https://kubernetes.io/docs/concepts/storage/volume-attributes-classes/ ↗
volumeMode	string	volumeMode defines what type of volume is required by the claim. Value of Filesystem is implied when not included in claim spec.
volumeName	string	volumeName is the binding reference to the PersistentVolume backing this claim.

`.spec.storage.volumeClaimTemplates[].spec.accessModes`

Description

accessModes contains the desired access modes the volume should have. More info: <https://kubernetes.io/docs/concepts/storage/persistent-volumes#access-modes-1>

Type

array

`.spec.storage.volumeClaimTemplates[].spec.accessModes[]`

Type

`string`

`.spec.storage.volumeClaimTemplates[].spec.dataSource`

Description

`dataSource` field can be used to specify either: * An existing VolumeSnapshot object (snapshot.storage.k8s.io/VolumeSnapshot) * An existing PVC (PersistentVolumeClaim) If the provisioner or an external controller can support the specified data source, it will create a new volume based on the contents of the specified data source. When the `AnyVolumeDataSource` feature gate is enabled, `dataSource` contents will be copied to `dataSourceRef`, and `dataSourceRef` contents will be copied to `dataSource` when `dataSourceRef.namespace` is not specified. If the namespace is specified, then `dataSourceRef` will not be copied to `dataSource`.

Type

`object`

Required

`kind``name`

Property	Type	Description
<code>apiGroup</code>	<code>string</code>	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
<code>kind</code>	<code>string</code>	Kind is the type of resource being referenced
<code>name</code>	<code>string</code>	Name is the name of resource being referenced

`.spec.storage.volumeClaimTemplates[].spec.dataSourceRef`

Description

dataSourceRef specifies the object from which to populate the volume with data, if a non-empty volume is desired. This may be any object from a non-empty API group (non core object) or a PersistentVolumeClaim object. When this field is specified, volume binding will only succeed if the type of the specified object matches some installed volume populator or dynamic provisioner. This field will replace the functionality of the dataSource field and as such if both fields are non-empty, they must have the same value. For backwards compatibility, when namespace isn't specified in dataSourceRef, both fields (dataSource and dataSourceRef) will be set to the same value automatically if one of them is empty and the other is non-empty. When namespace is specified in dataSourceRef, dataSource isn't set to the same value and must be empty. There are three important differences between dataSource and dataSourceRef: * While dataSource only allows two specific types of objects, dataSourceRef allows any non-core object, as well as PersistentVolumeClaim objects. * While dataSource ignores disallowed values (dropping them), dataSourceRef preserves all values, and generates an error if a disallowed value is specified. * While dataSource only allows local objects, dataSourceRef allows objects in any namespaces. (Beta) Using this field requires the AnyVolumeDataSource feature gate to be enabled. (Alpha) Using the namespace field of dataSourceRef requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

Type

object

Required

kind name

Property	Type	Description
apiGroup	string	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	string	Kind is the type of resource being referenced

Property	Type	Description
<code>name</code>	<code>string</code>	Name is the name of resource being referenced
<code>namespace</code>	<code>string</code>	Namespace is the namespace of resource being referenced Note that when a namespace is specified, a gateway.networking.k8s.io/ReferenceGrant object is required in the referent namespace to allow that namespace's owner to accept the reference. See the ReferenceGrant documentation for details. (Alpha) This field requires the CrossNamespaceVolumeDataSource feature gate to be enabled.

`.spec.storage.volumeClaimTemplates[].spec.resources`

Description

resources represents the minimum resources the volume should have. Users are allowed to specify resource requirements that are lower than previous value but must still be higher than capacity recorded in the status field of the claim. More info:

<https://kubernetes.io/docs/concepts/storage/persistent-volumes#resources>

Type

`object`

Property	Type	Description
<code>limits</code>	<code>object</code>	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

Property	Type	Description
<code>requests</code>	<code>object</code>	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

`.spec.storage.volumeClaimTemplates[].spec.resources.limits`

Description

Limits describes the maximum amount of compute resources allowed. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

`.spec.storage.volumeClaimTemplates[].spec.resources.requests`

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

`.spec.storage.volumeClaimTemplates[].spec.selector`

Description

selector is a label query over volumes to consider for binding.

Type

object

Property	Type	Description
matchExpressions	array	matchExpressions is a list of label selector requirements. The requirements are ANDed.
matchLabels	object	matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

.spec.storage.volumeClaimTemplates[].spec.selector.matchExpressions

Description

matchExpressions is a list of label selector requirements. The requirements are ANDed.

Type

array

.spec.storage.volumeClaimTemplates[].spec.selector.matchExpressions[]

Description

A label selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

object

Required

key

operator

Property	Type	Description
key	string	key is the label key that the selector applies to.
operator	string	operator represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists and DoesNotExist.
values	array	values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

.spec.storage.volumeClaimTemplates[].spec.selector.matchExpressions[].values

Description

values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

Type

array

.spec.storage.volumeClaimTemplates[].spec.selector.matchExpressions[].values[]

Type

string

.spec.storage.volumeClaimTemplates[].spec.selector.matchLabels

Description

matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

Type

object

.status

Description

ClusterStatus represents the status of a Ceph cluster

Type

object

Property	Type	Description
ceph	object	CephStatus is the details health of a Ceph Cluster
cephx	object	ClusterCephxStatus defines the cephx key rotation status of various daemons on the cephCluster resource
conditions	array	
message	string	

Property	Type	Description
<code>observedGeneration</code>	<code>integer</code>	ObservedGeneration is the latest generation observed by the controller.
<code>phase</code>	<code>string</code>	ConditionType represent a resource's status
<code>state</code>	<code>string</code>	ClusterState represents the state of a Ceph Cluster
<code>storage</code>	<code>object</code>	CephStorage represents flavors of Ceph Cluster Storage
<code>version</code>	<code>object</code>	ClusterVersion represents the version of a Ceph Cluster

.status.ceph

Description

CephStatus is the details health of a Ceph Cluster

Type

`object`

Property	Type	Description
<code>capacity</code>	<code>object</code>	Capacity is the capacity information of a Ceph Cluster
<code>details</code>	<code>object</code>	

Property	Type	Description
fsid	string	
health	string	
lastChanged	string	
lastChecked	string	
previousHealth	string	
versions	object	CephDaemonsVersions show the current ceph version for different ceph daemons

.status.ceph.capacity

Description

Capacity is the capacity information of a Ceph Cluster

Type

object

Property	Type	Description
bytesAvailable	integer	
bytesTotal	integer	
bytesUsed	integer	
lastUpdated	string	

.status.ceph.details

Type

object

.status.ceph.versions

Description

CephDaemonsVersions show the current ceph version for different ceph daemons

Type

object

Property	Type	Description
cephfs-mirror	object	CephFSMirror shows CephFSMirror Ceph version
mds	object	Mds shows Mds Ceph version
mgr	object	Mgr shows Mgr Ceph version
mon	object	Mon shows Mon Ceph version
osd	object	Osd shows Osd Ceph version
overall	object	Overall shows overall Ceph version
rbd-mirror	object	RbdMirror shows RbdMirror Ceph version

Property	Type	Description
<code>rgw</code>	<code>object</code>	Rgw shows Rgw Ceph version

`.status.ceph.versions.cephfs-mirror`

Description

CephFSMirror shows CephFSMirror Ceph version

Type

`object`

`.status.ceph.versions.mds`

Description

Mds shows Mds Ceph version

Type

`object`

`.status.ceph.versions.mgr`

Description

Mgr shows Mgr Ceph version

Type

`object`

`.status.ceph.versions.mon`

Description

Mon shows Mon Ceph version

Type

object

.status.ceph.versions.osd

Description

Osd shows Osd Ceph version

Type

object

.status.ceph.versions.overall

Description

Overall shows overall Ceph version

Type

object

.status.ceph.versions.rbd-mirror

Description

RbdMirror shows RbdMirror Ceph version

Type

object

.status.ceph.versions.rgw

Description

Rgw shows Rgw Ceph version

Type

object

.status.cephx

Description

ClusterCephxStatus defines the cephx key rotation status of various daemons on the cephCluster resource

Type

object

Property	Type	Description
<code>admin</code>	object	Admin shows the CephX key status for the client.admin key
<code>cephExporter</code>	object	Ceph Exporter represents the cephx key rotation status of the ceph exporter daemon
<code>crashCollector</code>	object	Crash Collector represents the cephx key rotation status of the crash collector daemon
<code>csi</code>	object	CSI shows the CephX key status for Ceph-CSI components.
<code>mgr</code>	object	Mgr represents the cephx key rotation status of the ceph manager daemon
<code>mon</code>	object	Mon represents the CephX key status of the Monitor daemons

Property	Type	Description
<code>osd</code>	<code>object</code>	OSD shows the CephX key status of of OSDs
<code>rbdmirrorPeer</code>	<code>object</code>	RBDMirrorPeer represents the cephx key rotation status of the <code>rbd-mirror-peer</code> user

`.status.cephx.admin`

Description

Admin shows the CephX key status for the client.admin key

Type

`object`

Property	Type	Description
<code>keyCephVersion</code>	<code>string</code>	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code> . For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.
<code>keyGeneration</code>	<code>integer</code>	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code> . When keys are rotated

Property	Type	Description
		due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.

.status.cephx.cephExporter

Description

Ceph Exporter represents the cephx key rotation status of the ceph exporter daemon

Type

object

Property	Type	Description
keyCephVersion	string	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code> . For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.
keyGeneration	integer	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code> . When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation.

Property	Type	Description
		Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.

.status.cephx.crashCollector

Description

Crash Collector represents the cephx key rotation status of the crash collector daemon

Type

object

Property	Type	Description
<code>keyCephVersion</code>	<code>string</code>	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code> . For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.

Property	Type	Description
<code>keyGeneration</code>	<code>integer</code>	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code>. When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.

`.status.cephx.csi`

Description

CSI shows the CephX key status for Ceph-CSI components.

Type

`object`

Property	Type	Description
<code>keyCephVersion</code>	<code>string</code>	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code>. For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.

Property	Type	Description
<code>keyGeneration</code>	<code>integer</code>	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code> . When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.
<code>priorKeyCount</code>	<code>integer</code>	PriorKeyCount reports the number of prior-generation CephX keys that remain active for the related component

`.status.cephx.mgr`

Description

Mgr represents the cephx key rotation status of the ceph manager daemon

Type

`object`

Property	Type	Description
<code>keyCephVersion</code>	<code>string</code>	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code> . For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An

Property	Type	Description
		empty string indicates that the version is unknown, as expected in brownfield deployments.
keyGeneration	integer	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to 1. When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation 0 indicates that keys existed prior to the implementation of key tracking.

.status.cephx.mon

Description

Mon represents the CephX key status of the Monitor daemons

Type

object

Property	Type	Description
keyCephVersion	string	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code>. For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.
keyGeneration	integer	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code>. When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.

.status.cephx.osd

Description

OSD shows the CephX key status of of OSDs

Type

object

Property	Type	Description
<code>keyCephVersion</code>	<code>string</code>	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code>. For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.
<code>keyGeneration</code>	<code>integer</code>	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code>. When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.

.status.cephx.rbdMirrorPeer

Description

RBDMirrorPeer represents the cephx key rotation status of the `rbd-mirror-peer` user

Type

`object`

Property	Type	Description
<code>keyCephVersion</code>	<code>string</code>	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code>. For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.
<code>keyGeneration</code>	<code>integer</code>	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code>. When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.

`.status.conditions`

Type

`array`

`.status.conditions[]`

Description

Condition represents a status condition on any Rook-Ceph Custom Resource.

Type

object

Property	Type	Description
lastHeartbeatTime	string	
lastTransitionTime	string	
message	string	
reason	string	ConditionReason is a reason for a condition
status	string	
type	string	ConditionType represent a resource's status

.status.storage

Description

CephStorage represents flavors of Ceph Cluster Storage

Type

object

Property	Type	Description
deprecatedOSDs	object	
deviceClasses	array	
osd	object	OSDStatus represents OSD status of the ceph Cluster

.status.storage.deprecatedOSDs

Type

object

.status.storage.deviceClasses

Type

array

.status.storage.deviceClasses[]

Description

DeviceClasses represents device classes of a Ceph Cluster

Type

object

Property	Type	Description
name	string	

.status.storage.osd

Description

OSDStatus represents OSD status of the ceph Cluster

Type

object

Property	Type	Description
migrationStatus	object	MigrationStatus status represents the current status of any OSD migration.

Property	Type	Description
<code>storeType</code>	<code>object</code>	StoreType is a mapping between the OSD backend stores and number of OSDs using these stores

`.status.storage.osd.migrationStatus`

Description

MigrationStatus status represents the current status of any OSD migration.

Type

`object`

Property	Type	Description
<code>pending</code>	<code>integer</code>	

`.status.storage.osd.storeType`

Description

StoreType is a mapping between the OSD backend stores and number of OSDs using these stores

Type

`object`

`.status.version`

Description

ClusterVersion represents the version of a Ceph Cluster

Type

`object`

Property	Type	Description
image	string	
version	string	

API Endpoints

The following API endpoints are available:

- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephclusters`
 - `DELETE` : delete collection of CephCluster
 - `GET` : list objects of kind CephCluster
 - `POST` : create a new CephCluster
- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephclusters/{name}`
 - `DELETE` : delete the specified CephCluster
 - `GET` : read the specified CephCluster
 - `PATCH` : partially update the specified CephCluster
 - `PUT` : replace the specified CephCluster
- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephclusters/{name}/status`
 - `GET` : read status of the specified CephCluster
 - `PATCH` : partially update status of the specified CephCluster
 - `PUT` : replace status of the specified CephCluster

`/apis/ceph.rook.io/v1/namespaces/{namespace}/cephclusters`

HTTP method

`DELETE`

Description

delete collection of CephCluster

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

GET

Description

list objects of kind CephCluster

HTTP responses

HTTP code	Response body
200 - OK	<code>CephClusterList</code> schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a new CephCluster

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Parameter	Type	Description
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+. - Strict: This will fail the request with a <code>BadRequest</code> error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephCluster</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephCluster</code> schema
201 - Created	<code>CephCluster</code> schema
202 - Accepted	<code>CephCluster</code> schema
401 - Unauthorized	Empty

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephclusters/{name}

HTTP method

DELETE

Description

delete the specified CephCluster

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
202 - Accepted	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

GET

Description

read the specified CephCluster

HTTP responses

HTTP code	Response body
200 - OK	<code>CephCluster</code> schema

HTTP code	Response body
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update the specified CephCluster

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephCluster</code> schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified CephCluster

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephCluster</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephCluster</code> schema
201 - Created	<code>CephCluster</code> schema
401 - Unauthorized	Empty

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephclusters/{name}/status

HTTP method

`GET`

Description

read status of the specified CephCluster

HTTP responses

HTTP code	Response body
200 - OK	<code>CephCluster</code> schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update status of the specified CephCluster

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephCluster</code> schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace status of the specified CephCluster

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized <code>dryRun</code> directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a <code>BadRequest</code> error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephCluster</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephCluster</code> schema
201 - Created	<code>CephCluster</code> schema
401 - Unauthorized	Empty

CephFilesystem

[cephfilesystems.ceph.rook.io/v1]

Description

CephFilesystem represents a Ceph Filesystem

Type

object

Required

metadata

spec

Specification

Property	Type	Description
apiVersion	string	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
kind	string	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be

Property	Type	Description
		updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
metadata	ObjectMeta	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
spec	object	FilesystemSpec represents the spec of a file system
status	object	CephFilesystemStatus represents the status of a Ceph Filesystem

.spec

Description

FilesystemSpec represents the spec of a file system

Type

object

Required

dataPools

metadataPool

metadataServer

Property	Type	Description
dataPools	array	The data pool settings, with optional predefined pool name.

Property	Type	Description
<code>metadataPool</code>	<code>object</code>	The metadata pool settings
<code>metadataServer</code>	<code>object</code>	The mds pod info
<code>mirroring</code>	<code>object</code>	The mirroring settings
<code>preserveFilesystemOnDelete</code>	<code>boolean</code>	Preserve the fs in the cluster on CephFilesystem CR deletion. Setting this to true automatically implies PreservePoolsOnDelete is true.
<code>preservePoolNames</code>	<code>boolean</code>	Preserve pool names as specified
<code>preservePoolsOnDelete</code>	<code>boolean</code>	Preserve pools on filesystem deletion
<code>statusCheck</code>	<code>object</code>	The mirroring statusCheck

.spec.dataPools

Description

The data pool settings, with optional predefined pool name.

Type

`array`

.spec.dataPools[]

Description

NamedPoolSpec represents the named ceph pool spec

Type

object

Property	Type	Description
application	string	The application name to set on the pool. Only expected to be set for rgw pools.
compressionMode	string	DEPRECATED: use Parameters instead, e.g., Parameters["compression_mode"] = "force" The inline compression mode in Bluestore OSD to set to (options are: none, passive, aggressive, force) Do NOT set a default value for kubebuilder as this will override the Parameters
crushRoot	string	The root of the crush hierarchy utilized by the pool
deviceClass	string	The device class the OSD should set to for use in the pool
enableCrushUpdates	boolean	Allow rook operator to change the pool CRUSH tunables once the pool is created

Property	Type	Description
<code>enableRBDStats</code>	<code>boolean</code>	EnableRBDStats is used to enable gathering of statistics for all RBD images in the pool
<code>erasureCoded</code>	<code>object</code>	The erasure code settings
<code>failureDomain</code>	<code>string</code>	The failure domain: osd/host/(region or zone if available) - technically also any type in the crush map
<code>mirroring</code>	<code>object</code>	The mirroring settings
<code>name</code>	<code>string</code>	Name of the pool
<code>parameters</code>	<code>object</code>	Parameters is a list of properties to enable on a given pool
<code>quotas</code>	<code>object</code>	The quota settings
<code>replicated</code>	<code>object</code>	The replication settings
<code>statusCheck</code>	<code>object</code>	The mirroring statusCheck

.spec.dataPools[].erasureCoded

Description

The erasure code settings

Type

object

Required

codingChunks

dataChunks

Property	Type	Description
algorithm	string	The algorithm for erasure coding. If absent, defaults to the plugin specified in <code>osd_pool_default_erasure_code_profile</code> .
codingChunks	integer	Number of coding chunks per object in an erasure coded storage pool (required for erasure-coded pool type). This is the number of OSDs that can be lost simultaneously before data cannot be recovered.
dataChunks	integer	Number of data chunks per object in an erasure coded storage pool (required for erasure-coded pool type). The number of chunks required to recover an object when any single OSD is lost is the same as <code>dataChunks</code> so be aware that the larger the number of data chunks, the higher the cost of recovery.

.spec.dataPools[].mirroring

Description

The mirroring settings

Type

object

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled whether this pool is mirrored or not
<code>mode</code>	<code>string</code>	Mode is the mirroring mode: pool, image or init-only.
<code>peers</code>	<code>object</code>	Peers represents the peers spec
<code>snapshotSchedules</code>	<code>array</code>	SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

.spec.dataPools[].mirroring.peers

Description

Peers represents the peers spec

Type

object

Property	Type	Description
<code>secretNames</code>	<code>array</code>	SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

.spec.dataPools[].mirroring.peers.secretNames

Description

SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

Type

array

.spec.dataPools[].mirroring.peers.secretNames[]

Type

string

.spec.dataPools[].mirroring.snapshotSchedules

Description

SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

Type

array

.spec.dataPools[].mirroring.snapshotSchedules[]

Description

SnapshotScheduleSpec represents the snapshot scheduling settings of a mirrored pool

Type

object

Property	Type	Description
interval	string	Interval represent the periodicity of the snapshot.
path	string	Path is the path to snapshot, only valid for CephFS

Property	Type	Description
<code>startTime</code>	<code>string</code>	StartTime indicates when to start the snapshot

`.spec.dataPools[].parameters`

Description

Parameters is a list of properties to enable on a given pool

Type

`object`

`.spec.dataPools[].quotas`

Description

The quota settings

Type

`object`

Property	Type	Description
<code>maxBytes</code>	<code>integer</code>	MaxBytes represents the quota in bytes Deprecated in favor of MaxSize
<code>maxObjects</code>	<code>integer</code>	MaxObjects represents the quota in objects
<code>maxSize</code>	<code>string</code>	MaxSize represents the quota in bytes as a string

.spec.dataPools[].replicated

Description

The replication settings

Type

object

Required

size

Property	Type	Description
hybridStorage	object	HybridStorage represents hybrid storage tier settings
replicasPerFailureDomain	integer	ReplicasPerFailureDomain the number of replica in the specified failure domain
requireSafeReplicaSize	boolean	RequireSafeReplicaSize if false allows you to set replica 1
size	integer	Size - Number of copies per object in a replicated storage pool, including the object itself (required for replicated pool type)
subFailureDomain	string	SubFailureDomain the name of the sub-failure domain

Property	Type	Description
<code>targetSizeRatio</code>	<code>number</code>	TargetSizeRatio gives a hint (%) to Ceph in terms of expected consumption of the total cluster capacity

`.spec.dataPools[].replicated.hybridStorage`

Description

HybridStorage represents hybrid storage tier settings

Type

`object`

Required

`primaryDeviceClass`

`secondaryDeviceClass`

Property	Type	Description
<code>primaryDeviceClass</code>	<code>string</code>	PrimaryDeviceClass represents high performance tier (for example SSD or NVME) for Primary OSD
<code>secondaryDeviceClass</code>	<code>string</code>	SecondaryDeviceClass represents low performance tier (for example HDDs) for remaining OSDs

`.spec.dataPools[].statusCheck`

Description

The mirroring statusCheck

Type

`object`

Property	Type	Description
<code>mirror</code>	<code>object</code>	HealthCheckSpec represents the health check of an object store bucket

`.spec.dataPools[].statusCheck.mirror`

Description

HealthCheckSpec represents the health check of an object store bucket

Type

`object`

Property	Type	Description
<code>disabled</code>	<code>boolean</code>	
<code>interval</code>	<code>string</code>	Interval is the internal in second or minute for the health check to run like 60s for 60 seconds
<code>timeout</code>	<code>string</code>	

`.spec.metadataPool`

Description

The metadata pool settings

Type

`object`

Property	Type	Description
<code>application</code>	<code>string</code>	The application name to set on the pool. Only expected to be set for rgw pools.
<code>compressionMode</code>	<code>string</code>	DEPRECATED: use Parameters instead, e.g., <code>Parameters["compression_mode"] = "force"</code> The inline compression mode in Bluestore OSD to set to (options are: none, passive, aggressive, force) Do NOT set a default value for kubebuilder as this will override the Parameters
<code>crushRoot</code>	<code>string</code>	The root of the crush hierarchy utilized by the pool
<code>deviceClass</code>	<code>string</code>	The device class the OSD should set to for use in the pool
<code>enableCrushUpdates</code>	<code>boolean</code>	Allow rook operator to change the pool CRUSH tunables once the pool is created
<code>enableRBDStats</code>	<code>boolean</code>	EnableRBDStats is used to enable gathering of statistics for all RBD images in the pool
<code>erasureCoded</code>	<code>object</code>	The erasure code settings
<code>failureDomain</code>	<code>string</code>	The failure domain: osd/host/(region or zone if available) - technically also any type in the crush

Property	Type	Description
		map
<code>mirroring</code>	<code>object</code>	The mirroring settings
<code>name</code>	<code>string</code>	Name of the pool
<code>parameters</code>	<code>object</code>	Parameters is a list of properties to enable on a given pool
<code>quotas</code>	<code>object</code>	The quota settings
<code>replicated</code>	<code>object</code>	The replication settings
<code>statusCheck</code>	<code>object</code>	The mirroring statusCheck

.spec.metadataPool.eraserCoded

Description

The erasure code settings

Type

`object`

Required

`codingChunks` `dataChunks`

Property	Type	Description
<code>algorithm</code>	<code>string</code>	The algorithm for erasure coding. If absent, defaults to the plugin specified in <code>osd_pool_default_erasure_code_profile</code> .
<code>codingChunks</code>	<code>integer</code>	Number of coding chunks per object in an erasure coded storage pool (required for erasure-coded pool type). This is the number of OSDs that can be lost simultaneously before data cannot be recovered.
<code>dataChunks</code>	<code>integer</code>	Number of data chunks per object in an erasure coded storage pool (required for erasure-coded pool type). The number of chunks required to recover an object when any single OSD is lost is the same as <code>dataChunks</code> so be aware that the larger the number of data chunks, the higher the cost of recovery.

.spec.metadataPool.mirroring

Description

The mirroring settings

Type

`object`

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled whether this pool is mirrored or not

Property	Type	Description
<code>mode</code>	<code>string</code>	Mode is the mirroring mode: pool, image or init-only.
<code>peers</code>	<code>object</code>	Peers represents the peers spec
<code>snapshotSchedules</code>	<code>array</code>	SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

`.spec.metadataPool.mirroring.peers`

Description

Peers represents the peers spec

Type

`object`

Property	Type	Description
<code>secretNames</code>	<code>array</code>	SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

`.spec.metadataPool.mirroring.peers.secretNames`

Description

SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

Type

`array`

`.spec.metadataPool.mirroring.peers.secretNames[]`

Type

`string`

`.spec.metadataPool.mirroring.snapshotSchedules`

Description

SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

Type

`array`

`.spec.metadataPool.mirroring.snapshotSchedules[]`

Description

SnapshotScheduleSpec represents the snapshot scheduling settings of a mirrored pool

Type

`object`

Property	Type	Description
<code>interval</code>	<code>string</code>	Interval represent the periodicity of the snapshot.
<code>path</code>	<code>string</code>	Path is the path to snapshot, only valid for CephFS
<code>startTime</code>	<code>string</code>	StartTime indicates when to start the snapshot

`.spec.metadataPool.parameters`

Description

Parameters is a list of properties to enable on a given pool

Type

object

.spec.metadataPool.quotas

Description

The quota settings

Type

object

Property	Type	Description
maxBytes	integer	MaxBytes represents the quota in bytes Deprecated in favor of MaxSize
maxObjects	integer	MaxObjects represents the quota in objects
maxSize	string	MaxSize represents the quota in bytes as a string

.spec.metadataPool.replicated

Description

The replication settings

Type

object

Required

size

Property	Type	Description
hybridStorage	object	HybridStorage represents hybrid storage tier settings
replicasPerFailureDomain	integer	ReplicasPerFailureDomain the number of replica in the specified failure domain
requireSafeReplicaSize	boolean	RequireSafeReplicaSize if false allows you to set replica 1
size	integer	Size - Number of copies per object in a replicated storage pool, including the object itself (required for replicated pool type)
subFailureDomain	string	SubFailureDomain the name of the sub-failure domain
targetSizeRatio	number	TargetSizeRatio gives a hint (%) to Ceph in terms of expected consumption of the total cluster capacity

.spec.metadataPool.replicated.hybridStorage

Description

HybridStorage represents hybrid storage tier settings

Type

object

Required

primaryDeviceClass secondaryDeviceClass

Property	Type	Description
primaryDeviceClass	string	PrimaryDeviceClass represents high performance tier (for example SSD or NVME) for Primary OSD
secondaryDeviceClass	string	SecondaryDeviceClass represents low performance tier (for example HDDs) for remaining OSDs

.spec.metadataPool.statusCheck

Description

The mirroring statusCheck

Type

object

Property	Type	Description
mirror	object	HealthCheckSpec represents the health check of an object store bucket

.spec.metadataPool.statusCheck.mirror

Description

HealthCheckSpec represents the health check of an object store bucket

Type

object

Property	Type	Description
<code>disabled</code>	<code>boolean</code>	
<code>interval</code>	<code>string</code>	Interval is the internal in second or minute for the health check to run like 60s for 60 seconds
<code>timeout</code>	<code>string</code>	

.spec.metadataServer

Description

The mds pod info

Type

object

Required

`activeCount`

Property	Type	Description
<code>activeCount</code>	<code>integer</code>	The number of metadata servers that are active. The remaining servers in the cluster will be in standby mode.
<code>activeStandby</code>	<code>boolean</code>	Whether each active MDS instance will have an active standby with a warm metadata cache for faster failover. If false, standbys

Property	Type	Description
		will still be available, but will not have a warm metadata cache.
annotations	object	The annotations-related configuration to add/set on each Pod related object.
cacheMemoryLimitFactor	number	CacheMemoryLimitFactor is the factor applied to the memory limit to determine the MDS cache memory limit. MDS cache memory limit should be set to 50-60% of RAM reserved for the MDS container. MDS uses approximately 125% of the value of mds_cache_memory_limit in RAM. This factor is applied when resources.limits.memory is set.
cacheMemoryRequestFactor	number	CacheMemoryRequestFactor is the factor applied to the memory request to determine the MDS cache memory limit. This factor is applied when resources.requests.memory is set and resources.limits.memory is not set.
labels	object	The labels-related configuration to add/set on each Pod related object.

Property	Type	Description
<code>livenessProbe</code>	<code>object</code>	ProbeSpec is a wrapper around Probe so it can be enabled or disabled for a Ceph daemon
<code>placement</code>	<code>object</code>	
<code>priorityClassName</code>	<code>string</code>	PriorityClassName sets priority classes on components
<code>resources</code>	<code>object</code>	The resource requirements for the mds pods
<code>startupProbe</code>	<code>object</code>	ProbeSpec is a wrapper around Probe so it can be enabled or disabled for a Ceph daemon

`.spec.metadataServer.annotations`

Description

The annotations-related configuration to add/set on each Pod related object.

Type

`object`

`.spec.metadataServer.labels`

Description

The labels-related configuration to add/set on each Pod related object.

Type

`object`

`.spec.metadataServer.livenessProbe`

Description

ProbeSpec is a wrapper around Probe so it can be enabled or disabled for a Ceph daemon

Type

`object`

Property	Type	Description
<code>disabled</code>	<code>boolean</code>	Disabled determines whether probe is disable or not
<code>probe</code>	<code>object</code>	Probe describes a health check to be performed against a container to determine whether it is alive or ready to receive traffic.

`.spec.metadataServer.livenessProbe.probe`

Description

Probe describes a health check to be performed against a container to determine whether it is alive or ready to receive traffic.

Type

`object`

Property	Type	Description
<code>exec</code>	<code>object</code>	Exec specifies a command to execute in the

Property	Type	Description
<code>failureThreshold</code>	<code>integer</code>	Minimum consecutive failures for the probe to be considered failed after having succeeded. Default value is 3. Minimum value is 1.
<code>grpc</code>	<code>object</code>	GRPC specifies a GRPC HealthCheckRequest
<code>httpGet</code>	<code>object</code>	HTTPGet specifies an HTTP GET request to
<code>initialDelaySeconds</code>	<code>integer</code>	Number of seconds after the container has started before liveness probes are initiated. More info: https://kubernetes.io/docs/concepts/workloads/controllers/lifecycle#container-probes
<code>periodSeconds</code>	<code>integer</code>	How often (in seconds) to perform the probe. Default is 10 seconds. Minimum value is 1.
<code>successThreshold</code>	<code>integer</code>	Minimum consecutive successes for the probe to be considered successful after having failed. Default is 1. Must be 1 for liveness and startup. Minimum value is 1.
<code>tcpSocket</code>	<code>object</code>	TCPSocket specifies a connection to a TCP
<code>terminationGracePeriodSeconds</code>	<code>integer</code>	

Property	Type	Description
<code>timeoutSeconds</code>	<code>integer</code>	Number of seconds after which the probe times out. Defaults to 1 second. Minimum value is 1. More info: https://kubernetes.io/docs/concepts/workloads/controllers/liveness-probes/

`.spec.metadataServer.livenessProbe.probe.exec`

Description

Exec specifies a command to execute in the container.

Type

`object`

Property	Type	Description
<code>command</code>	<code>array</code>	Command is the command line to execute inside the container, the working directory for the command is root ('/') in the container's filesystem. The command is simply exec'd, it is not run inside a shell, so traditional shell instructions (' ', etc) won't work. To use a shell, you need to explicitly call out to that shell. Exit status of 0 is treated as live/healthy and non-zero is unhealthy.

`.spec.metadataServer.livenessProbe.probe.exec.command`

Description

Command is the command line to execute inside the container, the working directory for the command is root ('/') in the container's filesystem. The command is simply exec'd, it is not run inside a shell, so traditional shell instructions ('|', etc) won't work. To use a shell, you

need to explicitly call out to that shell. Exit status of 0 is treated as live/healthy and non-zero is unhealthy.

Type

array

.spec.metadataServer.livenessProbe.probe.exec.command[]

Type

string

.spec.metadataServer.livenessProbe.probe.grpc

Description

GRPC specifies a GRPC HealthCheckRequest.

Type

object

Required

port

Property	Type	Description
port	integer	Port number of the gRPC service. Number must be in the range 1 to 65535.

Property	Type	Description
<code>service</code>	<code>string</code>	Service is the name of the service to place in the gRPC HealthCheckRequest (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md ↗). If this is not specified, the default behavior is defined by gRPC.

`.spec.metadataServer.livenessProbe.probe.httpGet`

Description

HTTPGet specifies an HTTP GET request to perform.

Type

`object`

Required

`port`

Property	Type	Description
<code>host</code>	<code>string</code>	Host name to connect to, defaults to the pod IP. You probably want to set "Host" in httpHeaders instead.
<code>httpHeaders</code>	<code>array</code>	Custom headers to set in the request. HTTP allows repeated headers.
<code>path</code>	<code>string</code>	Path to access on the HTTP server.

Property	Type	Description
<code>port</code>	<code>integer</code>	Name or number of the port to access on the container. Number must be in the range 1 to 65535. Name must be an IANA_SVC_NAME.
<code>scheme</code>	<code>string</code>	Scheme to use for connecting to the host. Defaults to HTTP.

`.spec.metadataServer.livenessProbe.probe.httpGet.httpHeaders`

Description

Custom headers to set in the request. HTTP allows repeated headers.

Type

`array`

`.spec.metadataServer.livenessProbe.probe.httpGet.httpHeaders[]`

Description

HTTPHeader describes a custom header to be used in HTTP probes

Type

`object`

Required

`name` `value`

Property	Type	Description
<code>name</code>	<code>string</code>	The header field name. This will be canonicalized upon output, so case-variant names will be understood as the same header.
<code>value</code>	<code>string</code>	The header field value

`.spec.metadataServer.livenessProbe.probe.tcpSocket`

Description

TCPSocket specifies a connection to a TCP port.

Type

`object`

Required

`port`

Property	Type	Description
<code>host</code>	<code>string</code>	Optional: Host name to connect to, defaults to the pod IP.
<code>port</code>		Number or name of the port to access on the container. Number must be in the range 1 to 65535. Name must be an IANA_SVC_NAME.

`.spec.metadataServer.placement`

Type

`object`

Property	Type	Description
nodeAffinity	object	
podAffinity	object	
podAntiAffinity	object	
tolerations	array	
topologySpreadConstraints	array	

.spec.metadataServer.placement.nodeAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	object	

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

- preference
- weight

Property	Type	Description
preference	object	
weight	integer	

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference

Type

- object

Property	Type	Description
matchExpressions	array	
matchFields	array	

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions

Type

- array

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[].values

Type

array

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[].values[]

Type

string

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields

Type

array

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[].values

Type

array

.spec.metadataServer.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[].values[]

Type

string

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution

Type

object

Required

nodeSelectorTerms

Property	Type	Description
nodeSelectorTerms	array	

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms

Type

array

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[]

Type

object

Property	Type	Description
matchExpressions	array	
matchFields	array	

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions

Type

array

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions[].values

Type

array

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions[].values[]

Type

string

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields

Type

array

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields[].values

Type

array

.spec.metadataServer.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields[].values[]

Type

string

.spec.metadataServer.placement.podAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	array	

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution

Type

array

`.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[]`

Type

`object`

Required

`podAffinityTerm``weight`

Property	Type	Description
<code>podAffinityTerm</code>	<code>object</code>	
<code>weight</code>	<code>integer</code>	

`.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm`

Type

`object`

Required

`topologyKey`

Property	Type	Description
<code>labelSelector</code>	<code>object</code>	
<code>matchLabelKeys</code>	<code>array</code>	
<code>mismatchLabelKeys</code>	<code>array</code>	
<code>namespaceSelector</code>	<code>object</code>	
<code>namespaces</code>	<code>array</code>	
<code>topologyKey</code>	<code>string</code>	

`.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector`

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

`.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions`

Type

array

`.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[]`

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values

Type

array

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values[]

Type

string

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchLabels

Type

object

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.

matchLabelKeys

Type

array

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys[]

Type

string

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys

Type

array

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys[]

Type

string

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchExpressions

Type

array

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchExpressions[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchExpressions[].values

Type

array

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchExpressions[].values[]

Type

string

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchLabels

Type

object

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces

Type

array

.spec.metadataServer.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces[]

Type

string

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	

Property	Type	Description
topologyKey	string	

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions

Type

array

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values

Type

array

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values[]

Type

string

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchLabels

Type

object

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys

Type

array

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys[]

Type

string

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys

Type

array

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys[]

Type

string

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions

Type

array

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[]

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector

or.matchExpressions[].values

Type

array

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values[]

Type

string

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchLabels

Type

object

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces

Type

array

.spec.metadataServer.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces[]

Type

string

.spec.metadataServer.placement.podAntiAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	array	

.spec.metadataServer.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.metadataServer.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

podAffinityTerm

weight

Property	Type	Description
podAffinityTerm	object	
weight	integer	

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm**

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.labelSelector**

Type

object

Property	Type	Description
matchExpressions	array	

Property	Type	Description
<code>matchLabels</code>	<code>object</code>	

`.spec.metadataServer.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions`

Type

`array`

`.spec.metadataServer.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[]`

Type

`object`

Required

`key` `operator`

Property	Type	Description
<code>key</code>	<code>string</code>	
<code>operator</code>	<code>string</code>	
<code>values</code>	<code>array</code>	

`.spec.metadataServer.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values`

Type

array

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.labelSelector.matchExpressions[].values[]**

Type

string

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.labelSelector.matchLabels**

Type

object

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.matchLabelKeys**

Type

array

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.matchLabelKeys[]**

Type

string

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.mismatchLabelKeys**

Type

array

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.mismatchLabelKeys[]**

Type

string

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.namespaceSelector**

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.namespaceSelector.matchExpressions**

Type

array

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.namespaceSelector.matchExpressions[]**

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.namespaceSelector.matchExpressions[].values**

Type

array

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.namespaceSelector.matchExpressions[].values[]**

Type

string

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.namespaceSelector.matchLabels**

Type

object

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.namespaces**

Type

array

**.spec.metadataServer.placement.podAntiAffinity.preferred
DuringSchedulingIgnoredDuringExecution[].podAffinityTe
rm.namespaces[]**

Type

string

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution**

Type

array

.spec.metadataServer.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

.spec.metadataServer.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].labelSelector.
matchExpressions**

Type

array

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].labelSelector.
matchExpressions[]**

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].labelSelector.
matchExpressions[].values**

Type

array

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].labelSelector.
matchExpressions[].values[]**

Type

string

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].labelSelector.
matchLabels**

Type

object

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].matchLabelK
eys**

Type

array

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].matchLabelK
eys[]**

Type

string

.spec.metadataServer.placement.podAntiAffinity.required DuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys

Type

array

.spec.metadataServer.placement.podAntiAffinity.required DuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys[]

Type

string

.spec.metadataServer.placement.podAntiAffinity.required DuringSchedulingIgnoredDuringExecution[].namespaceSelector

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

.spec.metadataServer.placement.podAntiAffinity.required DuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions

Type

array

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].namespaceS
elector.matchExpressions[]**

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].namespaceS
elector.matchExpressions[].values**

Type

array

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].namespaceS
elector.matchExpressions[].values[]**

Type

`string`

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].namespaceS
elector.matchLabels**

Type

`object`

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].namespaces**

Type

`array`

**.spec.metadataServer.placement.podAntiAffinity.required
DuringSchedulingIgnoredDuringExecution[].namespaces[
]**

Type

`string`

.spec.metadataServer.placement.tolerations

Type

`array`

.spec.metadataServer.placement.tolerations[]

Type

`object`

Property	Type	Description
effect	string	
key	string	
operator	string	
tolerationSeconds	integer	
value	string	

.spec.metadataServer.placement.topologySpreadConstraints

Type

array

.spec.metadataServer.placement.topologySpreadConstraints[]

Type

object

Required

maxSkew topologyKey whenUnsatisfiable

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
maxSkew	integer	
minDomains	integer	

Property	Type	Description
nodeAffinityPolicy	string	
nodeTaintsPolicy	string	
topologyKey	string	
whenUnsatisfiable	string	

.spec.metadataServer.placement.topologySpreadConstraints[].labelSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.metadataServer.placement.topologySpreadConstraints[].labelSelector.matchExpressions

Type

array

.spec.metadataServer.placement.topologySpreadConstraints[].labelSelector.matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.metadataServer.placement.topologySpreadConstraints[].labelSelector.matchExpressions[].values

Type

array

.spec.metadataServer.placement.topologySpreadConstraints[].labelSelector.matchExpressions[].values[]

Type

string

.spec.metadataServer.placement.topologySpreadConstraints[].labelSelector.matchLabels

Type

object

.spec.metadataServer.placement.topologySpreadConstraints[].matchLabelKeys

Type

array

.spec.metadataServer.placement.topologySpreadConstraints[].matchLabelKeys[]

Type

string

.spec.metadataServer.resources

Description

The resource requirements for the mds pods

Type

object

Property	Type	Description
		Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container.
claims	array	This field depends on the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to

Property	Type	Description
		Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

.spec.metadataServer.resources.claims

Description

Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This field depends on the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.

Type

array

.spec.metadataServer.resources.claims[]

Description

ResourceClaim references one entry in PodSpec.ResourceClaims.

Type

object

Required

name

Property	Type	Description
name	string	Name must match the name of one entry in pod.spec.resourceClaims of the Pod where this field is used. It makes that resource available inside a container.

Property	Type	Description
<code>request</code>	<code>string</code>	Request is the name chosen for a request in the referenced claim. If empty, everything from the claim is made available, otherwise only the result of this request.

`.spec.metadataServer.resources.limits`

Description

Limits describes the maximum amount of compute resources allowed. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

`.spec.metadataServer.resources.requests`

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

`object`

`.spec.metadataServer.startupProbe`

Description

ProbeSpec is a wrapper around Probe so it can be enabled or disabled for a Ceph daemon

Type

`object`

Property	Type	Description
<code>disabled</code>	<code>boolean</code>	Disabled determines whether probe is disable or not
<code>probe</code>	<code>object</code>	Probe describes a health check to be performed against a container to determine whether it is alive or ready to receive traffic.

`.spec.metadataServer.startupProbe.probe`

Description

Probe describes a health check to be performed against a container to determine whether it is alive or ready to receive traffic.

Type

`object`

Property	Type	Description
<code>exec</code>	<code>object</code>	Exec specifies a command to execute in the
<code>failureThreshold</code>	<code>integer</code>	Minimum consecutive failures for the probe to be considered failed after having succeeded. Default value is 3. Minimum value is 1.
<code>grpc</code>	<code>object</code>	GRPC specifies a GRPC HealthCheckRequest to send to the container.
<code>httpGet</code>	<code>object</code>	HTTPGet specifies an HTTP GET request to send to the container.

Property	Type	Description
<code>initialDelaySeconds</code>	<code>integer</code>	Number of seconds after the container has started before liveness probes are initiated. More info: https://kubernetes.io/docs/concepts/workloads/lifecycle#container-probes
<code>periodSeconds</code>	<code>integer</code>	How often (in seconds) to perform the probe. Default is 10 seconds. Minimum value is 1.
<code>successThreshold</code>	<code>integer</code>	Minimum consecutive successes for the probe to be considered successful after having failed. Default is 1. Must be 1 for liveness and startup. Minimum value is 1.
<code>tcpSocket</code>	<code>object</code>	TCPSocket specifies a connection to a TCP endpoint.
<code>terminationGracePeriodSeconds</code>	<code>integer</code>	Optional duration in seconds the pod may be gracefully terminated before the probe is initiated.
<code>timeoutSeconds</code>	<code>integer</code>	Number of seconds after which the probe times out. Defaults to 1 second. Minimum value is 1. More info: https://kubernetes.io/docs/concepts/workloads/lifecycle#container-probes

`.spec.metadataServer.startupProbe.probe.exec`

Description

Exec specifies a command to execute in the container.

Type

object

Property	Type	Description
command	array	Command is the command line to execute inside the container, the working directory for the command is root ('/') in the container's filesystem. The command is simply exec'd, it is not run inside a shell, so traditional shell instructions (' ', etc) won't work. To use a shell, you need to explicitly call out to that shell. Exit status of 0 is treated as live/healthy and non-zero is unhealthy.

.spec.metadataServer.startupProbe.probe.exec.command

Description

Command is the command line to execute inside the container, the working directory for the command is root ('/') in the container's filesystem. The command is simply exec'd, it is not run inside a shell, so traditional shell instructions ('|', etc) won't work. To use a shell, you need to explicitly call out to that shell. Exit status of 0 is treated as live/healthy and non-zero is unhealthy.

Type

array

.spec.metadataServer.startupProbe.probe.exec.command[]

Type

string

.spec.metadataServer.startupProbe.probe.grpc

Description

GRPC specifies a GRPC HealthCheckRequest.

Type

object

Required

port

Property	Type	Description
port	integer	Port number of the gRPC service. Number must be in the range 1 to 65535.
service	string	Service is the name of the service to place in the gRPC HealthCheckRequest (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md ↗). If this is not specified, the default behavior is defined by gRPC.

.spec.metadataServer.startupProbe.probe.httpGet

Description

HTTPGet specifies an HTTP GET request to perform.

Type

object

Required

port

Property	Type	Description
<code>host</code>	<code>string</code>	Host name to connect to, defaults to the pod IP. You probably want to set "Host" in httpHeaders instead.
<code>httpHeaders</code>	<code>array</code>	Custom headers to set in the request. HTTP allows repeated headers.
<code>path</code>	<code>string</code>	Path to access on the HTTP server.
<code>port</code>		Name or number of the port to access on the container. Number must be in the range 1 to 65535. Name must be an IANA_SVC_NAME.
<code>scheme</code>	<code>string</code>	Scheme to use for connecting to the host. Defaults to HTTP.

`.spec.metadataServer.startupProbe.probe.httpGet.httpHeaders`

Description

Custom headers to set in the request. HTTP allows repeated headers.

Type

`array`

`.spec.metadataServer.startupProbe.probe.httpGet.httpHeaders[]`

Description

HTTPHeader describes a custom header to be used in HTTP probes

Type

object

Required

namevalue

Property	Type	Description
name	string	The header field name. This will be canonicalized upon output, so case-variant names will be understood as the same header.
value	string	The header field value

.spec.metadataServer.startupProbe.probe.tcpSocket

Description

TCPSocket specifies a connection to a TCP port.

Type

object

Required

port

Property	Type	Description
host	string	Optional: Host name to connect to, defaults to the pod IP.

Property	Type	Description
<code>port</code>	<code>integer</code>	Number or name of the port to access on the container. Number must be in the range 1 to 65535. Name must be an IANA_SVC_NAME.

.spec.mirroring

Description

The mirroring settings

Type

`object`

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled whether this filesystem is mirrored or not
<code>peers</code>	<code>object</code>	Peers represents the peers spec
<code>snapshotRetention</code>	<code>array</code>	Retention is the retention policy for a snapshot schedule One path has exactly one retention policy. A policy can however contain multiple count-time period pairs in order to specify complex retention policies
<code>snapshotSchedules</code>	<code>array</code>	SnapshotSchedules is the scheduling of snapshot for mirrored filesystems

.spec.mirroring.peers

Description

Peers represents the peers spec

Type

object

Property	Type	Description
<code>secretNames</code>	array	SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

.spec.mirroring.peers.secretNames

Description

SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

Type

array

.spec.mirroring.peers.secretNames[]

Type

string

.spec.mirroring.snapshotRetention

Description

Retention is the retention policy for a snapshot schedule One path has exactly one retention policy. A policy can however contain multiple count-time period pairs in order to specify complex retention policies

Type

array

.spec.mirroring.snapshotRetention[]

Description

SnapshotScheduleRetentionSpec is a retention policy

Type

object

Property	Type	Description
duration	string	Duration represents the retention duration for a snapshot
path	string	Path is the path to snapshot

.spec.mirroring.snapshotSchedules

Description

SnapshotSchedules is the scheduling of snapshot for mirrored filesystems

Type

array

.spec.mirroring.snapshotSchedules[]

Description

SnapshotScheduleSpec represents the snapshot scheduling settings of a mirrored pool

Type

object

Property	Type	Description
<code>interval</code>	<code>string</code>	Interval represent the periodicity of the snapshot.
<code>path</code>	<code>string</code>	Path is the path to snapshot, only valid for CephFS
<code>startTime</code>	<code>string</code>	StartTime indicates when to start the snapshot

`.spec.statusCheck`

Description

The mirroring statusCheck

Type

`object`

Property	Type	Description
<code>mirror</code>	<code>object</code>	HealthCheckSpec represents the health check of an object store bucket

`.spec.statusCheck.mirror`

Description

HealthCheckSpec represents the health check of an object store bucket

Type

`object`

Property	Type	Description
<code>disabled</code>	<code>boolean</code>	
<code>interval</code>	<code>string</code>	Interval is the internal in second or minute for the health check to run like 60s for 60 seconds
<code>timeout</code>	<code>string</code>	

.status

Description

CephFilesystemStatus represents the status of a Ceph Filesystem

Type

`object`

Property	Type	Description
<code>cephx</code>	<code>object</code>	
<code>conditions</code>	<code>array</code>	
<code>info</code>	<code>object</code>	Use only info and put mirroringStatus in it?
<code>mirroringStatus</code>	<code>object</code>	MirroringStatus is the filesystem mirroring status
<code>observedGeneration</code>	<code>integer</code>	ObservedGeneration is the latest generation observed by the controller.

Property	Type	Description
<code>phase</code>	<code>string</code>	ConditionType represent a resource's status
<code>snapshotScheduleStatus</code>	<code>object</code>	FilesystemSnapshotScheduleStatusSpec is the status of the snapshot schedule

.status.cephx

Type

`object`

Property	Type	Description
<code>daemon</code>	<code>object</code>	Daemon shows the CephX key status for local Ceph daemons associated with this resources.

.status.cephx.daemon

Description

Daemon shows the CephX key status for local Ceph daemons associated with this resources.

Type

`object`

Property	Type	Description
<code>keyCephVersion</code>	<code>string</code>	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format

Property	Type	Description
		as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code> . For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.
<code>keyGeneration</code>	<code>integer</code>	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code> . When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.

`.status.conditions`

Type

`array`

`.status.conditions[]`

Description

Condition represents a status condition on any Rook-Ceph Custom Resource.

Type

`object`

Property	Type	Description
<code>lastHeartbeatTime</code>	<code>string</code>	
<code>lastTransitionTime</code>	<code>string</code>	
<code>message</code>	<code>string</code>	
<code>reason</code>	<code>string</code>	ConditionReason is a reason for a condition
<code>status</code>	<code>string</code>	
<code>type</code>	<code>string</code>	ConditionType represent a resource's status

.status.info

Description

Use only info and put mirroringStatus in it?

Type

`object`

.status.mirroringStatus

Description

MirroringStatus is the filesystem mirroring status

Type

`object`

Property	Type	Description
<code>daemonsStatus</code>	<code>array</code>	PoolMirroringStatus is the mirroring status of a filesystem
<code>details</code>	<code>string</code>	Details contains potential status errors
<code>lastChanged</code>	<code>string</code>	LastChanged is the last time time the status last changed
<code>lastChecked</code>	<code>string</code>	LastChecked is the last time time the status was checked

`.status.mirroringStatus.daemonsStatus`

Description

PoolMirroringStatus is the mirroring status of a filesystem

Type

`array`

`.status.mirroringStatus.daemonsStatus[]`

Description

FilesystemMirrorInfoSpec is the filesystem mirror status of a given filesystem

Type

`object`

Property	Type	Description
<code>daemon_id</code>	<code>integer</code>	DaemonID is the cephfs-mirror name

Property	Type	Description
<code>filesystems</code>	<code>array</code>	Filesystems is the list of filesystems managed by a given cephfs-mirror daemon

`.status.mirroringStatus.daemonsStatus[].filesystems`

Description

Filesystems is the list of filesystems managed by a given cephfs-mirror daemon

Type

`array`

`.status.mirroringStatus.daemonsStatus[].filesystems[]`

Description

FilesystemsSpec is spec for the mirrored filesystem

Type

`object`

Property	Type	Description
<code>directory_count</code>	<code>integer</code>	DirectoryCount is the number of directories in the filesystem
<code>filesystem_id</code>	<code>integer</code>	FilesystemID is the filesystem identifier
<code>name</code>	<code>string</code>	Name is name of the filesystem

Property	Type	Description
peers	array	Peers represents the mirroring peers

.status.mirroringStatus.daemonsStatus[].filesystems[].peers

Description

Peers represents the mirroring peers

Type

array

.status.mirroringStatus.daemonsStatus[].filesystems[].peers[]

Description

FilesystemMirrorInfoPeerSpec is the specification of a filesystem peer mirror

Type

object

Property	Type	Description
remote	object	Remote are the remote cluster information
stats	object	Stats are the stat a peer mirror
uuid	string	UUID is the peer unique identifier

`.status.mirroringStatus.daemonsStatus[].filesystems[].peers[].remote`

Description

Remote are the remote cluster information

Type

object

Property	Type	Description
<code>client_name</code>	<code>string</code>	ClientName is cephx name
<code>cluster_name</code>	<code>string</code>	ClusterName is the name of the cluster
<code>fs_name</code>	<code>string</code>	FsName is the filesystem name

`.status.mirroringStatus.daemonsStatus[].filesystems[].peers[].stats`

Description

Stats are the stat a peer mirror

Type

object

Property	Type	Description
<code>failure_count</code>	<code>integer</code>	FailureCount is the number of mirroring failure

Property	Type	Description
<code>recovery_count</code>	<code>integer</code>	RecoveryCount is the number of recovery attempted after failures

`.status.snapshotScheduleStatus`

Description

FilesystemSnapshotScheduleStatusSpec is the status of the snapshot schedule

Type

`object`

Property	Type	Description
<code>details</code>	<code>string</code>	Details contains potential status errors
<code>lastChanged</code>	<code>string</code>	LastChanged is the last time time the status last changed
<code>lastChecked</code>	<code>string</code>	LastChecked is the last time time the status was checked
<code>snapshotSchedules</code>	<code>array</code>	SnapshotSchedules is the list of snapshots scheduled

`.status.snapshotScheduleStatus.snapshotSchedules`

Description

SnapshotSchedules is the list of snapshots scheduled

Type

array

.status.snapshotScheduleStatus.snapshotSchedules[]

Description

FilesystemSnapshotSchedulesSpec is the list of snapshot scheduled for images in a pool

Type

object

Property	Type	Description
fs	string	Fs is the name of the Ceph Filesystem
path	string	Path is the path on the filesystem
rel_path	string	
retention	object	FilesystemSnapshotScheduleStatusRetention is the retention specification for a filesystem snapshot schedule
schedule	string	
subvol	string	Subvol is the name of the sub volume

.status.snapshotScheduleStatus.snapshotSchedules[].retention

Description

FilesystemSnapshotScheduleStatusRetention is the retention specification for a filesystem snapshot schedule

Type

object

Property	Type	Description
active	boolean	Active is whether the scheduled is active or not
created	string	Created is when the snapshot schedule was created
created_count	integer	CreatedCount is total amount of snapshots
first	string	First is when the first snapshot schedule was taken
last	string	Last is when the last snapshot schedule was taken
last_pruned	string	LastPruned is when the last snapshot schedule was pruned
pruned_count	integer	PrunedCount is total amount of pruned snapshots
start	string	Start is when the snapshot schedule starts

API Endpoints

The following API endpoints are available:

- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephfilesystems`
 - `DELETE` : delete collection of CephFilesystem
 - `GET` : list objects of kind CephFilesystem
 - `POST` : create a new CephFilesystem
- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephfilesystems/{name}`
 - `DELETE` : delete the specified CephFilesystem
 - `GET` : read the specified CephFilesystem
 - `PATCH` : partially update the specified CephFilesystem
 - `PUT` : replace the specified CephFilesystem
- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephfilesystems/{name}/status`
 - `GET` : read status of the specified CephFilesystem
 - `PATCH` : partially update status of the specified CephFilesystem
 - `PUT` : replace status of the specified CephFilesystem

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephfilesystems

HTTP method

`DELETE`

Description

delete collection of CephFilesystem

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

list objects of kind CephFilesystem

HTTP responses

HTTP code	Response body
200 - OK	<code>CephFilesystemList</code> schema
401 - Unauthorized	Empty

HTTP method

`POST`

Description

create a new CephFilesystem

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last

Parameter	Type	Description
		duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
body	CephFilesystem schema	application/json formatted

HTTP responses

HTTP code	Response body
200 - OK	CephFilesystem schema
201 - Created	CephFilesystem schema
202 - Accepted	CephFilesystem schema
401 - Unauthorized	Empty

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephfilesystems/{name}

HTTP method

DELETE

Description

delete the specified CephFilesystem

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
202 - Accepted	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

read the specified CephFilesystem

HTTP responses

HTTP code	Response body
200 - OK	<code>CephFilesystem</code> schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update the specified CephFilesystem

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephFilesystem</code> schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace the specified CephFilesystem

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephFilesystem</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephFilesystem</code> schema

HTTP code	Response body
201 - Created	<code>CephFilesystem</code> schema
401 - Unauthorized	Empty

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephfilesystems/{name}/status

HTTP method

`GET`

Description

read status of the specified CephFilesystem

HTTP responses

HTTP code	Response body
200 - OK	<code>CephFilesystem</code> schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update status of the specified CephFilesystem

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Parameter	Type	Description
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+. - Strict: This will fail the request with a <code>BadRequest</code> error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephFilesystem</code> schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace status of the specified `CephFilesystem`

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized <code>dryRun</code> directive will result in an error response and no further processing

Parameter	Type	Description
		of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephFilesystem</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephFilesystem</code> schema
201 - Created	<code>CephFilesystem</code> schema
401 - Unauthorized	Empty

CephBlockPool

[cephblockpools.ceph.rook.io/v1]

Description

CephBlockPool represents a Ceph Storage Pool

Type

object

Required

metadata

spec

Specification

Property	Type	Description
apiVersion	string	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
kind	string	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be

Property	Type	Description
		updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
metadata	ObjectMeta	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
spec	object	NamedBlockPoolSpec allows a block pool to be created with a non-default name. This is more specific than the NamedPoolSpec so we get schema validation on the allowed pool names that can be specified.
status	object	CephBlockPoolStatus represents the mirroring status of Ceph Storage Pool

.spec

Description

NamedBlockPoolSpec allows a block pool to be created with a non-default name. This is more specific than the NamedPoolSpec so we get schema validation on the allowed pool names that can be specified.

Type

object

Property	Type	Description
<code>application</code>	<code>string</code>	The application name to set on the pool. Only expected to be set for rgw pools.
<code>compressionMode</code>	<code>string</code>	DEPRECATED: use Parameters instead, e.g., <code>Parameters["compression_mode"] = "force"</code> The inline compression mode in Bluestore OSD to set to (options are: none, passive, aggressive, force) Do NOT set a default value for kubebuilder as this will override the Parameters
<code>crushRoot</code>	<code>string</code>	The root of the crush hierarchy utilized by the pool
<code>deviceClass</code>	<code>string</code>	The device class the OSD should set to for use in the pool
<code>enableCrushUpdates</code>	<code>boolean</code>	Allow rook operator to change the pool CRUSH tunables once the pool is created
<code>enableRBDStats</code>	<code>boolean</code>	EnableRBDStats is used to enable gathering of statistics for all RBD images in the pool
<code>erasureCoded</code>	<code>object</code>	The erasure code settings
<code>failureDomain</code>	<code>string</code>	The failure domain: <code>osd/host/(region or zone if available)</code> - technically also any type in the crush

Property	Type	Description
		map
<code>mirroring</code>	<code>object</code>	The mirroring settings
<code>name</code>	<code>string</code>	The desired name of the pool if different from the CephBlockPool CR name.
<code>parameters</code>	<code>object</code>	Parameters is a list of properties to enable on a given pool
<code>quotas</code>	<code>object</code>	The quota settings
<code>replicated</code>	<code>object</code>	The replication settings
<code>statusCheck</code>	<code>object</code>	The mirroring statusCheck

.spec.erasureCoded

Description

The erasure code settings

Type

`object`

Required

`codingChunks`

`dataChunks`

Property	Type	Description
<code>algorithm</code>	<code>string</code>	The algorithm for erasure coding. If absent, defaults to the plugin specified in <code>osd_pool_default_erasure_code_profile</code> .
<code>codingChunks</code>	<code>integer</code>	Number of coding chunks per object in an erasure coded storage pool (required for erasure-coded pool type). This is the number of OSDs that can be lost simultaneously before data cannot be recovered.
<code>dataChunks</code>	<code>integer</code>	Number of data chunks per object in an erasure coded storage pool (required for erasure-coded pool type). The number of chunks required to recover an object when any single OSD is lost is the same as <code>dataChunks</code> so be aware that the larger the number of data chunks, the higher the cost of recovery.

.spec.mirroring

Description

The mirroring settings

Type

`object`

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled whether this pool is mirrored or not

Property	Type	Description
<code>mode</code>	<code>string</code>	Mode is the mirroring mode: pool, image or init-only.
<code>peers</code>	<code>object</code>	Peers represents the peers spec
<code>snapshotSchedules</code>	<code>array</code>	SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

`.spec.mirroring.peers`

Description

Peers represents the peers spec

Type

`object`

Property	Type	Description
<code>secretNames</code>	<code>array</code>	SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

`.spec.mirroring.peers.secretNames`

Description

SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

Type

`array`

.spec.mirroring.peers.secretNames[]

Type

string

.spec.mirroring.snapshotSchedules

Description

SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

Type

array

.spec.mirroring.snapshotSchedules[]

Description

SnapshotScheduleSpec represents the snapshot scheduling settings of a mirrored pool

Type

object

Property	Type	Description
interval	string	Interval represent the periodicity of the snapshot.
path	string	Path is the path to snapshot, only valid for CephFS
startTime	string	StartTime indicates when to start the snapshot

.spec.parameters

Description

Parameters is a list of properties to enable on a given pool

Type

object

.spec.quotas

Description

The quota settings

Type

object

Property	Type	Description
<code>maxBytes</code>	<code>integer</code>	MaxBytes represents the quota in bytes Deprecated in favor of MaxSize
<code>maxObjects</code>	<code>integer</code>	MaxObjects represents the quota in objects
<code>maxSize</code>	<code>string</code>	MaxSize represents the quota in bytes as a string

.spec.replicated

Description

The replication settings

Type

object

Required

size

Property	Type	Description
hybridStorage	object	HybridStorage represents hybrid storage tier settings
replicasPerFailureDomain	integer	ReplicasPerFailureDomain the number of replica in the specified failure domain
requireSafeReplicaSize	boolean	RequireSafeReplicaSize if false allows you to set replica 1
size	integer	Size - Number of copies per object in a replicated storage pool, including the object itself (required for replicated pool type)
subFailureDomain	string	SubFailureDomain the name of the sub-failure domain
targetSizeRatio	number	TargetSizeRatio gives a hint (%) to Ceph in terms of expected consumption of the total cluster capacity

.spec.replicated.hybridStorage

Description

HybridStorage represents hybrid storage tier settings

Type

object

Required

primaryDeviceClass

secondaryDeviceClass

Property	Type	Description
<code>primaryDeviceClass</code>	<code>string</code>	PrimaryDeviceClass represents high performance tier (for example SSD or NVME) for Primary OSD
<code>secondaryDeviceClass</code>	<code>string</code>	SecondaryDeviceClass represents low performance tier (for example HDDs) for remaining OSDs

.spec.statusCheck

Description

The mirroring statusCheck

Type

object

Property	Type	Description
<code>mirror</code>	<code>object</code>	HealthCheckSpec represents the health check of an object store bucket

.spec.statusCheck.mirror

Description

HealthCheckSpec represents the health check of an object store bucket

Type

object

Property	Type	Description
disabled	boolean	
interval	string	Interval is the internal in second or minute for the health check to run like 60s for 60 seconds
timeout	string	

.status

Description

CephBlockPoolStatus represents the mirroring status of Ceph Storage Pool

Type

object

Property	Type	Description
cephx	object	PeerTokenCephxStatus represents the cephx key rotation status for peer tokens
conditions	array	
info	object	
mirroringInfo	object	MirroringInfoSpec is the status of the pool/radosnamespace mirroring

Property	Type	Description
<code>mirroringStatus</code>	<code>object</code>	MirroringStatusSpec is the status of the pool/radosNamespace mirroring
<code>observedGeneration</code>	<code>integer</code>	ObservedGeneration is the latest generation observed by the controller.
<code>phase</code>	<code>string</code>	ConditionType represent a resource's status
<code>poolID</code>	<code>integer</code>	optional
<code>snapshotScheduleStatus</code>	<code>object</code>	SnapshotScheduleStatusSpec is the status of the snapshot schedule

.status.cephx

Description

PeerTokenCephxStatus represents the cephx key rotation status for peer tokens

Type

`object`

Property	Type	Description
<code>peerToken</code>	<code>object</code>	PeerToken shows the rotation status of the peer token associated with the <code>rbd-mirror-peer</code> user.

.status.cephx.peerToken

Description

PeerToken shows the rotation status of the peer token associated with the `rbd-mirror-peer` user.

Type

object

Property	Type	Description
keyCephVersion	string	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code> . For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.
keyGeneration	integer	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code> . When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.

.status.conditions

Type

`array`

.status.conditions[]

Description

Condition represents a status condition on any Rook-Ceph Custom Resource.

Type

`object`

Property	Type	Description
<code>lastHeartbeatTime</code>	<code>string</code>	
<code>lastTransitionTime</code>	<code>string</code>	
<code>message</code>	<code>string</code>	
<code>reason</code>	<code>string</code>	ConditionReason is a reason for a condition
<code>status</code>	<code>string</code>	
<code>type</code>	<code>string</code>	ConditionType represent a resource's status

.status.info

Type

`object`

.status.mirroringInfo

Description

MirroringInfoSpec is the status of the pool/radosnamespace mirroring

Type

object

Property	Type	Description
details	string	
lastChanged	string	
lastChecked	string	
mode	string	Mode is the mirroring mode
peers	array	Peers are the list of peer sites connected to that cluster
site_name	string	SiteName is the current site name

.status.mirroringInfo.peers

Description

Peers are the list of peer sites connected to that cluster

Type

array

.status.mirroringInfo.peers[]

Description

PeersSpec contains peer details

Type

object

Property	Type	Description
<code>client_name</code>	<code>string</code>	ClientName is the CephX user used to connect to the peer
<code>direction</code>	<code>string</code>	Direction is the peer mirroring direction
<code>mirror_uuid</code>	<code>string</code>	MirrorUUID is the mirror UUID
<code>site_name</code>	<code>string</code>	SiteName is the current site name
<code>uuid</code>	<code>string</code>	UUID is the peer UUID

.status.mirroringStatus

Description

MirroringStatusSpec is the status of the pool/radosNamespace mirroring

Type

`object`

Property	Type	Description
<code>details</code>	<code>string</code>	Details contains potential status errors
<code>lastChanged</code>	<code>string</code>	LastChanged is the last time time the status last changed

Property	Type	Description
<code>lastChecked</code>	<code>string</code>	LastChecked is the last time time the status was checked
<code>summary</code>	<code>object</code>	Summary is the mirroring status summary

`.status.mirroringStatus.summary`

Description

Summary is the mirroring status summary

Type

`object`

Property	Type	Description
<code>daemon_health</code>	<code>string</code>	DaemonHealth is the health of the mirroring daemon
<code>group_health</code>	<code>string</code>	GroupHealth is the health of the mirrored image group
<code>group_states</code>	<code>object</code>	GroupStates is the various state for all mirrored image groups
<code>health</code>	<code>string</code>	Health is the mirroring health
<code>image_health</code>	<code>string</code>	ImageHealth is the health of the mirrored image

Property	Type	Description
<code>image_states</code>	<code>object</code>	ImageStates is the various state for all mirrored images
<code>states</code>	<code>object</code>	States is the various state for all mirrored images

`.status.mirroringStatus.summary.group_states`

Description

GroupStates is the various state for all mirrored image groups

Type

`object`

Property	Type	Description
<code>error</code>	<code>integer</code>	Error is when the mirroring state is errored
<code>replaying</code>	<code>integer</code>	Replaying is when the replay of the mirroring journal is on-going
<code>starting_replay</code>	<code>integer</code>	StartingReplay is when the replay of the mirroring journal starts
<code>stopped</code>	<code>integer</code>	Stopped is when the mirroring state is stopped

Property	Type	Description
<code>stopping_replay</code>	<code>integer</code>	StopReplaying is when the replay of the mirroring journal stops
<code>syncing</code>	<code>integer</code>	Syncing is when the image is syncing
<code>unknown</code>	<code>integer</code>	Unknown is when the mirroring state is unknown

`.status.mirroringStatus.summary.image_states`

Description

ImageStates is the various state for all mirrored images

Type

`object`

Property	Type	Description
<code>error</code>	<code>integer</code>	Error is when the mirroring state is errored
<code>replaying</code>	<code>integer</code>	Replaying is when the replay of the mirroring journal is on-going
<code>starting_replay</code>	<code>integer</code>	StartingReplay is when the replay of the mirroring journal starts

Property	Type	Description
<code>stopped</code>	<code>integer</code>	Stopped is when the mirroring state is stopped
<code>stopping_replay</code>	<code>integer</code>	StopReplaying is when the replay of the mirroring journal stops
<code>syncing</code>	<code>integer</code>	Syncing is when the image is syncing
<code>unknown</code>	<code>integer</code>	Unknown is when the mirroring state is unknown

`.status.mirroringStatus.summary.states`

Description

States is the various state for all mirrored images

Type

`object`

Property	Type	Description
<code>error</code>	<code>integer</code>	Error is when the mirroring state is errored
<code>replaying</code>	<code>integer</code>	Replaying is when the replay of the mirroring journal is on-going

Property	Type	Description
<code>starting_replay</code>	<code>integer</code>	StartingReplay is when the replay of the mirroring journal starts
<code>stopped</code>	<code>integer</code>	Stopped is when the mirroring state is stopped
<code>stopping_replay</code>	<code>integer</code>	StopReplaying is when the replay of the mirroring journal stops
<code>syncing</code>	<code>integer</code>	Syncing is when the image is syncing
<code>unknown</code>	<code>integer</code>	Unknown is when the mirroring state is unknown

.status.snapshotScheduleStatus

Description

SnapshotScheduleStatusSpec is the status of the snapshot schedule

Type

`object`

Property	Type	Description
<code>details</code>	<code>string</code>	Details contains potential status errors

Property	Type	Description
<code>lastChanged</code>	<code>string</code>	LastChanged is the last time time the status last changed
<code>lastChecked</code>	<code>string</code>	LastChecked is the last time time the status was checked
<code>snapshotSchedules</code>	<code>array</code>	SnapshotSchedules is the list of snapshots scheduled

.status.snapshotScheduleStatus.snapshotSchedules

Description

SnapshotSchedules is the list of snapshots scheduled

Type

`array`

.status.snapshotScheduleStatus.snapshotSchedules[]

Description

SnapshotSchedulesSpec is the list of snapshot scheduled for images in a pool

Type

`object`

Property	Type	Description
<code>image</code>	<code>string</code>	Image is the mirrored image

Property	Type	Description
<code>items</code>	<code>array</code>	Items is the list schedules times for a given snapshot
<code>namespace</code>	<code>string</code>	Namespace is the RADOS namespace the image is part of
<code>pool</code>	<code>string</code>	Pool is the pool name

`.status.snapshotScheduleStatus.snapshotSchedules[].items`

Description

Items is the list schedules times for a given snapshot

Type

`array`

`.status.snapshotScheduleStatus.snapshotSchedules[].items[]`

Description

SnapshotSchedule is a schedule

Type

`object`

Property	Type	Description
<code>interval</code>	<code>string</code>	Interval is the interval in which snapshots will be taken

Property	Type	Description
<code>start_time</code>	<code>string</code>	StartTime is the snapshot starting time

API Endpoints

The following API endpoints are available:

- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephblockpools`
 - `DELETE` : delete collection of CephBlockPool
 - `GET` : list objects of kind CephBlockPool
 - `POST` : create a new CephBlockPool
- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephblockpools/{name}`
 - `DELETE` : delete the specified CephBlockPool
 - `GET` : read the specified CephBlockPool
 - `PATCH` : partially update the specified CephBlockPool
 - `PUT` : replace the specified CephBlockPool
- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephblockpools/{name}/status`
 - `GET` : read status of the specified CephBlockPool
 - `PATCH` : partially update status of the specified CephBlockPool
 - `PUT` : replace status of the specified CephBlockPool

`/apis/ceph.rook.io/v1/namespaces/{namespace}/cephblockpools`

HTTP method

`DELETE`

Description

delete collection of CephBlockPool

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

GET

Description

list objects of kind CephBlockPool

HTTP responses

HTTP code	Response body
200 - OK	<code>CephBlockPoolList</code> schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a new CephBlockPool

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Parameter	Type	Description
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephBlockPool</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephBlockPool</code> schema
201 - Created	<code>CephBlockPool</code> schema
202 - Accepted	<code>CephBlockPool</code> schema
401 - Unauthorized	Empty

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephblockpools/{name}

HTTP method

DELETE

Description

delete the specified CephBlockPool

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
202 - Accepted	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

GET

Description

read the specified CephBlockPool

HTTP responses

HTTP code	Response body
200 - OK	<code>CephBlockPool</code> schema

HTTP code	Response body
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update the specified CephBlockPool

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephBlockPool</code> schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified CephBlockPool

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephBlockPool</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephBlockPool</code> schema
201 - Created	<code>CephBlockPool</code> schema
401 - Unauthorized	Empty

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephblockpools/{name}/status

HTTP method

`GET`

Description

read status of the specified CephBlockPool

HTTP responses

HTTP code	Response body
200 - OK	<code>CephBlockPool</code> schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update status of the specified CephBlockPool

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephBlockPool</code> schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace status of the specified CephBlockPool

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized <code>dryRun</code> directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a <code>BadRequest</code> error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephBlockPool</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephBlockPool</code> schema
201 - Created	<code>CephBlockPool</code> schema
401 - Unauthorized	Empty

CephObjectStore

[cephobjectstores.ceph.rook.io/v1]

Description

CephObjectStore represents a Ceph Object Store Gateway

Type

object

Required

metadata

spec

Specification

Property	Type	Description
apiVersion	string	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
kind	string	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be

Property	Type	Description
		updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
metadata	ObjectMeta	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
spec	object	ObjectStoreSpec represent the spec of a pool
status	object	ObjectStoreStatus represents the status of a Ceph Object Store resource

.spec

Description

ObjectStoreSpec represent the spec of a pool

Type

object

Property	Type	Description
allowUsersInNamespaces	array	The list of allowed namespaces in addition to the object store namespace where ceph object store users may be created. Specify "*" to allow all namespaces, otherwise list individual namespaces that are to be allowed. This is useful for applications that need object store credentials to be created in their own

Property	Type	Description
		namespace, where neither OBCs nor COSI is being used to create buckets. The default is empty.
<code>auth</code>	<code>object</code>	The authentication configuration
<code>dataPool</code>	<code>object</code>	The data pool settings
<code>defaultRealm</code>	<code>boolean</code>	Set this realm as the default in Ceph. Only one realm should be default. Do not set this true on more than one CephObjectStore. This may not be set when zone is also specified; in this case, the realm referenced by the zone's zonegroup should configure defaulting behavior.
<code>gateway</code>	<code>object</code>	The rgw pod info
<code>healthCheck</code>	<code>object</code>	The RGW health probes
<code>hosting</code>	<code>object</code>	Hosting settings for the object store. A common use case for hosting configuration is to inform Rook of endpoints that support DNS wildcards, which in turn allows virtual host-style bucket addressing.

Property	Type	Description
<code>metadataPool</code>	<code>object</code>	The metadata pool settings
<code>preservePoolsOnDelete</code>	<code>boolean</code>	Preserve pools on object store deletion
<code>protocols</code>	<code>object</code>	The protocol specification
<code>security</code>	<code>object</code>	Security represents security settings
<code>sharedPools</code>	<code>object</code>	The pool information when configuring RADOS namespaces in existing pools.
<code>zone</code>	<code>object</code>	The multisite info

.spec.allowUsersInNamespaces

Description

The list of allowed namespaces in addition to the object store namespace where ceph object store users may be created. Specify "*" to allow all namespaces, otherwise list individual namespaces that are to be allowed. This is useful for applications that need object store credentials to be created in their own namespace, where neither OBCs nor COSI is being used to create buckets. The default is empty.

Type

`array`

.spec.allowUsersInNamespaces[]

Type

string

.spec.auth

Description

The authentication configuration

Type

object

Property	Type	Description
keystone	object	The spec for Keystone

.spec.auth.keystone

Description

The spec for Keystone

Type

object

Required

acceptedRoles

serviceUserSecretName

url

Property	Type	Description
acceptedRoles	array	The roles requires to serve requests.

Property	Type	Description
<code>implicitTenants</code>	<code>string</code>	Create new users in their own tenants of the same name. Possible values are true, false, swift and s3. The latter have the effect of splitting the identity space such that only the indicated protocol will use implicit tenants.
<code>revocationInterval</code>	<code>integer</code>	The number of seconds between token revocation checks.
<code>serviceUserSecretName</code>	<code>string</code>	The name of the secret containing the credentials for the service user account used by RGW. It has to be in the same namespace as the object store resource.
<code>tokenCacheSize</code>	<code>integer</code>	The maximum number of entries in each Keystone token cache.
<code>url</code>	<code>string</code>	The URL for the Keystone server.

`.spec.auth.keystone.acceptedRoles`

Description

The roles requires to serve requests.

Type

`array`

.spec.auth.keystone.acceptedRoles[]

Type

`string`

.spec.dataPool

Description

The data pool settings

Type

`object`

Property	Type	Description
<code>application</code>	<code>string</code>	The application name to set on the pool. Only expected to be set for rgw pools.
<code>compressionMode</code>	<code>string</code>	DEPRECATED: use Parameters instead, e.g., Parameters["compression_mode"] = "force" The inline compression mode in Bluestore OSD to set to (options are: none, passive, aggressive, force) Do NOT set a default value for kubebuilder as this will override the Parameters
<code>crushRoot</code>	<code>string</code>	The root of the crush hierarchy utilized by the pool
<code>deviceClass</code>	<code>string</code>	The device class the OSD should set to for use in the pool

Property	Type	Description
<code>enableCrushUpdates</code>	<code>boolean</code>	Allow rook operator to change the pool CRUSH tunables once the pool is created
<code>enableRBDStats</code>	<code>boolean</code>	EnableRBDStats is used to enable gathering of statistics for all RBD images in the pool
<code>erasureCoded</code>	<code>object</code>	The erasure code settings
<code>failureDomain</code>	<code>string</code>	The failure domain: osd/host/(region or zone if available) - technically also any type in the crush map
<code>mirroring</code>	<code>object</code>	The mirroring settings
<code>parameters</code>	<code>object</code>	Parameters is a list of properties to enable on a given pool
<code>quotas</code>	<code>object</code>	The quota settings
<code>replicated</code>	<code>object</code>	The replication settings
<code>statusCheck</code>	<code>object</code>	The mirroring statusCheck

.spec.dataPool.erasureCoded

Description

The erasure code settings

Type

object

Required

codingChunks

dataChunks

Property	Type	Description
algorithm	string	The algorithm for erasure coding. If absent, defaults to the plugin specified in <code>osd_pool_default_erasure_code_profile</code> .
codingChunks	integer	Number of coding chunks per object in an erasure coded storage pool (required for erasure-coded pool type). This is the number of OSDs that can be lost simultaneously before data cannot be recovered.
dataChunks	integer	Number of data chunks per object in an erasure coded storage pool (required for erasure-coded pool type). The number of chunks required to recover an object when any single OSD is lost is the same as <code>dataChunks</code> so be aware that the larger the number of data chunks, the higher the cost of recovery.

.spec.dataPool.mirroring

Description

The mirroring settings

Type

object

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled whether this pool is mirrored or not
<code>mode</code>	<code>string</code>	Mode is the mirroring mode: pool, image or init-only.
<code>peers</code>	<code>object</code>	Peers represents the peers spec
<code>snapshotSchedules</code>	<code>array</code>	SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

.spec.dataPool.mirroring.peers

Description

Peers represents the peers spec

Type

object

Property	Type	Description
<code>secretNames</code>	<code>array</code>	SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

.spec.dataPool.mirroring.peers.secretNames

Description

SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

Type

array

.spec.dataPool.mirroring.peers.secretNames[]

Type

string

.spec.dataPool.mirroring.snapshotSchedules

Description

SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

Type

array

.spec.dataPool.mirroring.snapshotSchedules[]

Description

SnapshotScheduleSpec represents the snapshot scheduling settings of a mirrored pool

Type

object

Property	Type	Description
interval	string	Interval represent the periodicity of the snapshot.
path	string	Path is the path to snapshot, only valid for CephFS

Property	Type	Description
<code>startTime</code>	<code>string</code>	StartTime indicates when to start the snapshot

`.spec.dataPool.parameters`

Description

Parameters is a list of properties to enable on a given pool

Type

`object`

`.spec.dataPool.quotas`

Description

The quota settings

Type

`object`

Property	Type	Description
<code>maxBytes</code>	<code>integer</code>	MaxBytes represents the quota in bytes Deprecated in favor of MaxSize
<code>maxObjects</code>	<code>integer</code>	MaxObjects represents the quota in objects
<code>maxSize</code>	<code>string</code>	MaxSize represents the quota in bytes as a string

.spec.dataPool.replicated

Description

The replication settings

Type

object

Required

size

Property	Type	Description
hybridStorage	object	HybridStorage represents hybrid storage tier settings
replicasPerFailureDomain	integer	ReplicasPerFailureDomain the number of replica in the specified failure domain
requireSafeReplicaSize	boolean	RequireSafeReplicaSize if false allows you to set replica 1
size	integer	Size - Number of copies per object in a replicated storage pool, including the object itself (required for replicated pool type)
subFailureDomain	string	SubFailureDomain the name of the sub-failure domain

Property	Type	Description
<code>targetSizeRatio</code>	<code>number</code>	TargetSizeRatio gives a hint (%) to Ceph in terms of expected consumption of the total cluster capacity

`.spec.dataPool.replicated.hybridStorage`

Description

HybridStorage represents hybrid storage tier settings

Type

`object`

Required

`primaryDeviceClass`

`secondaryDeviceClass`

Property	Type	Description
<code>primaryDeviceClass</code>	<code>string</code>	PrimaryDeviceClass represents high performance tier (for example SSD or NVME) for Primary OSD
<code>secondaryDeviceClass</code>	<code>string</code>	SecondaryDeviceClass represents low performance tier (for example HDDs) for remaining OSDs

`.spec.dataPool.statusCheck`

Description

The mirroring statusCheck

Type

`object`

Property	Type	Description
<code>mirror</code>	<code>object</code>	HealthCheckSpec represents the health check of an object store bucket

`.spec.dataPool.statusCheck.mirror`

Description

HealthCheckSpec represents the health check of an object store bucket

Type

`object`

Property	Type	Description
<code>disabled</code>	<code>boolean</code>	
<code>interval</code>	<code>string</code>	Interval is the internal in second or minute for the health check to run like 60s for 60 seconds
<code>timeout</code>	<code>string</code>	

`.spec.gateway`

Description

The rgw pod info

Type

`object`

Property	Type	Description
<code>additionalVolumeMounts</code>	<code>array</code>	AdditionalVolumeMounts allows additional volumes to be mounted to the RGW pod. The root directory for each additional volume mount is <code>/var/rgw</code>. Example: for an additional mount at subPath <code>ldap</code>, mounted from a secret that has key <code>bindpass.secret</code>, the file would reside at <code>/var/rgw/ldap/bindpass.secret</code>.
<code>annotations</code>	<code>object</code>	The annotations-related configuration to add/set on each Pod related object.
<code>caBundleRef</code>	<code>string</code>	The name of the secret that stores custom ca-bundle with root and intermediate certificates.
<code>dashboardEnabled</code>	<code>boolean</code>	Whether rgw dashboard is enabled for the rgw daemon. If not set, the rgw dashboard will be enabled.
<code>disableMultisiteSyncTraffic</code>	<code>boolean</code>	DisableMultisiteSyncTraffic, when true, prevents this object store's gateways from transmitting multisite replication data. Note that this value does not affect whether gateways receive multisite replication traffic: see <code>ObjectZone.spec.customEndpoints</code> for

Property	Type	Description
		that. If false or unset, this object store's gateways will be able to transmit multisite replication data.
<code>externalRgwEndpoints</code>	<code>array</code>	ExternalRgwEndpoints points to external RGW endpoint(s). Multiple endpoints can be given, but for stability of ObjectBucketClaims, we highly recommend that users give only a single external RGW endpoint that is a load balancer that sends requests to the multiple RGWs.
<code>hostNetwork</code>	<code>boolean</code>	Whether host networking is enabled for the rgw daemon. If not set, the network settings from the cluster CR will be applied.
<code>instances</code>	<code>integer</code>	The number of pods in the rgw replicaset.
<code>labels</code>	<code>object</code>	The labels-related configuration to add/set on each Pod related object.
<code>opsLogSidecar</code>	<code>object</code>	Enable enhanced operation Logs for S3 in a sidecar named ops-log
<code>placement</code>	<code>object</code>	

Property	Type	Description
<code>port</code>	<code>integer</code>	The port the rgw service will be listening on (http)
<code>priorityClassName</code>	<code>string</code>	PriorityClassName sets priority classes on the rgw pods
<code>readAffinity</code>	<code>object</code>	ReadAffinity defines the RGW read affinity policy to optimize the read requests for the RGW clients Note: Only supported from Ceph Tentacle (v20)
<code>resources</code>	<code>object</code>	The resource requirements for the rgw pods
<code>rgwCommandFlags</code>	<code>object</code>	RgwCommandFlags sets Ceph RGW config values for the gateway clients that serve this object store. Values are modified at RGW startup, resulting in RGW pod restarts. This feature is intended for advanced users. It allows breaking configurations to be easily applied. Use with caution.
<code>rgwConfig</code>	<code>object</code>	RgwConfig sets Ceph RGW config values for the gateway clients that serve this object store. Values are modified at runtime without RGW restart. This

Property	Type	Description
		feature is intended for advanced users. It allows breaking configurations to be easily applied. Use with caution.
<code>rgwConfigFromSecret</code>	<code>object</code>	RgwConfigFromSecret works exactly like RgwConfig but takes config value from Secret Key reference. Values are modified at runtime without RGW restart. This feature is intended for advanced users. It allows breaking configurations to be easily applied. Use with caution.
<code>securePort</code>	<code>integer</code>	The port the rgw service will be listening on (https)
<code>service</code>	<code>object</code>	The configuration related to add/set on each rgw service.
<code>sslCertificateRef</code>	<code>string</code>	The name of the secret that stores the ssl certificate for secure rgw connections

.spec.gateway.additionalVolumeMounts

Description

AdditionalVolumeMounts allows additional volumes to be mounted to the RGW pod. The root directory for each additional volume mount is `/var/rgw`. Example: for an additional mount at subPath `ldap`, mounted from a secret that has key `bindpass.secret`, the file would reside at `/var/rgw/ldap/bindpass.secret`.

Type

array

.spec.gateway.additionalVolumeMounts[]

Description

AdditionalVolumeMount represents the source from where additional files in pod containers should come from and what subdirectory they are made available in.

Type

object

Required

subPath

volumeSource

Property	Type	Description
subPath	string	SubPath defines the sub-path (subdirectory) of the directory root where the volumeSource will be mounted. All files/keys in the volume source's volume will be mounted to the subdirectory. This is not the same as the Kubernetes <code>subPath</code> volume mount option. Each subPath definition must be unique and must not contain ':
volumeSource	object	

.spec.gateway.additionalVolumeMounts[].volumeSource

Type

object

Property	Type	Description
configMap	object	

Property	Type	Description
<code>emptyDir</code>	<code>object</code>	
<code>hostPath</code>	<code>object</code>	
<code>persistentVolumeClaim</code>	<code>object</code>	
<code>projected</code>	<code>object</code>	
<code>secret</code>	<code>object</code>	

`.spec.gateway.additionalVolumeMounts[].volumeSource.configMap`

Type

`object`

Property	Type	Description
<code>defaultMode</code>	<code>integer</code>	
<code>items</code>	<code>array</code>	
<code>name</code>	<code>string</code>	
<code>optional</code>	<code>boolean</code>	

`.spec.gateway.additionalVolumeMounts[].volumeSource.configMap.items`

Type

`array`

`.spec.gateway.additionalVolumeMounts[].volumeSource.configMap.items[]`

Type

`object`

Required

`key``path`

Property	Type	Description
<code>key</code>	<code>string</code>	
<code>mode</code>	<code>integer</code>	
<code>path</code>	<code>string</code>	

`.spec.gateway.additionalVolumeMounts[].volumeSource.emptyDir`

Type

`object`

Property	Type	Description
<code>medium</code>	<code>string</code>	
<code>sizeLimit</code>		

`.spec.gateway.additionalVolumeMounts[].volumeSource.hostPath`

Type

`object`

Required

path

Property	Type	Description
path	string	
type	string	

.spec.gateway.additionalVolumeMounts[].volumeSource.persistentVolumeClaim

Type

object

Required

claimName

Property	Type	Description
claimName	string	
readOnly	boolean	

.spec.gateway.additionalVolumeMounts[].volumeSource.projected

Type

object

Property	Type	Description
defaultMode	integer	
sources	array	

`.spec.gateway.additionalVolumeMounts[].volumeSource.p rojected.sources`

Type

array

`.spec.gateway.additionalVolumeMounts[].volumeSource.p rojected.sources[]`

Type

object

Property	Type	Description
<code>clusterTrustBundle</code>	object	
<code>configMap</code>	object	
<code>downwardAPI</code>	object	
<code>podCertificate</code>	object	
<code>secret</code>	object	
<code>serviceAccountToken</code>	object	

`.spec.gateway.additionalVolumeMounts[].volumeSource.p rojected.sources[].clusterTrustBundle`

Type

object

Required

path

Property	Type	Description
labelSelector	object	
name	string	
optional	boolean	
path	string	
signerName	string	

.spec.gateway.additionalVolumeMounts[].volumeSource.projected.sources[].clusterTrustBundle.labelSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.gateway.additionalVolumeMounts[].volumeSource.projected.sources[].clusterTrustBundle.labelSelector.matchExpressions

Type

array

.spec.gateway.additionalVolumeMounts[].volumeSource.projected.sources[].clusterTrustBundle.labelSelector.match

Expressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].clusterTrustBundle.labelSelector.match
Expressions[].values**

Type

array

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].clusterTrustBundle.labelSelector.match
Expressions[].values[]**

Type

string

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].clusterTrustBundle.labelSelector.match**

Labels

Type

object

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].configMap**

Type

object

Property	Type	Description
items	array	
name	string	
optional	boolean	

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].configMap.items**

Type

array

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].configMap.items[]**

Type

object

Required

key

path

Property	Type	Description
key	string	
mode	integer	
path	string	

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].downwardAPI**

Type

object

Property	Type	Description
items	array	

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].downwardAPI.items**

Type

array

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].downwardAPI.items[]**

Type

object

Required

path

Property	Type	Description
fieldRef	object	
mode	integer	
path	string	
resourceFieldRef	object	

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].downwardAPI.items[].fieldRef**

Type

object

Required

fieldPath

Property	Type	Description
apiVersion	string	
fieldPath	string	

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].downwardAPI.items[].resourceFieldRef**

Type

object

Required

resource

Property	Type	Description
containerName	string	
divisor		
resource	string	

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].podCertificate**

Type

object

Required

keyType signerName

Property	Type	Description
certificateChainPath	string	
credentialBundlePath	string	
keyPath	string	
keyType	string	
maxExpirationSeconds	integer	
signerName	string	
userAnnotations	object	

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].podCertificate.userAnnotations**

Type

object

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].secret**

Type

object

Property	Type	Description
items	array	
name	string	
optional	boolean	

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].secret.items**

Type

array

**.spec.gateway.additionalVolumeMounts[].volumeSource.p
rojected.sources[].secret.items[]**

Type

object

Required

key

path

Property	Type	Description
key	string	

Property	Type	Description
mode	integer	
path	string	

`.spec.gateway.additionalVolumeMounts[].volumeSource.projected.sources[].serviceAccountToken`

Type

object

Required

path

Property	Type	Description
audience	string	
expirationSeconds	integer	
path	string	

`.spec.gateway.additionalVolumeMounts[].volumeSource.secret`

Type

object

Property	Type	Description
defaultMode	integer	
items	array	

Property	Type	Description
<code>optional</code>	<code>boolean</code>	
<code>secretName</code>	<code>string</code>	

`.spec.gateway.additionalVolumeMounts[].volumeSource.secret.items`

Type

`array`

`.spec.gateway.additionalVolumeMounts[].volumeSource.secret.items[]`

Type

`object`

Required

`key``path`

Property	Type	Description
<code>key</code>	<code>string</code>	
<code>mode</code>	<code>integer</code>	
<code>path</code>	<code>string</code>	

`.spec.gateway.annotations`

Description

The annotations-related configuration to add/set on each Pod related object.

Type

object

.spec.gateway.externalRgwEndpoints

Description

ExternalRgwEndpoints points to external RGW endpoint(s). Multiple endpoints can be given, but for stability of ObjectBucketClaims, we highly recommend that users give only a single external RGW endpoint that is a load balancer that sends requests to the multiple RGWs.

Type

array

.spec.gateway.externalRgwEndpoints[]

Description

EndpointAddress is a tuple that describes a single IP address or host name. This is a subset of Kubernetes's v1.EndpointAddress.

Type

object

Property	Type	Description
hostname	string	The DNS-addressable Hostname of this endpoint. This field will be preferred over IP if both are given.
ip	string	The IP of this endpoint. As a legacy behavior, this supports being given a DNS-addressable hostname as well.

.spec.gateway.labels

Description

The labels-related configuration to add/set on each Pod related object.

Type

object

.spec.gateway.opsLogSidecar

Description

Enable enhanced operation Logs for S3 in a sidecar named ops-log

Type

object

Property	Type	Description
<code>resources</code>	object	Resources represents the way to specify resource requirements for the ops-log sidecar

.spec.gateway.opsLogSidecar.resources

Description

Resources represents the way to specify resource requirements for the ops-log sidecar

Type

object

Property	Type	Description
		Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container.
claims	array	This field depends on the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

.spec.gateway.opsLogSidecar.resources.claims

Description

Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This field depends on the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.

Type

array

.spec.gateway.opsLogSidecar.resources.claims[]

Description

ResourceClaim references one entry in PodSpec.ResourceClaims.

Type

object

Required

name

Property	Type	Description
name	string	Name must match the name of one entry in pod.spec.resourceClaims of the Pod where this field is used. It makes that resource available inside a container.
request	string	Request is the name chosen for a request in the referenced claim. If empty, everything from the claim is made available, otherwise only the result of this request.

.spec.gateway.opsLogSidecar.resources.limits

Description

Limits describes the maximum amount of compute resources allowed. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

.spec.gateway.opsLogSidecar.resources.requests

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info:

<https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

.spec.gateway.placement

Type

object

Property	Type	Description
nodeAffinity	object	
podAffinity	object	
podAntiAffinity	object	
tolerations	array	
topologySpreadConstraints	array	

.spec.gateway.placement.nodeAffinity

Type

object

Property	Type	Description
preferredDuringSchedulingIgnoredDuringExecution	array	
requiredDuringSchedulingIgnoredDuringExecution	object	

`.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution`

Type

array

`.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[]`

Type

object

Required

preference

weight

Property	Type	Description
preference	object	
weight	integer	

`.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference`

Type

object

Property	Type	Description
matchExpressions	array	
matchFields	array	

`.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions`

Type

array

`.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[]`

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[].values`

Type

array

.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchExpressions[].values[]

Type

string

.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields

Type

array

.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[].values

Type

array

.spec.gateway.placement.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution[].preference.matchFields[].values[]

Type

string

.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution

Type

object

Required

nodeSelectorTerms

Property	Type	Description
nodeSelectorTerms	array	

.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms

Type

array

`.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[]`

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchFields</code>	array	

`.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions`

Type

array

`.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions[]`

Type

object

Required

key

operator

Property	Type	Description
<code>key</code>	string	

Property	Type	Description
<code>operator</code>	<code>string</code>	
<code>values</code>	<code>array</code>	

`.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions[].values`

Type

`array`

`.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchExpressions[].values[]`

Type

`string`

`.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields`

Type

`array`

`.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields[]`

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields[].values

Type

array

.spec.gateway.placement.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms[].matchFields[].values[]

Type

string

.spec.gateway.placement.podAffinity

Type

object

Property	Type	Description
<code>preferredDuringSchedulingIgnoredDuringExecution</code>	<code>array</code>	
<code>requiredDuringSchedulingIgnoredDuringExecution</code>	<code>array</code>	

`.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution`

Type

`array`

`.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[]`

Type

`object`

Required

`podAffinityTerm`

`weight`

Property	Type	Description
<code>podAffinityTerm</code>	<code>object</code>	
<code>weight</code>	<code>integer</code>	

`.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm`

Type

`object`

Required

`topologyKey`

Property	Type	Description
<code>labelSelector</code>	<code>object</code>	
<code>matchLabelKeys</code>	<code>array</code>	
<code>mismatchLabelKeys</code>	<code>array</code>	
<code>namespaceSelector</code>	<code>object</code>	
<code>namespaces</code>	<code>array</code>	
<code>topologyKey</code>	<code>string</code>	

`.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector`

Type

`object`

Property	Type	Description
<code>matchExpressions</code>	<code>array</code>	
<code>matchLabels</code>	<code>object</code>	

`.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions`

Type

`array`

`.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[]`

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values`

Type

array

`.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values[]`

Type

string

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchLabels

Type

object

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys

Type

array

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys[]

Type

string

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys

Type

array

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys[]

Type

string

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions

Type

array

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values

Type

array

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchExpressions[].values[]

Type

string

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaceSelector.matchLabels

Type

object

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces

Type

array

.spec.gateway.placement.podAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces[]

Type

string

.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution

Type

array

.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[]

Type

object

Required

topologyKey

Property	Type	Description
<code>labelSelector</code>	<code>object</code>	
<code>matchLabelKeys</code>	<code>array</code>	
<code>mismatchLabelKeys</code>	<code>array</code>	
<code>namespaceSelector</code>	<code>object</code>	
<code>namespaces</code>	<code>array</code>	
<code>topologyKey</code>	<code>string</code>	

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector`

Type

`object`

Property	Type	Description
<code>matchExpressions</code>	<code>array</code>	
<code>matchLabels</code>	<code>object</code>	

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions`

Type

`array`

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[]`

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values`

Type

array

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values[]`

Type

string

.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchLabels

Type

object

.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys

Type

array

.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys[]

Type

string

.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys

Type

array

.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys[]

Type

string

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector`

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions`

Type

array

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[]`

Type

object

Required

key operator

Property	Type	Description
<code>key</code>	string	

Property	Type	Description
<code>operator</code>	<code>string</code>	
<code>values</code>	<code>array</code>	

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values`

Type

`array`

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values[]`

Type

`string`

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchLabels`

Type

`object`

`.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces`

Type

`array`

.spec.gateway.placement.podAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces[]

Type

`string`

.spec.gateway.placement.podAntiAffinity

Type

`object`

Property	Type	Description
<code>preferredDuringSchedulingIgnoredDuringExecution</code>	<code>array</code>	
<code>requiredDuringSchedulingIgnoredDuringExecution</code>	<code>array</code>	

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution

Type

`array`

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[]

Type

`object`

Required

`podAffinityTerm``weight`

Property	Type	Description
podAffinityTerm	object	
weight	integer	

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm

Type

object

Required

topologyKey

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
mismatchLabelKeys	array	
namespaceSelector	object	
namespaces	array	
topologyKey	string	

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector

Type

object

Property	Type	Description
matchExpressions	array	
matchLabels	object	

`.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions`

Type

array

`.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[]`

Type

object

Required

key operator

Property	Type	Description
key	string	
operator	string	
values	array	

`.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.lab`

`elSelector.matchExpressions[].values`

Type

array

`.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchExpressions[].values[]`

Type

string

`.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.labelSelector.matchLabels`

Type

object

`.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys`

Type

array

`.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.matchLabelKeys[]`

Type

string

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys

Type

array

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.mismatchLabelKeys[]

Type

string

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchExpressions

Type

array

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchExpressions[].values

Type

array

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchExpressions[].values[]

Type

string

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces.selector.matchLabels

Type

object

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces

Type

array

.spec.gateway.placement.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution[].podAffinityTerm.namespaces[]

Type

string

`.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution`

Type

array

`.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[]`

Type

object

Required

topologyKey

Property	Type	Description
<code>labelSelector</code>	object	
<code>matchLabelKeys</code>	array	
<code>mismatchLabelKeys</code>	array	
<code>namespaceSelector</code>	object	
<code>namespaces</code>	array	
<code>topologyKey</code>	string	

`.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector`

Type

object

Property	Type	Description
<code>matchExpressions</code>	<code>array</code>	
<code>matchLabels</code>	<code>object</code>	

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions

Type

`array`

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[]

Type

`object`

Required

`key`

`operator`

Property	Type	Description
<code>key</code>	<code>string</code>	
<code>operator</code>	<code>string</code>	
<code>values</code>	<code>array</code>	

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchE

xpressions[].values

Type

array

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchExpressions[].values[]

Type

string

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].labelSelector.matchLabels

Type

object

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys

Type

array

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].matchLabelKeys[]

Type

string

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys

Type

array

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].mismatchLabelKeys[]

Type

string

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector

Type

object

Property	Type	Description
<code>matchExpressions</code>	array	
<code>matchLabels</code>	object	

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions

Type

array

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[]

Type

object

Required

key

operator

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values

Type

array

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchExpressions[].values[]

Type

string

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaceSelector.matchLabels

Type

object

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces

Type

array

.spec.gateway.placement.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution[].namespaces[]

Type

string

.spec.gateway.placement.tolerations

Type

array

.spec.gateway.placement.tolerations[]

Type

object

Property	Type	Description
effect	string	
key	string	
operator	string	
tolerationSeconds	integer	
value	string	

`.spec.gateway.placement.topologySpreadConstraints`

Type

array

`.spec.gateway.placement.topologySpreadConstraints[]`

Type

object

Required

maxSkew

topologyKey

whenUnsatisfiable

Property	Type	Description
labelSelector	object	
matchLabelKeys	array	
maxSkew	integer	
minDomains	integer	
nodeAffinityPolicy	string	

Property	Type	Description
<code>nodeTaintsPolicy</code>	<code>string</code>	
<code>topologyKey</code>	<code>string</code>	
<code>whenUnsatisfiable</code>	<code>string</code>	

`.spec.gateway.placement.topologySpreadConstraints[].labelSelector`

Type

`object`

Property	Type	Description
<code>matchExpressions</code>	<code>array</code>	
<code>matchLabels</code>	<code>object</code>	

`.spec.gateway.placement.topologySpreadConstraints[].labelSelector.matchExpressions`

Type

`array`

`.spec.gateway.placement.topologySpreadConstraints[].labelSelector.matchExpressions[]`

Type

`object`

Required

`key` `operator`

Property	Type	Description
key	string	
operator	string	
values	array	

.spec.gateway.placement.topologySpreadConstraints[].labelSelector.matchExpressions[].values

Type

array

.spec.gateway.placement.topologySpreadConstraints[].labelSelector.matchExpressions[].values[]

Type

string

.spec.gateway.placement.topologySpreadConstraints[].labelSelector.matchLabels

Type

object

.spec.gateway.placement.topologySpreadConstraints[].matchLabelKeys

Type

array

.spec.gateway.placement.topologySpreadConstraints[].matchLabelKeys[]

Type

string

.spec.gateway.readAffinity

Description

ReadAffinity defines the RGW read affinity policy to optimize the read requests for the RGW clients Note: Only supported from Ceph Tentacle (v20)

Type

object

Required

type

Property	Type	Description
type	string	Type defines the RGW ReadAffinity type localize: read from the nearest OSD based on crush location of the RGW client balance: picks a random OSD from the PG's active set default: read from the primary OSD

.spec.gateway.resources

Description

The resource requirements for the rgw pods

Type

object

Property	Type	Description
		Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container.
claims	array	This field depends on the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗
requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ ↗

.spec.gateway.resources.claims

Description

Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This field depends on the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.

Type

array

.spec.gateway.resources.claims[]

Description

ResourceClaim references one entry in PodSpec.ResourceClaims.

Type

object

Required

name

Property	Type	Description
name	string	Name must match the name of one entry in pod.spec.resourceClaims of the Pod where this field is used. It makes that resource available inside a container.
request	string	Request is the name chosen for a request in the referenced claim. If empty, everything from the claim is made available, otherwise only the result of this request.

.spec.gateway.resources.limits

Description

Limits describes the maximum amount of compute resources allowed. More info: <https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

.spec.gateway.resources.requests

Description

Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info:

<https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/>

Type

object

.spec.gateway.rgwCommandFlags

Description

RgwCommandFlags sets Ceph RGW config values for the gateway clients that serve this object store. Values are modified at RGW startup, resulting in RGW pod restarts. This feature is intended for advanced users. It allows breaking configurations to be easily applied. Use with caution.

Type

object

.spec.gateway.rgwConfig

Description

RgwConfig sets Ceph RGW config values for the gateway clients that serve this object store. Values are modified at runtime without RGW restart. This feature is intended for advanced users. It allows breaking configurations to be easily applied. Use with caution.

Type

object

.spec.gateway.rgwConfigFromSecret

Description

RgwConfigFromSecret works exactly like RgwConfig but takes config value from Secret Key reference. Values are modified at runtime without RGW restart. This feature is intended for advanced users. It allows breaking configurations to be easily applied. Use with caution.

Type

`object`

.spec.gateway.service

Description

The configuration related to add/set on each rgw service.

Type

`object`

Property	Type	Description
<code>annotations</code>	<code>object</code>	The annotations-related configuration to add/set on each rgw service. nullable optional

.spec.gateway.service.annotations

Description

The annotations-related configuration to add/set on each rgw service. nullable optional

Type

`object`

.spec.healthCheck

Description

The RGW health probes

Type

`object`

Property	Type	Description
<code>readinessProbe</code>	<code>object</code>	ProbeSpec is a wrapper around Probe so it can be enabled or disabled for a Ceph daemon
<code>startupProbe</code>	<code>object</code>	ProbeSpec is a wrapper around Probe so it can be enabled or disabled for a Ceph daemon

`.spec.healthCheck.readinessProbe`

Description

ProbeSpec is a wrapper around Probe so it can be enabled or disabled for a Ceph daemon

Type

`object`

Property	Type	Description
<code>disabled</code>	<code>boolean</code>	Disabled determines whether probe is disable or not
<code>probe</code>	<code>object</code>	Probe describes a health check to be performed against a container to determine whether it is alive or ready to receive traffic.

`.spec.healthCheck.readinessProbe.probe`

Description

Probe describes a health check to be performed against a container to determine whether it is alive or ready to receive traffic.

Type

object

Property	Type	Description
<code>exec</code>	object	Exec specifies a command to execute in the
<code>failureThreshold</code>	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Default value is 3. Minimum value is 1.
<code>grpc</code>	object	GRPC specifies a GRPC HealthCheckRequest to
<code>httpGet</code>	object	HTTPGet specifies an HTTP GET request to
<code>initialDelaySeconds</code>	integer	Number of seconds after the container has started before liveness probes are initiated. More info: https://kubernetes.io/docs/concepts/workloads/pods/lifecycle#container-probes
<code>periodSeconds</code>	integer	How often (in seconds) to perform the probe. Default is 10 seconds. Minimum value is 1.
<code>successThreshold</code>	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Default is 1. Must be 1 for liveness and startup. Minimum value is 1.

Property	Type	Description
<code>tcpSocket</code>	<code>object</code>	TCPSocket specifies a connection to a TCP
<code>terminationGracePeriodSeconds</code>	<code>integer</code>	
<code>timeoutSeconds</code>	<code>integer</code>	Number of seconds after which the probe tin Defaults to 1 second. Minimum value is 1. M https://kubernetes.io/docs/concepts/workloads/lifecycle#container-probes ↗

`.spec.healthCheck.readinessProbe.probe.exec`

Description

Exec specifies a command to execute in the container.

Type

`object`

Property	Type	Description
<code>command</code>	<code>array</code>	Command is the command line to execute inside the container, the working directory for the command is root ('/') in the container's filesystem. The command is simply exec'd, it is not run inside a shell, so traditional shell instructions (' ', etc) won't work. To use a shell, you need to explicitly call out to that shell. Exit status of 0 is treated as live/healthy and non-zero is unhealthy.

`.spec.healthCheck.readinessProbe.probe.exec.command`

Description

Command is the command line to execute inside the container, the working directory for the command is root ('/') in the container's filesystem. The command is simply exec'd, it is not run inside a shell, so traditional shell instructions ('|', etc) won't work. To use a shell, you need to explicitly call out to that shell. Exit status of 0 is treated as live/healthy and non-zero is unhealthy.

Type

array

.spec.healthCheck.readinessProbe.probe.exec.command[]

Type

string

.spec.healthCheck.readinessProbe.probe.grpc

Description

GRPC specifies a GRPC HealthCheckRequest.

Type

object

Required

port

Property	Type	Description
port	integer	Port number of the gRPC service. Number must be in the range 1 to 65535.
service	string	Service is the name of the service to place in the gRPC HealthCheckRequest (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md ↗).

Property	Type	Description
If this is not specified, the default behavior is defined by gRPC.		

.spec.healthCheck.readinessProbe.probe.httpGet

Description

HTTPGet specifies an HTTP GET request to perform.

Type

object

Required

port

Property	Type	Description
host	string	Host name to connect to, defaults to the pod IP. You probably want to set "Host" in httpHeaders instead.
httpHeaders	array	Custom headers to set in the request. HTTP allows repeated headers.
path	string	Path to access on the HTTP server.
port		Name or number of the port to access on the container. Number must be in the range 1 to 65535. Name must be an IANA_SVC_NAME.

Property	Type	Description
<code>scheme</code>	<code>string</code>	Scheme to use for connecting to the host. Defaults to HTTP.

`.spec.healthCheck.readinessProbe.probe.httpGet.httpHeaders`

Description

Custom headers to set in the request. HTTP allows repeated headers.

Type

`array`

`.spec.healthCheck.readinessProbe.probe.httpGet.httpHeaders[]`

Description

HTTPHeader describes a custom header to be used in HTTP probes

Type

`object`

Required

`name`

`value`

Property	Type	Description
<code>name</code>	<code>string</code>	The header field name. This will be canonicalized upon output, so case-variant names will be understood as the same header.
<code>value</code>	<code>string</code>	The header field value

.spec.healthCheck.readinessProbe.probe.tcpSocket

Description

TCPSocket specifies a connection to a TCP port.

Type

object

Required

port

Property	Type	Description
host	string	Optional: Host name to connect to, defaults to the pod IP.
port		Number or name of the port to access on the container. Number must be in the range 1 to 65535. Name must be an IANA_SVC_NAME.

.spec.healthCheck.startupProbe

Description

ProbeSpec is a wrapper around Probe so it can be enabled or disabled for a Ceph daemon

Type

object

Property	Type	Description
disabled	boolean	Disabled determines whether probe is disable or not

Property	Type	Description
<code>probe</code>	<code>object</code>	Probe describes a health check to be performed against a container to determine whether it is alive or ready to receive traffic.

`.spec.healthCheck.startupProbe.probe`

Description

Probe describes a health check to be performed against a container to determine whether it is alive or ready to receive traffic.

Type

`object`

Property	Type	Description
<code>exec</code>	<code>object</code>	Exec specifies a command to execute in the
<code>failureThreshold</code>	<code>integer</code>	Minimum consecutive failures for the probe to be considered failed after having succeeded. Default value is 3. Minimum value is 1.
<code>grpc</code>	<code>object</code>	GRPC specifies a GRPC HealthCheckRequest to send to the container.
<code>httpGet</code>	<code>object</code>	HTTPGet specifies an HTTP GET request to send to the container.
<code>initialDelaySeconds</code>	<code>integer</code>	Number of seconds after the container has started before liveness probes are initiated. More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes

Property	Type	Description
		https://kubernetes.io/docs/concepts/workload-lifecycle#container-probes ↗
<code>periodSeconds</code>	<code>integer</code>	How often (in seconds) to perform the probe 10 seconds. Minimum value is 1.
<code>successThreshold</code>	<code>integer</code>	Minimum consecutive successes for the probe to be considered successful after having failed. Default is 1. Must be 1 for liveness and startup. Minimum
<code>tcpSocket</code>	<code>object</code>	TCPSocket specifies a connection to a TCP
<code>terminationGracePeriodSeconds</code>	<code>integer</code>	
<code>timeoutSeconds</code>	<code>integer</code>	Number of seconds after which the probe times out. Defaults to 1 second. Minimum value is 1. Must be greater than 0. https://kubernetes.io/docs/concepts/workload-lifecycle#container-probes ↗

`.spec.healthCheck.startupProbe.probe.exec`

Description

Exec specifies a command to execute in the container.

Type

`object`

Property	Type	Description
<code>command</code>	<code>array</code>	Command is the command line to execute inside the container, the working directory for the command is root ('/') in the container's filesystem. The command is simply exec'd, it is not run inside a shell, so traditional shell instructions (' ', etc) won't work. To use a shell, you need to explicitly call out to that shell. Exit status of 0 is treated as live/healthy and non-zero is unhealthy.

`.spec.healthCheck.startupProbe.probe.exec.command`

Description

Command is the command line to execute inside the container, the working directory for the command is root ('/') in the container's filesystem. The command is simply exec'd, it is not run inside a shell, so traditional shell instructions ('|', etc) won't work. To use a shell, you need to explicitly call out to that shell. Exit status of 0 is treated as live/healthy and non-zero is unhealthy.

Type

`array`

`.spec.healthCheck.startupProbe.probe.exec.command[]`

Type

`string`

`.spec.healthCheck.startupProbe.probe.grpc`

Description

GRPC specifies a GRPC HealthCheckRequest.

Type

`object`

Required

port

Property	Type	Description
port	integer	Port number of the gRPC service. Number must be in the range 1 to 65535.
service	string	Service is the name of the service to place in the gRPC HealthCheckRequest (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md ↗). If this is not specified, the default behavior is defined by gRPC.

.spec.healthCheck.startupProbe.probe.httpGet

Description

HTTPGet specifies an HTTP GET request to perform.

Type

object

Required

port

Property	Type	Description
host	string	Host name to connect to, defaults to the pod IP. You probably want to set "Host" in httpHeaders instead.

Property	Type	Description
<code>httpHeaders</code>	<code>array</code>	Custom headers to set in the request. HTTP allows repeated headers.
<code>path</code>	<code>string</code>	Path to access on the HTTP server.
<code>port</code>	<code></code>	Name or number of the port to access on the container. Number must be in the range 1 to 65535. Name must be an IANA_SVC_NAME.
<code>scheme</code>	<code>string</code>	Scheme to use for connecting to the host. Defaults to HTTP.

`.spec.healthCheck.startupProbe.probe.httpGet.httpHeaderS`

Description

Custom headers to set in the request. HTTP allows repeated headers.

Type

`array`

`.spec.healthCheck.startupProbe.probe.httpGet.httpHeaderS[]`

Description

HTTPHeader describes a custom header to be used in HTTP probes

Type

`object`

Required

`name``value`

Property	Type	Description
<code>name</code>	<code>string</code>	The header field name. This will be canonicalized upon output, so case-variant names will be understood as the same header.
<code>value</code>	<code>string</code>	The header field value

.spec.healthCheck.startupProbe.probe.tcpSocket

Description

TCPSocket specifies a connection to a TCP port.

Type

`object`

Required

`port`

Property	Type	Description
<code>host</code>	<code>string</code>	Optional: Host name to connect to, defaults to the pod IP.
<code>port</code>		Number or name of the port to access on the container. Number must be in the range 1 to 65535. Name must be an IANA_SVC_NAME.

.spec.hosting

Description

Hosting settings for the object store. A common use case for hosting configuration is to inform Rook of endpoints that support DNS wildcards, which in turn allows virtual host-style bucket addressing.

Type

object

Property	Type	Description
<code>advertiseEndpoint</code>	<code>object</code>	AdvertiseEndpoint is the default endpoint Rook will return for resources dependent on this object store. This endpoint will be returned to CephObjectStoreUsers, Object Bucket Claims, and COSI Buckets/Accesses. By default, Rook returns the endpoint for the object store's Kubernetes service using HTTPS with <code>gateway.securePort</code> if it is defined (otherwise, HTTP with <code>gateway.port</code>).

Property	Type	Description
<code>dnsNames</code>	<code>array</code>	A list of DNS host names on which object store gateways will accept client S3 connections. When specified, object store gateways will reject client S3 connections to hostnames that are not present in this list, so include all endpoints. The object store's <code>advertiseEndpoint</code> and Kubernetes service endpoint, plus <code>CephObjectZone</code> <code>customEndpoints</code> are automatically added to the list but may be set here again if desired. Each DNS name must be valid according RFC-1123. If the DNS name corresponds to an endpoint with DNS wildcard support, do not include the wildcard itself in the list of hostnames. E.g., use "mystore.example.com" instead of "*.mystore.example.com".

.spec.hosting.advertiseEndpoint

Description

`AdvertiseEndpoint` is the default endpoint Rook will return for resources dependent on this object store. This endpoint will be returned to `CephObjectStoreUsers`, `Object Bucket Claims`, and `COSI Buckets/Accesses`. By default, Rook returns the endpoint for the object store's Kubernetes service using HTTPS with ``gateway.securePort`` if it is defined (otherwise, HTTP with ``gateway.port``).

Type

`object`

Required

`dnsName``port``useTls`

Property	Type	Description
<code>dnsName</code>	<code>string</code>	DnsName is the DNS name (in RFC-1123 format) of the endpoint. If the DNS name corresponds to an endpoint with DNS wildcard support, do not include the wildcard itself in the list of hostnames. E.g., use "mystore.example.com" instead of "*.mystore.example.com".
<code>port</code>	<code>integer</code>	Port is the port on which S3 connections can be made for this endpoint.
<code>useTls</code>	<code>boolean</code>	UseTls defines whether the endpoint uses TLS (HTTPS) or not (HTTP).

`.spec.hosting.dnsNames`

Description

A list of DNS host names on which object store gateways will accept client S3 connections. When specified, object store gateways will reject client S3 connections to hostnames that are not present in this list, so include all endpoints. The object store's advertiseEndpoint and Kubernetes service endpoint, plus CephObjectZone `customEndpoints` are automatically added to the list but may be set here again if desired. Each DNS name must be valid according RFC-1123. If the DNS name corresponds to an endpoint with DNS wildcard support, do not include the wildcard itself in the list of hostnames. E.g., use "mystore.example.com" instead of "*.mystore.example.com".

Type

`array`

`.spec.hosting.dnsNames[]`

Type

`string`

.spec.metadataPool

Description

The metadata pool settings

Type

`object`

Property	Type	Description
<code>application</code>	<code>string</code>	The application name to set on the pool. Only expected to be set for rgw pools.
<code>compressionMode</code>	<code>string</code>	DEPRECATED: use Parameters instead, e.g., Parameters["compression_mode"] = "force" The inline compression mode in Bluestore OSD to set to (options are: none, passive, aggressive, force) Do NOT set a default value for kubebuilder as this will override the Parameters
<code>crushRoot</code>	<code>string</code>	The root of the crush hierarchy utilized by the pool
<code>deviceClass</code>	<code>string</code>	The device class the OSD should set to for use in the pool
<code>enableCrushUpdates</code>	<code>boolean</code>	Allow rook operator to change the pool CRUSH tunables once the pool is created

Property	Type	Description
<code>enableRBDStats</code>	<code>boolean</code>	EnableRBDStats is used to enable gathering of statistics for all RBD images in the pool
<code>erasureCoded</code>	<code>object</code>	The erasure code settings
<code>failureDomain</code>	<code>string</code>	The failure domain: osd/host/(region or zone if available) - technically also any type in the crush map
<code>mirroring</code>	<code>object</code>	The mirroring settings
<code>parameters</code>	<code>object</code>	Parameters is a list of properties to enable on a given pool
<code>quotas</code>	<code>object</code>	The quota settings
<code>replicated</code>	<code>object</code>	The replication settings
<code>statusCheck</code>	<code>object</code>	The mirroring statusCheck

`.spec.metadataPool.erasureCoded`

Description

The erasure code settings

Type

object

Required

codingChunks

dataChunks

Property	Type	Description
algorithm	string	The algorithm for erasure coding. If absent, defaults to the plugin specified in <code>osd_pool_default_erasure_code_profile</code> .
codingChunks	integer	Number of coding chunks per object in an erasure coded storage pool (required for erasure-coded pool type). This is the number of OSDs that can be lost simultaneously before data cannot be recovered.
dataChunks	integer	Number of data chunks per object in an erasure coded storage pool (required for erasure-coded pool type). The number of chunks required to recover an object when any single OSD is lost is the same as <code>dataChunks</code> so be aware that the larger the number of data chunks, the higher the cost of recovery.

.spec.metadataPool.mirroring

Description

The mirroring settings

Type

object

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled whether this pool is mirrored or not
<code>mode</code>	<code>string</code>	Mode is the mirroring mode: pool, image or init-only.
<code>peers</code>	<code>object</code>	Peers represents the peers spec
<code>snapshotSchedules</code>	<code>array</code>	SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

`.spec.metadataPool.mirroring.peers`

Description

Peers represents the peers spec

Type

`object`

Property	Type	Description
<code>secretNames</code>	<code>array</code>	SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

`.spec.metadataPool.mirroring.peers.secretNames`

Description

SecretNames represents the Kubernetes Secret names to add rbd-mirror or cephfs-mirror peers

Type

array

.spec.metadataPool.mirroring.peers.secretNames[]

Type

string

.spec.metadataPool.mirroring.snapshotSchedules

Description

SnapshotSchedules is the scheduling of snapshot for mirrored images/pools

Type

array

.spec.metadataPool.mirroring.snapshotSchedules[]

Description

SnapshotScheduleSpec represents the snapshot scheduling settings of a mirrored pool

Type

object

Property	Type	Description
interval	string	Interval represent the periodicity of the snapshot.
path	string	Path is the path to snapshot, only valid for CephFS

Property	Type	Description
<code>startTime</code>	<code>string</code>	StartTime indicates when to start the snapshot

`.spec.metadataPool.parameters`

Description

Parameters is a list of properties to enable on a given pool

Type

`object`

`.spec.metadataPool.quotas`

Description

The quota settings

Type

`object`

Property	Type	Description
<code>maxBytes</code>	<code>integer</code>	MaxBytes represents the quota in bytes Deprecated in favor of MaxSize
<code>maxObjects</code>	<code>integer</code>	MaxObjects represents the quota in objects
<code>maxSize</code>	<code>string</code>	MaxSize represents the quota in bytes as a string

.spec.metadataPool.replicated

Description

The replication settings

Type

object

Required

size

Property	Type	Description
hybridStorage	object	HybridStorage represents hybrid storage tier settings
replicasPerFailureDomain	integer	ReplicasPerFailureDomain the number of replica in the specified failure domain
requireSafeReplicaSize	boolean	RequireSafeReplicaSize if false allows you to set replica 1
size	integer	Size - Number of copies per object in a replicated storage pool, including the object itself (required for replicated pool type)
subFailureDomain	string	SubFailureDomain the name of the sub-failure domain

Property	Type	Description
<code>targetSizeRatio</code>	<code>number</code>	TargetSizeRatio gives a hint (%) to Ceph in terms of expected consumption of the total cluster capacity

`.spec.metadataPool.replicated.hybridStorage`

Description

HybridStorage represents hybrid storage tier settings

Type

`object`

Required

`primaryDeviceClass`

`secondaryDeviceClass`

Property	Type	Description
<code>primaryDeviceClass</code>	<code>string</code>	PrimaryDeviceClass represents high performance tier (for example SSD or NVME) for Primary OSD
<code>secondaryDeviceClass</code>	<code>string</code>	SecondaryDeviceClass represents low performance tier (for example HDDs) for remaining OSDs

`.spec.metadataPool.statusCheck`

Description

The mirroring statusCheck

Type

`object`

Property	Type	Description
<code>mirror</code>	<code>object</code>	HealthCheckSpec represents the health check of an object store bucket

`.spec.metadataPool.statusCheck.mirror`

Description

HealthCheckSpec represents the health check of an object store bucket

Type

`object`

Property	Type	Description
<code>disabled</code>	<code>boolean</code>	
<code>interval</code>	<code>string</code>	Interval is the internal in second or minute for the health check to run like 60s for 60 seconds
<code>timeout</code>	<code>string</code>	

`.spec.protocols`

Description

The protocol specification

Type

`object`

Property	Type	Description
<code>enableAPIs</code>	<code>array</code>	Represents RGW 'rgw_enable_apis' config option. See: https://docs.ceph.com/en/reef/radosgw/config-ref/#confval-rgw_enable_apis ↗ If no value provided then all APIs will be enabled: s3, s3website, swift, swift_auth, admin, sts, iam, notifications If enabled APIs are set, all remaining APIs will be disabled. This option overrides S3.Enabled value.
<code>s3</code>	<code>object</code>	The spec for S3
<code>swift</code>	<code>object</code>	The spec for Swift

.spec.protocols.enableAPIs

Description

Represents RGW 'rgw_enable_apis' config option. See: https://docs.ceph.com/en/reef/radosgw/config-ref/#confval-rgw_enable_apis If no value provided then all APIs will be enabled: s3, s3website, swift, swift_auth, admin, sts, iam, notifications If enabled APIs are set, all remaining APIs will be disabled. This option overrides S3.Enabled value.

Type

`array`

.spec.protocols.enableAPIs[]

Type

`string`

.spec.protocols.s3

Description

The spec for S3

Type

object

Property	Type	Description
<code>authUseKeystone</code>	boolean	Whether to use Keystone for authentication. This option maps directly to the <code>rgw_s3_auth_use_keystone</code> option. Enabling it allows generating S3 credentials via an OpenStack API call, see the docs. If not given, the defaults of the corresponding RGW option apply.
<code>enabled</code>	boolean	Deprecated: use <code>protocol.enableAPIs</code> instead. Whether to enable S3. This defaults to true (even if <code>protocols.s3</code> is not present in the CRD). This maintains backwards compatibility – by default S3 is enabled.

.spec.protocols.swift

Description

The spec for Swift

Type

object

Property	Type	Description
<code>accountInUrl</code>	boolean	Whether or not the Swift account name should be included in the Swift API URL. If set to false (the default), then the Swift API will listen on a URL formed like http://host:port/

Property	Type	Description
		<rgw_swift_url_prefix>/v1. If set to true, the Swift API URL will be http://host:port/ ↗ <rgw_swift_url_prefix>/v1/AUTH_<account_name>. You must set this option to true (and update the Keystone service catalog) if you want radosgw to support publicly-readable containers and temporary URLs.
urlPrefix	string	The URL prefix for the Swift API, to distinguish it from the S3 API endpoint. The default is swift, which makes the Swift API available at the URL http://host:port/swift/v1 ↗ (or http://host:port/swift/v1/AUTH_%(tenant_id)s ↗ if rgw swift account in url is enabled).
versioningEnabled	boolean	Enables the Object Versioning of OpenStack Object Storage API. This allows clients to put the X-Versions-Location attribute on containers that should be versioned.

.spec.security

Description

Security represents security settings

Type

object

Property	Type	Description
<code>keyRotation</code>	<code>object</code>	KeyRotation defines options for Key Rotation.
<code>kms</code>	<code>object</code>	KeyManagementService is the main Key Management option
<code>s3</code>	<code>object</code>	The settings for supporting AWS-SSE:S3 with RGW

.spec.security.keyRotation

Description

KeyRotation defines options for Key Rotation.

Type

`object`

Property	Type	Description
<code>enabled</code>	<code>boolean</code>	Enabled represents whether the key rotation is enabled.
<code>schedule</code>	<code>string</code>	Schedule represents the cron schedule for key rotation.

.spec.security.kms

Description

KeyManagementService is the main Key Management option

Type

`object`

Property	Type	Description
<code>connectionDetails</code>	<code>object</code>	ConnectionDetails contains the KMS connection details (address, port etc)
<code>tokenSecretName</code>	<code>string</code>	TokenSecretName is the kubernetes secret containing the KMS token

.spec.security.kms.connectionDetails

Description

ConnectionDetails contains the KMS connection details (address, port etc)

Type

`object`

.spec.security.s3

Description

The settings for supporting AWS-SSE:S3 with RGW

Type

`object`

Property	Type	Description
<code>connectionDetails</code>	<code>object</code>	ConnectionDetails contains the KMS connection details (address, port etc)

Property	Type	Description
<code>tokenSecretName</code>	<code>string</code>	TokenSecretName is the kubernetes secret containing the KMS token

`.spec.security.s3.connectionDetails`

Description

ConnectionDetails contains the KMS connection details (address, port etc)

Type

`object`

`.spec.sharedPools`

Description

The pool information when configuring RADOS namespaces in existing pools.

Type

`object`

Property	Type	Description
<code>dataPoolName</code>	<code>string</code>	The data pool used for creating RADO object store
<code>metadataPoolName</code>	<code>string</code>	The metadata pool used for creating R the object store
<code>poolPlacements</code>	<code>array</code>	PoolPlacements control which Pools a particular RGW bucket. Once PoolPlac

Property	Type	Description
		RGW client will be able to associate pool with ObjectStore bucket by providing "" during s3 bucket creation or "X-Storage-Policy" header during swift container creation. See: https://docs.ceph.com/en/latest/radosgw/placement/#placement-targets PoolPlacement with name: "default" will be used as a default pool if no option is provided during bucket creation. If default placement is not provided, spec.sharedPools.dataPoolName and spec.sharedPools.MetadataPoolName will be used as default pools. If spec.sharedPools are also empty, then RGW pools (spec.dataPool and spec.metadataPool) will be used as defaults.
<code>preserveRadosNamespaceDataOnDelete</code>	<code>boolean</code>	Whether the RADOS namespaces should be preserved on deletion of the object store

.spec.sharedPools.poolPlacements

Description

PoolPlacements control which Pools are associated with a particular RGW bucket. Once PoolPlacements are defined, RGW client will be able to associate pool with ObjectStore bucket by providing "<LocationConstraint>" during s3 bucket creation or "X-Storage-Policy" header during swift container creation. See:

<https://docs.ceph.com/en/latest/radosgw/placement/#placement-targets> PoolPlacement with name: "default" will be used as a default pool if no option is provided during bucket creation. If default placement is not provided, spec.sharedPools.dataPoolName and spec.sharedPools.MetadataPoolName will be used as default pools. If spec.sharedPools are also empty, then RGW pools (spec.dataPool and spec.metadataPool) will be used as defaults.

Type

`array`

.spec.sharedPools.poolPlacements[]

Type

`object`

Required

`dataPoolName``metadataPoolName``name`

Property	Type	Description
<code>dataNonECPoolName</code>	<code>string</code>	The data pool used to store ObjectStore data that cannot use erasure coding (ex: multi-part uploads). If <code>dataPoolName</code> is not erasure coded, then there is no need for <code>dataNonECPoolName</code> .
<code>dataPoolName</code>	<code>string</code>	The data pool used to store ObjectStore objects data.
<code>default</code>	<code>boolean</code>	Sets given placement as default. Only one placement in the list can be marked as default. Default is false.
<code>metadataPoolName</code>	<code>string</code>	The metadata pool used to store ObjectStore bucket index.
<code>name</code>	<code>string</code>	Pool placement name. Name can be arbitrary. Placement with name "default" will be used as default.

Property	Type	Description
<code>storageClasses</code>	<code>array</code>	StorageClasses can be selected by user to override dataPoolName during object creation. Each placement has default STANDARD StorageClass pointing to dataPoolName. This list allows defining additional StorageClasses on top of default STANDARD storage class.

`.spec.sharedPools.poolPlacements[].storageClasses`

Description

StorageClasses can be selected by user to override dataPoolName during object creation. Each placement has default STANDARD StorageClass pointing to dataPoolName. This list allows defining additional StorageClasses on top of default STANDARD storage class.

Type

`array`

`.spec.sharedPools.poolPlacements[].storageClasses[]`

Type

`object`

Required

`dataPoolName``name`

Property	Type	Description
<code>dataPoolName</code>	<code>string</code>	DataPoolName is the data pool used to store ObjectStore objects data.

Property	Type	Description
<code>name</code>	<code>string</code>	Name is the StorageClass name. Ceph allows arbitrary name for StorageClasses, however most clients/libs insist on AWS names so it is recommended to use one of the valid x-amz-storage-class values for better compatibility: REDUCED_REDUNDANCY STANDARD_IA ONEZONE_IA INTELLIGENT_TIERING GLACIER DEEP_ARCHIVE OUTPOSTS GLACIER_IR SNOW EXPRESS_ONEZONE See AWS docs: https://aws.amazon.com/de/s3/storage-classes/ ↗

.spec.zone

Description

The multisite info

Type

`object`

Required

`name`

Property	Type	Description
<code>name</code>	<code>string</code>	CephObjectStoreZone name this CephObjectStore is part of

.status

Description

ObjectStoreStatus represents the status of a Ceph Object Store resource

Type

`object`

Property	Type	Description
<code>cephx</code>	<code>object</code>	
<code>conditions</code>	<code>array</code>	
<code>endpoints</code>	<code>object</code>	
<code>info</code>	<code>object</code>	
<code>message</code>	<code>string</code>	
<code>observedGeneration</code>	<code>integer</code>	ObservedGeneration is the latest generation observed by the controller.
<code>phase</code>	<code>string</code>	ConditionType represent a resource's status
<code>replicas</code>	<code>integer</code>	
<code>selector</code>	<code>string</code>	

`.status.cephx`

Type

`object`

Property	Type	Description
<code>daemon</code>	<code>object</code>	Daemon shows the CephX key status for local Ceph daemons associated with this resources.

.status.cephx.daemon

Description

Daemon shows the CephX key status for local Ceph daemons associated with this resources.

Type

object

Property	Type	Description
keyCephVersion	string	KeyCephVersion reports the Ceph version that created the current generation's keys. This is same string format as reported by <code>CephCluster.status.version.version</code> to allow them to be compared. E.g., <code>20.2.0-0</code> . For all newly-created resources, this field set to the version of Ceph that created the key. The special value "Uninitialized" indicates that keys are being created for the first time. An empty string indicates that the version is unknown, as expected in brownfield deployments.
keyGeneration	integer	KeyGeneration represents the CephX key generation for the last successful reconcile. For all newly-created resources, this field is set to <code>1</code> . When keys are rotated due to any rotation policy, the generation is incremented or updated to the configured policy generation. Generation <code>0</code> indicates that keys existed prior to the implementation of key tracking.

.status.conditions

Type

`array`

`.status.conditions[]`

Description

Condition represents a status condition on any Rook-Ceph Custom Resource.

Type

`object`

Property	Type	Description
<code>lastHeartbeatTime</code>	<code>string</code>	
<code>lastTransitionTime</code>	<code>string</code>	
<code>message</code>	<code>string</code>	
<code>reason</code>	<code>string</code>	ConditionReason is a reason for a condition
<code>status</code>	<code>string</code>	
<code>type</code>	<code>string</code>	ConditionType represent a resource's status

`.status.endpoints`

Type

`object`

Property	Type	Description
<code>insecure</code>	<code>array</code>	
<code>secure</code>	<code>array</code>	

.status.endpoints.insecure

Type

array

.status.endpoints.insecure[]

Type

string

.status.endpoints.secure

Type

array

.status.endpoints.secure[]

Type

string

.status.info

Type

object

API Endpoints

The following API endpoints are available:

- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstores`
 - **DELETE** : delete collection of CephObjectStore
 - **GET** : list objects of kind CephObjectStore

- **POST** : create a new CephObjectStore
- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstores/{name}`
- **DELETE** : delete the specified CephObjectStore
- **GET** : read the specified CephObjectStore
- **PATCH** : partially update the specified CephObjectStore
- **PUT** : replace the specified CephObjectStore
- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstores/{name}/status`
- **GET** : read status of the specified CephObjectStore
- **PATCH** : partially update status of the specified CephObjectStore
- **PUT** : replace status of the specified CephObjectStore

`/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstores`

HTTP method

DELETE

Description

delete collection of CephObjectStore

HTTP responses

HTTP code	Response body
200 - OK	Status schema
401 - Unauthorized	Empty

HTTP method

GET

Description

list objects of kind CephObjectStore

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStoreList</code> schema
401 - Unauthorized	Empty

HTTP method

`POST`

Description

create a new CephObjectStore

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephObjectStore</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStore</code> schema
201 - Created	<code>CephObjectStore</code> schema
202 - Accepted	<code>CephObjectStore</code> schema
401 - Unauthorized	Empty

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstores/{name}

HTTP method

`DELETE`

Description

delete the specified CephObjectStore

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
202 - Accepted	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

read the specified CephObjectStore

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStore</code> schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update the specified CephObjectStore

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore:

Parameter	Type	Description
		This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStore</code> schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified CephObjectStore

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Parameter	Type	Description
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephObjectStore</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStore</code> schema
201 - Created	<code>CephObjectStore</code> schema
401 - Unauthorized	Empty

`/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstores/{name}/status`

HTTP method

GET

Description

read status of the specified CephObjectStore

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStore</code> schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update status of the specified CephObjectStore

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default

Parameter	Type	Description
		in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStore</code> schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace status of the specified CephObjectStore

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request

Parameter	Type	Description
		will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
body	CephObjectStore schema	application/json formatted

HTTP responses

HTTP code	Response body
200 - OK	CephObjectStore schema
201 - Created	CephObjectStore schema
401 - Unauthorized	Empty

CephObjectStoreUser

[cephobjectstoreusers.ceph.rook.io/v1]

Description

CephObjectStoreUser represents a Ceph Object Store Gateway User

Type

object

Required

metadata

spec

Specification

Property	Type	Description
apiVersion	string	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
kind	string	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be

Property	Type	Description
		updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
metadata	ObjectMeta	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
spec	object	ObjectStoreUserSpec represent the spec of an Objectstoreuser
status	object	ObjectStoreUserStatus represents the status Ceph Object Store Gateway User

.spec

Description

ObjectStoreUserSpec represent the spec of an Objectstoreuser

Type

object

Property	Type	Description
capabilities	object	Additional admin-level capabilities for the Ceph object store user

Property	Type	Description
<code>clusterNamespace</code>	<code>string</code>	The namespace where the parent CephCluster and CephObjectStore are found
<code>displayName</code>	<code>string</code>	The display name for the ceph user.
<code>keys</code>	<code>array</code>	Allows specifying credentials for the user. If not provided, the operator will generate them.
<code>quotas</code>	<code>object</code>	ObjectUserQuotaSpec can be used to set quotas for the object store user to limit their usage. See the Ceph docs [↗] for more
<code>store</code>	<code>string</code>	The store the user will be created in

.spec.capabilities

Description

Additional admin-level capabilities for the Ceph object store user

Type

`object`

Property	Type	Description
<code>amz-cache</code>	<code>string</code>	Add capabilities for user to send request to RGW Cache API header. Documented in

Property	Type	Description
		https://docs.ceph.com/en/latest/radosgw/rgw-cache/#cache-api ↗
<code>biolog</code>	<code>string</code>	Add capabilities for user to change bucket index logging. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
<code>bucket</code>	<code>string</code>	Admin capabilities to read/write Ceph object store buckets. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
<code>buckets</code>	<code>string</code>	Admin capabilities to read/write Ceph object store buckets. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
<code>datalog</code>	<code>string</code>	Add capabilities for user to change data logging. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
<code>info</code>	<code>string</code>	Admin capabilities to read/write information about the user. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗

Property	Type	Description
mdlog	string	Add capabilities for user to change metadata logging. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
metadata	ObjectMeta	Admin capabilities to read/write Ceph object store metadata. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
oidc-provider	string	Add capabilities for user to change oidc provider. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
ratelimit	string	Add capabilities for user to set rate limiter for user and bucket. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
roles	string	Admin capabilities to read/write roles for user. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
usage	string	Admin capabilities to read/write Ceph object store usage. Documented in

Property	Type	Description
		https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
user	string	Admin capabilities to read/write Ceph object store users. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
user - policy	string	Add capabilities for user to change user policies. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
users	string	Admin capabilities to read/write Ceph object store users. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗
zone	string	Admin capabilities to read/write Ceph object store zones. Documented in https://docs.ceph.com/en/latest/radosgw/admin/?#add-remove-admin-capabilities ↗

.spec.keys

Description

Allows specifying credentials for the user. If not provided, the operator will generate them.

Type

array

.spec.keys[]

Description

ObjectUserKey defines a set of rgw user access credentials to be retrieved from secret resources.

Type

object

Property	Type	Description
<code>accessKeyRef</code>	object	Secret key selector for the access_key (commonly referred to as AWS_ACCESS_KEY_ID).
<code>secretKeyRef</code>	object	Secret key selector for the secret_key (commonly referred to as AWS_SECRET_ACCESS_KEY).

.spec.keys[].accessKeyRef

Description

Secret key selector for the access_key (commonly referred to as AWS_ACCESS_KEY_ID).

Type

object

Required

key

Property	Type	Description
<code>key</code>	<code>string</code>	The key of the secret to select from. Must be a valid secret key.
<code>name</code>	<code>string</code>	Name of the referent. This field is effectively required, but due to backwards compatibility is allowed to be empty. Instances of this type with an empty value here are almost certainly wrong. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names ↗
<code>optional</code>	<code>boolean</code>	Specify whether the Secret or its key must be defined

`.spec.keys[].secretKeyRef`

Description

Secret key selector for the secret_key (commonly referred to as AWS_SECRET_ACCESS_KEY).

Type

`object`

Required

`key`

Property	Type	Description
<code>key</code>	<code>string</code>	The key of the secret to select from. Must be a valid secret key.

Property	Type	Description
<code>name</code>	<code>string</code>	Name of the referent. This field is effectively required, but due to backwards compatibility is allowed to be empty. Instances of this type with an empty value here are almost certainly wrong. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names ↗
<code>optional</code>	<code>boolean</code>	Specify whether the Secret or its key must be defined

.spec.quotas

Description

ObjectUserQuotaSpec can be used to set quotas for the object store user to limit their usage. See the [Ceph docs](https://docs.ceph.com/en/latest/radosgw/admin/?#quota-management) for more

Type

`object`

Property	Type	Description
<code>maxBuckets</code>	<code>integer</code>	Maximum bucket limit for the ceph user
<code>maxObjects</code>	<code>integer</code>	Maximum number of objects across all the user's buckets
<code>maxSize</code>		Maximum size limit of all objects across all the user's buckets See

Property	Type	Description
		https://pkg.go.dev/k8s.io/apimachinery/pkg/api/resource#Quantity ↗ for more info.

.status

Description

ObjectStoreUserStatus represents the status Ceph Object Store Gateway User

Type

object

Property	Type	Description
info	object	
keys	array	
observedGeneration	integer	ObservedGeneration is the latest generation observed by the controller.
phase	string	

.status.info

Type

object

.status.keys

Type

array

.status.keys[]

Type

object

Property	Type	Description
name	string	name is unique within a namespace to reference a secret resource.
namespace	string	namespace defines the space within which the secret name must be unique.
resourceVersion	string	
uid	string	UID is a type that holds unique ID values, including UUIDs. Because we don't ONLY use UUIDs, this is an alias to string. Being a type captures intent and helps make sure that UUIDs and names do not get conflated.

API Endpoints

The following API endpoints are available:

- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstoreusers`
 - `DELETE` : delete collection of CephObjectStoreUser
 - `GET` : list objects of kind CephObjectStoreUser
 - `POST` : create a new CephObjectStoreUser
- `/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstoreusers/{name}`

- **DELETE** : delete the specified CephObjectStoreUser
 - **GET** : read the specified CephObjectStoreUser
 - **PATCH** : partially update the specified CephObjectStoreUser
 - **PUT** : replace the specified CephObjectStoreUser
- **/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstoreusers/{name}/status**
 - **GET** : read status of the specified CephObjectStoreUser
 - **PATCH** : partially update status of the specified CephObjectStoreUser
 - **PUT** : replace status of the specified CephObjectStoreUser

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstoreusers

HTTP method

DELETE

Description

delete collection of CephObjectStoreUser

HTTP responses

HTTP code	Response body
200 - OK	Status schema
401 - Unauthorized	Empty

HTTP method

GET

Description

list objects of kind CephObjectStoreUser

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStoreUserList</code> schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a new CephObjectStoreUser

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephObjectStoreUser</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStoreUser</code> schema
201 - Created	<code>CephObjectStoreUser</code> schema
202 - Accepted	<code>CephObjectStoreUser</code> schema
401 - Unauthorized	Empty

/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstoreusers/{name}

HTTP method

DELETE

Description

delete the specified CephObjectStoreUser

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	<code>Status</code> schema
202 - Accepted	<code>Status</code> schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

read the specified CephObjectStoreUser

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStoreUser</code> schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update the specified CephObjectStoreUser

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore:

Parameter	Type	Description
		This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStoreUser</code> schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified CephObjectStoreUser

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Parameter	Type	Description
<code>fieldValidation</code>	<code>string</code>	<code>fieldValidation</code> instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+. - Strict: This will fail the request with a <code>BadRequest</code> error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
<code>body</code>	<code>CephObjectStoreUser</code> schema	<code>application/json</code> formatted

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStoreUser</code> schema
201 - Created	<code>CephObjectStoreUser</code> schema
401 - Unauthorized	Empty

`/apis/ceph.rook.io/v1/namespaces/{namespace}/cephobjectstoreusers/{name}/status`

HTTP method

GET

Description

read status of the specified CephObjectStoreUser

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStoreUser</code> schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update status of the specified CephObjectStoreUser

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default

Parameter	Type	Description
		in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP responses

HTTP code	Response body
200 - OK	<code>CephObjectStoreUser</code> schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace status of the specified CephObjectStoreUser

Query parameters

Parameter	Type	Description
<code>dryRun</code>	<code>string</code>	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
<code>fieldValidation</code>	<code>string</code>	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request

Parameter	Type	Description
		will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
body	CephObjectStoreUser schema	application/json formatted

HTTP responses

HTTP code	Response body
200 - OK	CephObjectStoreUser schema
201 - Created	CephObjectStoreUser schema
401 - Unauthorized	Empty