NetApp

Manage storage aggregates

REST API reference

NetApp
September 12, 2025

Table of Contents

Manage storage aggregates

Storage aggregates UUID endpoint overview

Updating storage aggregates

The PATCH operation is used to modify properties of the aggregate. There are several properties that can be modified on an aggregate. Only one property can be modified for each PATCH request. PATCH operations on the aggregate's disk count will be blocked while one or more nodes in the cluster are simulating or implementing automatic aggregate creation.

The following is a list of properties that can be modified using the PATCH operation including a brief description for each:

- name - This property can be changed to rename the aggregate.

- node.name and node.uuid - Either property can be updated in order to relocate the aggregate to a different node in the cluster.

- state - This property can be changed to 'online' or 'offline'. Setting an aggregate 'offline' would automatically offline all the volumes currently hosted on the aggregate.

- block_storage.mirror.enabled - This property can be changed from 'false' to 'true' in order to mirror the aggregate, if the system is capable of doing so.

- block_storage.primary.disk_count - This property can be updated to increase the number of disks in an aggregate.

- block_storage.primary.raid_size - This property can be updated to set the desired RAID size.

- block_storage.primary.raid_type - This property can be updated to set the desired RAID type.

- cloud_storage.tiering_fullness_threshold - This property can be updated to set the desired tiering fullness threshold if using FabricPool.

- cloud_storage.migrate_threshold - This property can be updated to set the desired migrate threshold if using FabricPool.

- data_encryption.software_encryption_enabled - This property enables or disables NAE on the aggregate.

- block_storage.hybrid_cache.storage_pools.allocation_units_count - This property can be updated to add a storage pool to the aggregate specifying the number of allocation units.

- block_storage.hybrid_cache.storage_pools.name - This property can be updated to add a storage pool to the aggregate specifying the storage pool name. block_storage.hybrid_cache.storage_pools.uuid or this field must be specified with block_storage.hybrid_cache.storage_pools.allocation_units_count.

- block_storage.hybrid_cache.storage_pools.uuid - This property can be updated to add a storage pool to the aggregate specifying the storage pool uuid. block_storage.hybrid_cache.storage_pools.name or this field must be specified with block_storage.hybrid_cache.storage_pools.allocation_units_count.

- block_storage.hybrid_cache.raid_size - This property can be updated to set the desired RAID size. This property can also be specified on the first time addition of a storage pool to the aggregate.

- block_storage.hybrid_cache.raid_type - This property can be updated to set the desired RAID type of a physical SSD Flash Pool. This property can also be specified on the first time addition of a storage pool to the aggregate. When specifying a raidtype of raid4, the node is required to have spare SSDs for the storage pool as well.

- block_storage.hybrid_cache.disk_count - This property can be specified on the first time addition of physical SSD cache to the aggregate. It can also be updated to increase the number of disks in the

physical SSD cache of a hybrid aggregate.

Aggregate expansion

The PATCH operation also supports automatically expanding an aggregate based on the spare disks which are present within the system. Running PATCH with the query "auto_provision_policy" set to "expand" starts the recommended expansion job. In order to see the expected change in capacity before starting the job, call GET on an aggregate instance with the query "auto_provision_policy" set to "expand".

Manual simulated aggregate expansion

The PATCH operation also supports simulated manual expansion of an aggregate. Running PATCH with the query "simulate" set to "true" and "block_storage.primary.disk_count" set to the final disk count will start running the prechecks associated with expanding the aggregate to the proposed size. The response body will include information on how many disks the aggregate can be expanded to, any associated warnings, along with the proposed final size of the aggregate.

Deleting storage aggregates

If volumes exist on an aggregate, they must be deleted or moved before the aggregate can be deleted. See the /storage/volumes API for details on moving or deleting volumes.

Adding a storage pool to an aggregate

A storage pool can be added to an aggregate by patching the field "block_storage.hybrid_cache.storage_pools.allocation_units_count" while also specifying the specific storage pool using the "block_storage.hybrid_cache.storage_pools.name" or "block_storage.hybrid_cache.storage_pools.uuid". Subsequent patches to the aggregate can be completed to increase allocation unit counts or adding additional storage pools. On the first time addition of a storage pool to the aggregate, the raidtype can be optionally specified using the "block_storage.hybrid_cache.raid_type" field.

Adding physical SSD cache capacity to an aggregate

The PATCH operation supports addition of a new physical SSD cache to an aggregate. It also supports expansion of existing physical SSD cache in the hybrid aggregate. Running PATCH with "block_storage.hybrid_cache.disk_count" set to the final disk count will expand the physical SSD cache of the hybrid aggregate to the proposed size. The RAID type can be optionally specified using the "block_storage.hybrid_cache.raid_type" field. The RAID size can be optionally specified using the "block_storage.hybrid_cache.raid_size" field. These operations can also be simulated by setting the query "simulate" to "true".

Examples

Retrieving a specific aggregate from the cluster

The following example shows the response of the requested aggregate. If there is no aggregate with the requested UUID, an error is returned.

```
# The API:
/api/storage/aggregates/{uuid}
```

```
# The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates/870dd9f2-bdfa-4167-
b692-57d1cec874d4" -H "accept: application/json"

# The response:
{
"uuid": "19425837-f2fa-4a9f-8f01-712f626c983c",
"name": "test1",
"node": {
  "uuid": "caf95bec-f801-11e8-8af9-005056bbe5c1",
  "name": "node-1",
},
"home_node": {
  "uuid": "caf95bec-f801-11e8-8af9-005056bbe5c1",
  "name": "node-1",
},
"space": {
  "block_storage": {
    "size": 235003904,
    "available": 191942656,
    "used": 43061248,
    "full_threshold_percent": 98,
    "physical_used": 5271552,
    "physical_used_percent": 1,
    "volume_footprints_percent": 14,
    "aggregate_metadata": 2655,
    "aggregate_metadata_percent": 8,
    "used_including_snapshot_reserve": 674685,
    "used_including_snapshot_reserve_percent": 35,
    "data_compacted_count": 666666,
    "data_compaction_space_saved": 654566,
    "data_compaction_space_saved_percent": 47,
    "volume_deduplication_shared_count": 567543,
    "volume_deduplication_space_saved": 23765,
    "volume_deduplication_space_saved_percent": 32,
    "used_percent": 50,
  },
  "snapshot": {
    "used_percent": 45,
    "available": 2000,
    "total": 5000,
    "used": 3000,
    "reserve_percent": 20
  },
  "cloud_storage": {
    "used": 0
```

```
    },
    "efficiency": {
      "savings": 1408029,
      "ratio": 6.908119720880661,
      "logical_used": 1646350,
      "cross_volume_background_dedupe": true,
      "cross_volume_inline_dedupe": false,
      "cross_volume_dedupe_savings": true,
      "auto_adaptive_compression_savings": false
    },
    "efficiency_without_snapshots": {
      "savings": 0,
      "ratio": 1,
      "logical_used": 737280
    },
    "efficiency_without_snapshots_flexclones": {
      "savings": 5000,
      "ratio": 2,
      "logical_used": 10000
    }
  },
  "snapshot": {
    "files_total": 10,
    "files_used": 3,
    "max_files_available": 5,
    "max_files_used": 50
  },
  "state": "online",
  "snaplock_type": "non_snaplock",
  "create_time": "2018-12-04T15:40:38-05:00",
  "data_encryption": {
    "software_encryption_enabled": false,
    "drive_protection_enabled": false
  },
  "block_storage": {
    "uses_partitions": false,
    "storage_type": "vmdisk",
    "primary": {
      "disk_count": 6,
      "disk_class": "solid_state",
      "raid_type": "raid_dp",
      "raid_size": 24,
      "checksum_style": "block",
      "disk_type": "ssd"
    },
    "hybrid_cache": {
```

```
      "enabled": false
    },
    "mirror": {
      "enabled": false,
      "state": "unmirrored"
    },
    "plexes": [
      {
        "name": "plex0",
      }
    ]
  },
  "cloud_storage": {
    "attach_eligible": false
  },
  "inode_attributes": {
    "files_total": 31136,
    "files_used": 97,
    "max_files_available": 31136,
    "max_files_possible": 2844525,
    "max_files_used": 97,
    "used_percent": 5
  },
  "volume_count": 0,
  }
```

Retrieving statistics and metric for an aggregate

In this example, the API returns the "statistics" and "metric" properties for the aggregate requested.

```
#The API:
/api/storage/aggregates/{uuid}?fields=statistics,metric

#The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates/538bf337-1b2c-11e8-
bad0-005056b48388?fields=statistics,metric" -H "accept: application/json"

#The response:
{
"uuid": "538bf337-1b2c-11e8-bad0-005056b48388",
"name": "aggr4",
"metric": {
    "timestamp": "2019-07-08T22:16:45Z",
    "duration": "PT15S",
    "status": "ok",
    "throughput": {
```

```
        "read": 7099,
        "write": 840226,
        "other": 193293789,
        "total": 194141115
        },
      "latency": {
        "read": 149,
        "write": 230,
        "other": 123,
        "total": 124
      },
      "iops": {
        "read": 1,
        "write": 17,
        "other": 11663,
        "total": 11682
      },
    },
  "statistics": {
      "timestamp": "2019-07-08T22:17:09Z",
      "status": "ok",
      "throughput_raw": {
        "read": 3106045952,
        "write": 63771742208,
        "other": 146185560064,
        "total": 213063348224
      },
      "latency_raw": {
        "read": 54072313,
        "write": 313354426,
        "other": 477201985,
        "total": 844628724
      },
      "iops_raw": {
        "read": 328267,
        "write": 1137230,
        "other": 1586535,
        "total": 3052032
      }
    },
  }
```

For more information and examples on viewing historical performance metrics for any given aggregate, see
DOC /storage/aggregates/{uuid}/metrics

Simulating aggregate expansion

The following example shows the response for a simulated data aggregate expansion based on the values of the 'block_storage.primary.disk_count' attribute passed in. The query does not modify the existing aggregate but returns how the aggregate will look after the expansion along with any associated warnings. Simulated data aggregate expansion will be blocked while one or more nodes in the cluster are simulating or implementing automatic aggregate creation. This will be reflected in the following attributes:

- space.block_storage.size - Total usable space in bytes, not including WAFL reserve and aggregate Snapshot copy reserve.

- block_storage.primary.disk_count - Number of disks that could be used to create the aggregate.

```
# The API:
/api/storage/aggregates/{uuid}?simulate=true

# The call:
curl -X PATCH "https://<mgmt-ip>/api/storage/aggregates/cae60cfe-deae-
42bd-babb-ef437d118314?simulate=true" -H "accept: application/json" -d
"{\"block_storage\": {\"primary\": {\"disk_count\": 13}}}"

# The response:
{
"warnings": [
  {
    "name": "node_2_SSD_1",
    "warning": {
      "message": "One or more disks will not be added. 10 disks specified,
9 disks will be added.",
      "code": 787170,
      "arguments": [
        "10",
        "9"
      ]
    }
  }
],
"num_records": 1,
"records": [
  {
    "uuid": "cae60cfe-deae-42bd-babb-ef437d118314",
    "name": "node_2_SSD_1",
    "node": {
      "uuid": "4046dda8-f802-11e8-8f6d-005056bb2030",
      "name": "node-2"
    },
    "space": {
      "block_storage": {
```

```
            "size": 1116180480
        }
    },
    "block_storage": {
        "primary": {
            "disk_count": 12,
            "disk_class": "solid_state",
            "raid_type": "raid_dp",
            "disk_type": "ssd",
            "raid_size": 12,
            "simulated_raid_groups": [
                {
                    "name": "test/plex0/rg0",
                    "existing_parity_disk_count": 2,
                    "added_parity_disk_count": 0,
                    "existing_data_disk_count": 1,
                    "added_data_disk_count": 9,
                    "usable_size": 12309487,
                    "is_partition": false
                }
            ]
        },
        "hybrid_cache": {
            "enabled": false
        },
        "mirror": {
            "enabled": false
        }
    },
}
]
}
```

Manual aggregate expansion with disk size query

The following example shows the response for aggregate expansion based on the values of the 'block_storage.hybrid_cache.disk_count' attribute based on the disk size passed in.

```
# The API:
/api/storage/aggregate/{uuid}?disk_size={disk_size}

# The call:
curl -X PATCH  "https://<mgmt-ip>/api/storage/aggregates/cae60cfe-deae-
42bd-babb-ef437d118314?disk_size=1902379008" -H "accept: application/json"
-d "{\"block_storage\": {\"hybrid_cache\": {\"disk_count\": 4}}}"

# The response:
{
"job": {
  "uuid": "c103d15e-730b-11e8-a57f-005056b465d6",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/c103d15e-730b-11e8-a57f-005056b465d6"
    }
  }
}
}
```

Simulating a manual aggregate expansion with disk size query

The following example shows the response for a manual aggregate expansion based on the values of the 'block_storage.hybrid_cache.disk_count' attribute based on the disk size passed in. The query internally maps out the appropriate expansion as well as warnings that may be associated for the hybrid enabled aggregate.

```
# The API:
/api/storage/aggregate/{uuid}?simulate=true&disk_size=1902379008

# The call:
curl -X PATCH  "https://<mgmt-ip>/api/storage/aggregates/cae60cfe-deae-
42bd-babb-ef437d118314?simulate=true&disk_size=1902379008" -H "accept:
application/json" -d "{\"block_storage\": {\"hybrid_cache\":
{\"disk_count\": 4}}}"

# The response:
{
"num_records": 1,
"records": [
  {
    "uuid": "cae60cfe-deae-42bd-babb-ef437d118314",
    "name": "ag1",
    "node": {
      "uuid": "4046dda8-f802-11e8-8f6d-005056bb2030",
      "name": "node-2",
```

```
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/4046dda8-f802-11e8-8f6d-
005056bb2030"
        }
      }
    },
    "block_storage": {
      "primary": {
        "disk_count": 4,
        "disk_class": "virtual",
        "raid_type": "raid_dp",
        "disk_type": "vm_disk",
      },
      "hybrid_cache": {
        "disk_type": "ssd",
        "enabled": true,
        "disk_count": 4,
        "raid_type": "raid_dp",
        "size": 3761766400,
        "simulated_raid_groups": [
          {
            "name": "test/plex0/rg0",
            "existing_parity_disk_count": 2,
            "existing_data_disk_count": 1,
            "added_parity_disk_count": 0,
            "added_data_disk_count": 1,
            "usable_size": 1880883200,
            "is_partition": false
          },
        ]
      },
      "mirror": {
        "enabled": false
      },
      "_links": {
        "self": {
          "href": "/api/storage/aggregates/cae60cfe-deae-42bd-babb-
ef437d118314"
        }
      }
    }
  }
]
}
```

Simulating a manual aggregate expansion with raid group query

The following example shows the response for a manual aggregate expansion based on the values of the 'block_storage.primary.disk_count' attribute passed in. The query internally maps out the appropriate expansion as well as warnings that may be associated and lays out the new raidgroups in a more detailed view. An additional query can be passed in to specify raidgroup addition by new raidgroup, all raidgroups or a specific raidgroup.

```
# The API:
/api/storage/aggregate/{uuid}?simulate=true&raid_group=[new|all|
rgX]

# The call:
curl -X PATCH  "https://<mgmt-ip>/api/storage/aggregates/cae60cfe-deae-
42bd-babb-ef437d118314?simulate=true&raid_group=new" -H "accept:
application/json" -d "{\"block_storage\": {\"primary\": {\"disk_count\":
24}}}"

# The response:
{
"warnings": [
  {
    "name": "test",
    "warning": {
      "code": 11,
      "message": "Number of unassigned disks attached to node \"node-2\":
6.",
      "arguments": [
        "6",
        "node-2"
      ]
    }
  }
],
"num_records": 1,
"records": [
  {
    "uuid": "cae60cfe-deae-42bd-babb-ef437d118314",
    "name": "test",
    "node": {
      "uuid": "4046dda8-f802-11e8-8f6d-005056bb2030",
      "name": "node-2"
    },
    "space": {
      "block_storage": {
        "size": 33292025856
      }
```

```
      },
      "block_storage": {
        "primary": {
          "disk_count": 24,
          "disk_class": "solid_state",
          "raid_type": "raid_dp",
          "disk_type": "ssd",
          "raid_size": 24,
          "simulated_raid_groups": [
              {
                "name": "test/plex0/rg0",
                "existing_parity_disk_count": 0,
                "added_parity_disk_count": 2,
                "existing_data_disk_count": 0,
                "added_data_disk_count": 10,
                "usable_size": 12309487,
                "is_partition": false
              },
              {
                "name": "test/plex1/rg1",
                "existing_parity_disk_count": 0,
                "added_parity_disk_count": 2,
                "existing_data_disk_count": 0,
                "added_data_disk_count": 10,
                "usable_size": 12309487,
                "is_partition": false
              }
          ]
        },
        "hybrid_cache": {
          "enabled": false
        },
        "mirror": {
          "enabled": false
        }
      }
    }
  ]
  }
```

Retrieving the usable spare information for the cluster

The following example shows the response from retrieving usable spare information for the expansion of this particular aggregate. The output is restricted to only spares that are compatible with this aggregate.

```
# The API:
/api/storage/aggregates?show_spares=true&uuid={uuid}

# The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates?uuid=cae60cfe-deae-
42bd-babb-ef437d118314&show_spares=true" -H "accept: application/json"

# The response:
{
"records": [],
"num_reecords": 0,
"spares": [
  {
    "node": {
      "uuid": "0cdd84fa-b99c-11eb-b0ed-005056bb4fc2",
      "name": "node-2"
    },
    "disk_class": "solid_state",
    "disk_type": "ssd",
    "size": 3720609792,
    "checksum_style": "block",
    "syncmirror_pool": "pool0",
    "usable": 12,
    "layout_requirements": [
      {
        "raid_type": "raid_dp",
        "default": true,
        "aggregate_min_disks": 3,
        "raid_group": {
          "min": 3,
          "max": 28,
          "default": 24
        }
      }
    ]
  }
]
}
```

Retrieving the SSD spare count for the cluster

The following example shows the response from retrieving SSD spare count information for the expansion of this particular aggregate's hybrid cache tier. The output is restricted to only spares that are compatible with this aggregate.

```
# The API:
/api/storage/aggregates?show_spares=true&uuid={uuid}&flash_pool_eligible=t
rue

# The response:
{
"records": [],
"num_records": 0,
"spares": [
  {
    "node": {
      "uuid": "c35c5975-cbcb-11ec-a3e1-005056bbdb46",
      "name": "node-2"
    },
    "disk_class": "solid_state",
    "disk_type": "ssd",
    "size": 1902379008,
    "checksum_style": "block",
    "syncmirror_pool": "pool0",
    "is_partition": false,
    "usable": 1,
    "layout_requirements": [
      {
        "raid_type": "raid4",
        "default": true,
        "aggregate_min_disks": 2,
        "raid_group": {
          "min": 2,
          "max": 14,
          "default": 8
        }
      }
    ]
  }
]
}
```

Retrieving a recommendation for an aggregate expansion

The following example shows the response with the recommended data aggregate expansion based on what
disks are present within the system. The query does not modify the existing aggregate but returns how the
aggregate will look after the expansion. The recommendation will be reflected in the attributes -
'space.block_storage.size' and 'block_storage.primary.disk_count'. Recommended data aggregate expansion
will be blocked while one or more nodes in the cluster are simulating or implementing automatic aggregate
creation.

```
# The API:
/api/storage/aggregates/{uuid}?auto_provision_policy=expand

# The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates/cae60cfe-deae-42bd-
babb-ef437d118314?auto_provision_policy=expand" -H "accept:
application/json"

# The response:
{
"uuid": "cae60cfe-deae-42bd-babb-ef437d118314",
"name": "node_2_SSD_1",
"node": {
  "uuid": "4046dda8-f802-11e8-8f6d-005056bb2030",
  "name": "node-2"
},
"space": {
  "block_storage": {
    "size": 1116180480
  }
},
"block_storage": {
  "primary": {
    "disk_count": 12,
    "disk_class": "solid_state",
    "raid_type": "raid_dp",
    "disk_type": "ssd",
    "raid_size": 24,
    "simulated_raid_groups": [
        {
          "name": "test/plex0/rg0",
          "parity_disk_count": 2,
          "data_disk_count": 10,
          "usable_size": 12309487,
          "is_partition": false
        }
      ]
  },
  "hybrid_cache": {
    "enabled": false
  },
  "mirror": {
    "enabled": false
  }
}
}
```

Updating an aggregate in the cluster

The following example shows the workflow of adding disks to the aggregate.

Step 1: Check the current disk count on the aggregate.

```
# The API:
/api/storage/aggregates

# The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates/19425837-f2fa-4a9f-
8f01-712f626c983c?fields=block_storage.primary.disk_count" -H "accept:
application/json"

# The response:
{
"uuid": "19425837-f2fa-4a9f-8f01-712f626c983c",
"name": "test1",
"block_storage": {
  "primary": {
    "disk_count": 6
  }
},
}
```

Step 2: Update the aggregate with the new disk count in 'block_storage.primary.disk_count'. The response to PATCH is a job unless the request is invalid.

```
# The API:
/api/storage/aggregates

# The call:
curl -X PATCH "https://<mgmt-ip>/api/storage/aggregates/19425837-f2fa-
4a9f-8f01-712f626c983c" -H "accept: application/hal+json" -d
"{\"block_storage\": {\"primary\": {\"disk_count\": 8}}}"

# The response:
{
"job": {
  "uuid": "c103d15e-730b-11e8-a57f-005056b465d6",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/c103d15e-730b-11e8-a57f-005056b465d6"
    }
  }
}
}
```

Step 3: Wait for the job to finish, then call GET to see the reflected change.

```
# The API:
/api/storage/aggregates

# The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates/19425837-f2fa-4a9f-
8f01-712f626c983c?fields=block_storage.primary.disk_count" -H "accept:
application/json"

# The response:
{
"uuid": "19425837-f2fa-4a9f-8f01-712f626c983c",
"name": "test1",
"block_storage": {
  "primary": {
    "disk_count": 8
  }
},
}
```

Adding a storage pool to an aggregate

The following example shows how to add cache capacity from an existing storage pool to an aggregate. Step
1: Update the aggregate with the new storage pool allocation unit in

'block_storage.hybrid_cache.storage_pools.allocation_units_count'. Additionally, specify 'block_storage.hybrid_cache.storage_pools.name' or 'block_storage.hybrid_cache.storage_pools.uuid' to the storage pool. On the first storage pool, 'block_storage.hybrid_cache.raid_type' can be specified for the raidtype of the hybrid cache. The response to PATCH is a job unless the request is invalid.

```
# The API:
/api/storage/aggregates

# The call:
curl -X PATCH "https://<mgmt-ip>/api/storage/aggregates/19425837-f2fa-
4a9f-8f01-712f626c983c" -H "accept: application/json" -d
"{\"block_storage\": {\"hybrid_cache\": {\"raid_type\": \"raid_dp\",
\"storage_pools\": [{ \"allocation_units_count\": 2, \"storage_pool\": {
\"name\": \"sp1\"}}]}}}"

# The response:
{
"job": {
  "uuid": "c103d15e-730b-11e8-a57f-005056b465d6",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/c103d15e-730b-11e8-a57f-005056b465d6"
    }
  }
}
}
```

Step 2: Wait for the job to finish, then call GET to see the reflected change.

```
# The API:
/api/storage/aggregates

# The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates/19425837-f2fa-4a9f-
8f01-712f626c983c?fields=block_storage.hybrid_cache" -H "accept:
application/json"

# The response:
{
"uuid": "19425837-f2fa-4a9f-8f01-712f626c983c",
"name": "test1",
"hybrid_cache": {
    "enabled": true,
    "disk_count": 3,
    "raid_size": 24,
    "raid_type": "raid_dp",
    "size": 880279552,
    "used": 73728,
    "storage_pools": [
        {
            "allocation_units_count": 2,
            "storage_pool": {
                "name": "sp1",
                "uuid": "eeef0b24-846b-11ec-8fcb-005056bb12c7"
            }
        }
    ]
}
}
```

Adding physical SSD cache capacity to an aggregate

The following example shows how to add physical SSD cache capacity to an aggregate. Step 1: Specify the number of disks to be added to cache in 'block_storage.hybrid_cache.disk_count'. 'block_storage.hybrid_cache.raid_type' can be specified for the RAID type of the hybrid cache. 'block_storage.hybrid_cache.raid_size' can be specified for the RAID size of the hybrid cache. The response to PATCH is a job unless the request is invalid.

```
# The API:
/api/storage/aggregates

# The call:
curl -X PATCH "https://<mgmt-ip>/api/storage/aggregates/caa8a9f1-0219-
4eaf-bcad-e29c05042fe1" -H "accept: application/json" -d
'{"block_storage.hybrid_cache.disk_count":3,"block_storage.hybrid_cache.ra
id_type":"raid4"}'

# The response:
{
"job": {
  "uuid": "c103d15e-730b-11e8-a57f-005056b465d6",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/c103d15e-730b-11e8-a57f-005056b465d6"
    }
  }
}
}
```

Step 2: Wait for the job to finish, then call GET to see the reflected change.

```
# The API:
/api/storage/aggregates

# The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates/caa8a9f1-0219-4eaf-
bcad-e29c05042fe1?fields=block_storage.hybrid_cache" -H "accept:
application/json"

# The response:
{
"uuid": "caa8a9f1-0219-4eaf-bcad-e29c05042fe1",
"name": "test1",
"hybrid_cache": {
    "enabled": true,
    "disk_count": 3,
    "raid_size": 24,
    "raid_type": "raid4",
    "size": 880279552,
    "used": 73728
}
}
```

Simulated addition of physical SSD cache capacity to an aggregate

The following example shows the response for a simulated addition of physical SSD cache capacity to an aggregate based on the values of the 'block_storage.hybrid_cache.disk_count', 'block_storage.hybrid_cache.raid_type' and 'block_storage.hybrid_cache.raid_size' attributes passed in. The query does not modify the existing aggregate but returns how the aggregate will look after the expansion along with any associated warnings. Simulated addition of physical SSD cache capacity to an aggregate will be blocked while one or more nodes in the cluster are simulating or implementing automatic aggregate creation. This will be reflected in the following attributes:

- block_storage.hybrid_cache.size - Total usable cache space in bytes, not including WAFL reserve and aggregate Snapshot copy reserve.

- block_storage.hybrid_cache.disk_count - Number of disks that can be added to the aggregate's cache tier.

```
# The API:
/api/storage/aggregates/{uuid}?simulate=true

# The call:
curl -X PATCH "https://<mgmt-ip>/api/storage/aggregates/7eb630d1-0e55-
4cb6-8d90-957d6f4db54e?simulate=true" -H "accept: application/json" -d
'{"block_storage.hybrid_cache.disk_count":6,"block_storage.hybrid_cache.ra
id_type":"raid4","block_storage.hybrid_cache.raid_size":3}'

# The response:
{
 "warnings": [
   {
     "name": "test",
     "warning": {
       "code": 18316,
       "message": "Operation will lead to creation of new raid group"
     }
   }
 ],
 "num_records": 1,
 "records": [
   {
     "uuid": "7eb630d1-0e55-4cb6-8d90-957d6f4db54e",
     "name": "test",
     "node": {
       "uuid": "30d69eb5-f046-11ec-9bba-005056bba492",
       "name": "node-1",
       "_links": {
         "self": {
           "href": "/api/cluster/nodes/30d69eb5-f046-11ec-9bba-
005056bba492"
         }
       }
```

```
      },
      "space": {
        "block_storage": {
          "size": 833777664
        }
      },
      "block_storage": {
        "primary": {
          "disk_count": 3,
          "disk_class": "virtual",
          "raid_type": "raid_dp",
          "disk_type": "vm_disk"
        },
        "hybrid_cache": {
          "disk_class": "solid_state",
          "disk_type": "ssd",
          "enabled": false,
          "disk_count": 6,
          "raid_type": "raid4",
          "size": 6771179520,
          "simulated_raid_groups": [
            {
              "name": "/test/plex0/rg1",
              "existing_parity_disk_count": 0,
              "existing_data_disk_count": 0,
              "added_parity_disk_count": 1,
              "added_data_disk_count": 2,
              "usable_size": 1880883200,
              "is_partition": false
            },
            {
              "name": "/test/plex0/rg2",
              "existing_parity_disk_count": 0,
              "existing_data_disk_count": 0,
              "added_parity_disk_count": 1,
              "added_data_disk_count": 2,
              "usable_size": 1880883200,
              "is_partition": false
            }
          ]
        },
        "mirror": {
          "enabled": false
        }
      },
      "_links": {
```

```
        "self": {
          "href": "/api/storage/aggregates/7eb630d1-0e55-4cb6-8d90-
957d6f4db54e"
        }
      }
    }
  ]
  }
```

The following example shows the workflow to enable software encryption on an aggregate.

Step 1: Check the current software encryption status of the aggregate.

```
# The API:
/api/storage/aggregates

# The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates/f3aafdc6-be35-4d93-
9590-5a402bffbe4b?fields=data_encryption.software_encryption_enabled" -H
"accept: application/json"

# The response:
{
"uuid": "f3aafdc6-be35-4d93-9590-5a402bffbe4b",
"name": "aggr5",
"data_encryption": {
  "software_encryption_enabled": false
},
}
```

Step 2: Update the aggregate with the encryption status in 'data_encryption.software_encryption_enabled'.
The response to PATCH is a job unless the request is invalid.

```
# The API:
/api/storage/aggregates

# The call:
curl -X PATCH "https://<mgmt-ip>/api/storage/aggregates/f3aafdc6-be35-
4d93-9590-5a402bffbe4b" -H "accept: application/hal+json" -d
'{"data_encryption": {"software_encryption_enabled": "true"}}'

# The response:
{
"job": {
  "uuid": "6b7ab28e-168d-11ea-8a50-0050568eca76",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/6b7ab28e-168d-11ea-8a50-0050568eca76"
    }
  }
}
}
```

Step 3: Wait for the job to finish, then call GET to see the reflected change.

```
# The API:
/api/storage/aggregates

# The call:
curl -X GET "https://<mgmt-ip>/api/storage/aggregates/f3aafdc6-be35-4d93-
9590-5a402bffbe4b?fields=data_encryption.software_encryption_enabled" -H
"accept: application/json"

# The response:
{
"uuid": "f3aafdc6-be35-4d93-9590-5a402bffbe4b",
"name": "aggr5",
"data_encryption": {
  "software_encryption_enabled": true
},
}
```

Delete an aggregate specified by the UUID

DELETE /storage/aggregates/{uuid}

Introduced In: 9.6

Deletes the aggregate specified by the UUID. This request starts a job and returns a link to that job.

Related ONTAP commands

- `storage aggregate delete`

Parameters

Name	Type	In	Required	Description
uuid	string	path	True	Aggregate UUID
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202. • Default value: 1 • Max value: 120 • Min value: 0

Response

```
Status: 202, Accepted
```

Name	Type	Description
job	job_link	

Example response

```
{
  "job": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "string"
  }
}
```

Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
460770	The aggregate delete job failed to delete the aggregate.
460777	Failed to get information on the delete job.
786435	Internal Error. Failed to create a communication handle.
786451	Failed to delete specified aggregate.
786468	VLDB is offline.
786472	Node that hosts the aggregate is offline.
786497	Cannot delete an aggregate that has volumes.
786771	Aggregate does not exist.
786867	Specified aggregate resides on the remote cluster.
786897	Specified aggregate cannot be deleted as it is a switched-over root aggregate.

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

job_link

Name	Type	Description
_links	_links	
uuid	string	The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Retrieve an aggregate specified by the UUID

GET `/storage/aggregates/{uuid}`

Introduced In: 9.6

Retrieves the aggregate specified by the UUID. The recommend query cannot be used for this operation.

Expensive properties

There is an added computational cost to retrieving values for these properties. They are not included by default in GET results and must be explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

- `metric.*`

- `space.block_storage.inactive_user_data`

- `space.block_storage.inactive_user_data_percent`

- `space.footprint`

- `is_spare_low`

- `statistics.*`

Related ONTAP commands

- `storage aggregate show`

Parameters

Name	Type	In	Required	Description
uuid	string	path	True	Aggregate UUID
show_spares	boolean	query	False	If set to 'true', the spares object is returned instead of records to show the spare groups in the cluster that are compatible with the aggregate. The default setting is 'false'. • Introduced in: 9.12

Name	Type	In	Required	Description
flash_pool_eligible	boolean	query	False	If set to 'true' along with show_spares, the spares object is restricted to return spare groups that are compatible with flash pool creation or addition for the specified aggregate. The default setting is 'false'. • Introduced in: 9.12
auto_provision_policy	string	query	False	If set to expand, a query is run on the system for the recommended optimal expansion layout of the aggregate. • Introduced in: 9.8
fields	array[string]	query	False	Specify the fields to return.

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
_tags	array[string]	Tags are an optional way to track the uses of a resource. Tag values must be formatted as key:value strings.
block_storage	block_storage	Configuration information for the locally attached portion of the aggregate. When a cloud store is also used by this aggregate, this is referred to as the performance tier.

Name	Type	Description
cloud_storage	cloud_storage	Configuration information for the cloud storage portion of the aggregate. This is referred to as the capacity tier.
create_time	string	Timestamp of aggregate creation.
data_encryption	data_encryption	
dr_home_node	dr_home_node	Node where the aggregate resides after disaster recovery. The value for this field might differ from the 'node' field during switchover.
home_node	home_node	Node where the aggregate resides after giveback. The value for this field might differ from the value of the 'node' field during takeover.
inactive_data_reporting	inactive_data_reporting	
inode_attributes	inode_attributes	
is_spare_low	boolean	Specifies whether the aggregate is in a spares low condition on any of the RAID groups. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.
metric	metric	The most recent sample of I/O metrics for the aggregate.
name	string	Aggregate name.
node	node	Node where the aggregate currently resides.
recommendation_spares	array[aggregate_spare]	Information on the aggregate's remaining hot spare disks.
sidl_enabled	boolean	Specifies whether or not SIDL is enabled on the aggregate.

Name	Type	Description
snaplock_type	string	SnapLock type.
snapshot	snapshot	
space	space	
state	string	Operational state of the aggregate.
statistics	statistics	The real time I/O statistics for the aggregate.
uuid	string	Aggregate UUID.
volume-count	integer	Number of volumes in the aggregate.

Example response

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "_tags": [
    "team:csi",
    "environment:test"
  ],
  "block_storage": {
    "hybrid_cache": {
      "disk_count": 6,
      "disk_type": "string",
      "raid_size": 24,
      "raid_type": "string",
      "simulated_raid_groups": [
        {
          "name": "string"
        }
      ],
      "size": 1612709888,
      "storage_pools": [
        {
          "storage_pool": {
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "name": "storage_pool_1",
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          }
        }
      ],
      "used": 26501122
    },
    "mirror": {
      "enabled": "",
      "state": "string"
    },
    "plexes": [
      {
        "_links": {
```

```
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "plex0"
      }
    ],
    "primary": {
      "checksum_style": "string",
      "disk_class": "performance",
      "disk_count": 8,
      "disk_type": "string",
      "raid_size": 16,
      "raid_type": "string",
      "simulated_raid_groups": [
        {
          "name": "string",
          "raid_type": "string"
        }
      ]
    },
    "storage_type": "string"
  },
  "cloud_storage": {
    "stores": [
      {
        "cloud_store": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "store1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        },
        "used": 0
      }
    ]
  },
  "create_time": "2018-01-01 16:00:00 +0000",
  "dr_home_node": {
    "name": "node1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "home_node": {
    "_links": {
```

```
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "node1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "inactive_data_reporting": {
      "start_time": "2019-12-12 16:00:00 +0000"
    },
    "inode_attributes": {
      "file_private_capacity": 31136,
      "file_public_capacity": 31136,
      "files_private_used": 502,
      "files_total": 31136,
      "files_used": 97,
      "max_files_available": 31136,
      "max_files_possible": 2844525,
      "max_files_used": 97,
      "used_percent": 5,
      "version": 4
    },
    "is_spare_low": "",
    "metric": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "duration": "PT15S",
      "iops": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "latency": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "status": "ok",
      "throughput": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
```

```
        "timestamp": "2017-01-25 11:20:13 +0000"
      },
      "name": "node1_aggr_1",
      "node": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "node1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "recommendation_spares": [
        {
          "checksum_style": "string",
          "disk_class": "solid_state",
          "disk_type": "string",
          "is_partition": 1,
          "layout_requirements": [
            {
              "aggregate_min_disks": 6,
              "raid_group": {
                "default": 16,
                "max": 28,
                "min": 5
              },
              "raid_type": "string"
            }
          ],
          "node": {
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "name": "node1",
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          },
          "size": 10156769280,
          "syncmirror_pool": "string",
          "total": 10,
          "usable": 9
        }
      ],
      "snaplock_type": "string",
      "snapshot": {
```

```
      "files_total": 10,
      "files_used": 3,
      "max_files_available": 5,
      "max_files_used": 50
    },
    "space": {
      "block_storage": {
        "aggregate_metadata": 2655,
        "aggregate_metadata_percent": 8,
        "available": 10156560384,
        "data_compacted_count": 1990000,
        "data_compaction_space_saved": 1996000,
        "data_compaction_space_saved_percent": 27,
        "full_threshold_percent": 0,
        "inactive_user_data": 304448,
        "inactive_user_data_percent": 0,
        "physical_used": 2461696,
        "physical_used_percent": 50,
        "size": 10156769280,
        "used": 2088960,
        "used_including_snapshot_reserve": 674685,
        "used_including_snapshot_reserve_percent": 35,
        "used_percent": 50,
        "volume_deduplication_shared_count": 1990000,
        "volume_deduplication_space_saved": 1996000,
        "volume_deduplication_space_saved_percent": 27,
        "volume_footprints_percent": 14
      },
      "cloud_storage": {
        "used": 402743264
      },
      "efficiency": {
        "logical_used": 0,
        "ratio": 0,
        "savings": 0
      },
      "efficiency_without_snapshots": {
        "logical_used": 0,
        "ratio": 0,
        "savings": 0
      },
      "efficiency_without_snapshots_flexclones": {
        "logical_used": 0,
        "ratio": 0,
        "savings": 0
      },
```

```
        "footprint": 608896,
        "snapshot": {
          "available": 2000,
          "reserve_percent": 20,
          "total": 5000,
          "used": 3000,
          "used_percent": 45
        }
      },
      "state": "string",
      "statistics": {
        "iops_raw": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "latency_raw": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "status": "ok",
        "throughput_raw": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "timestamp": "2017-01-25 11:20:13 +0000"
      },
      "uuid": "string",
      "volume-count": 0
  }
```

Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
787092	The target field cannot be specified for this operation.
7209049	Cannot perform the operation because the aggregate is currently expanding.

Error Code	Description
8586225	Unexpected error encountered when retrieving metrics and statistics for this aggregate.
19726382	Another provisioning operation is in progress on this cluster. Wait a few minutes, and try the operation again.
19726390	Unable to provide a recommmendation to expand the aggregate.
19726391	Too many unassigned disks visible to the node that owns this aggregate.
19726392	Layout of this aggregate is not a supported configuration.
19726393	Failed to expand the aggregate. Aggregate expansion is not supported on this system.
19726394	Automatic aggregate expansion is not supported on systems with multiple data aggregates.
19726395	Automatic aggregate expansion is not supported when MetroCluster is not configured
19726396	Automatic aggregate expansion is not supported when the DR group is not in a normal state
19726397	Aggregates must contain disks with identical disk-types and disk-sizes.
19726402	Internal error. Unable to determine the MetroCluster configuration state.
19726538	Cannot perform the operation because the aggregate is not in a healthy state.

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

simulated_raid_groups

Name	Type	Description
added_data_disk_count	integer	Number of added data disks in RAID group.
added_parity_disk_count	integer	Number of added parity disks in RAID group.
existing_data_disk_count	integer	Number of existing data disks in the RAID group.
existing_parity_disk_count	integer	Number of existing parity disks in the RAID group.
is_partition	boolean	Indicates whether the disk is partitioned (true) or whole (false).
name	string	Name of the raid group.
usable_size	integer	Usable size of each disk, in bytes.

storage_pool_reference

Shared Storage Pool

Name	Type	Description
_links	_links	
name	string	
uuid	string	

storage_pools

Name	Type	Description
allocation_units_count	integer	Allocation count of storage pool.
storage_pool	storage_pool_reference	Shared Storage Pool

hybrid_cache

Contains the configuration for the hybrid cache. The hybrid cache is made up of either whole SSDs or storage pool SSDs.

Name	Type	Description
disk_count	integer	Number of disks used in the cache tier of the aggregate. Only provided when hybrid_cache.enabled is 'true'.
disk_type	string	Type of disk being used by the aggregate's cache tier.
enabled	boolean	Specifies whether the aggregate uses HDDs with SSDs as a cache.
raid_size	integer	Option to specify the maximum number of disks that can be included in a RAID group.
raid_type	string	RAID type for SSD cache of the aggregate. Only provided when hybrid_cache.enabled is 'true'.
simulated_raid_groups	array[simulated_raid_groups]	
size	integer	Total usable space in bytes of SSD cache. Only provided when hybrid_cache.enabled is 'true'.
storage_pools	array[storage_pools]	List of storage pool properties and allocation_units_count for aggregate.
used	integer	Space used in bytes of SSD cache. Only provided when hybrid_cache.enabled is 'true'.

mirror

Name	Type	Description
enabled	boolean	Aggregate is SyncMirror protected
state	string	

plex_reference

Plex

Name	Type	Description
_links	_links	
name	string	

simulated_raid_groups

Name	Type	Description
added_data_disk_count	integer	Number of added data disks in RAID group.
added_parity_disk_count	integer	Number of added parity disks in RAID group.
data_disk_count	integer	Number of data disks in RAID group.
existing_data_disk_count	integer	Number of existing data disks in the RAID group.
existing_parity_disk_count	integer	Number of existing parity disks in the RAID group.
is_partition	boolean	Indicates whether the disk is partitioned (true) or whole (false).
name	string	Name of the raid group.
parity_disk_count	integer	Number of parity disks in RAID group.
raid_type	string	RAID type of the aggregate.
usable_size	integer	Usable size of each disk, in bytes.

primary

Configuration information for the primary storage portion of the aggregate. This excludes the hybrid cache details.

Name	Type	Description
checksum_style	string	The checksum style used by the aggregate.
disk_class	string	The class of disks being used by the aggregate.
disk_count	integer	Number of disks used in the aggregate. This includes parity disks, but excludes disks in the hybrid cache.
disk_type	string	The type of disk being used by the aggregate.
raid_size	integer	Option to specify the maximum number of disks that can be included in a RAID group.
raid_type	string	RAID type of the aggregate.
simulated_raid_groups	array[simulated_raid_groups]	

block_storage

Configuration information for the locally attached portion of the aggregate. When a cloud store is also used by this aggregate, this is referred to as the performance tier.

Name	Type	Description
hybrid_cache	hybrid_cache	Contains the configuration for the hybrid cache. The hybrid cache is made up of either whole SSDs or storage pool SSDs.
mirror	mirror	
plexes	array[plex_reference]	Plex reference for each plex in the aggregate.
primary	primary	Configuration information for the primary storage portion of the aggregate. This excludes the hybrid cache details.
storage_type	string	Type of aggregate.

Name	Type	Description
uses_partitions	boolean	If true, aggregate is using shared disks.

cloud_store

Cloud store

Name	Type	Description
_links	_links	
name	string	
uuid	string	

cloud_storage_tier

Name	Type	Description
cloud_store	cloud_store	Cloud store
used	integer	Capacity used in bytes in the cloud store by this aggregate. This is a cached value calculated every 5 minutes.

cloud_storage

Configuration information for the cloud storage portion of the aggregate. This is referred to as the capacity tier.

Name	Type	Description
attach_eligible	boolean	Specifies whether the aggregate is eligible for a cloud store to be attached.
migrate_threshold	integer	Specifies the minimum percentage of performance tier free space that must exist in order for migration of data from the capacity tier to performance tier to be allowed. Only valid for PATCH operations.
stores	array[cloud_storage_tier]	Configuration information for each cloud storage portion of the aggregate.

Name	Type	Description
tiering_fullness_threshold	integer	The percentage of space in the performance tier that must be used before data is tiered out to the cloud store. Only valid for PATCH operations.

data_encryption

Name	Type	Description
drive_protection_enabled	boolean	Specifies whether the aggregate uses self-encrypting drives with data protection enabled.
software_encryption_enabled	boolean	Specifies whether NetApp aggregate encryption is enabled. All data in the aggregate is encrypted.

dr_home_node

Node where the aggregate resides after disaster recovery. The value for this field might differ from the 'node' field during switchover.

Name	Type	Description
name	string	
uuid	string	

home_node

Node where the aggregate resides after giveback. The value for this field might differ from the value of the 'node' field during takeover.

Name	Type	Description
_links	_links	
name	string	
uuid	string	

inactive_data_reporting

Name	Type	Description
enabled	boolean	Specifes whether or not inactive data reporting is enabled on the aggregate.

Name	Type	Description
start_time	string	Timestamp at which inactive data reporting was enabled on the aggregate.

inode_attributes

Name	Type	Description
file_private_capacity	integer	Number of files that can currently be stored on disk for system metadata files. This number will dynamically increase as more system files are created. This is an advanced property; there is an added computationl cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.
file_public_capacity	integer	Number of files that can currently be stored on disk for user-visible files. This number will dynamically increase as more user-visible files are created. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.
files_private_used	integer	Number of system metadata files used. If the referenced file system is restricted or offline, a value of 0 is returned. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.

Name	Type	Description
files_total	integer	Maximum number of user-visible files that this referenced file system can currently hold. If the referenced file system is restricted or offline, a value of 0 is returned.
files_used	integer	Number of user-visible files used in the referenced file system. If the referenced file system is restricted or offline, a value of 0 is returned.
max_files_available	integer	The count of the maximum number of user-visible files currently allowable on the referenced file system.
max_files_possible	integer	The largest value to which the maxfiles-available parameter can be increased by reconfiguration, on the referenced file system.
max_files_used	integer	The number of user-visible files currently in use on the referenced file system.
used_percent	integer	The percentage of disk space currently in use based on user-visible file count on the referenced file system.
version	integer	The inofile-version of the aggregate. If the referenced file system is restricted or offline, a value of 0 is returned. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.

iops

The rate of I/O operations observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

latency

The round trip latency in microseconds observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

throughput

The rate of throughput bytes per second observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.

Name	Type	Description
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

metric

The most recent sample of I/O metrics for the aggregate.

Name	Type	Description
_links	_links	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	iops	The rate of I/O operations observed at the storage object.
latency	latency	The round trip latency in microseconds observed at the storage object.

Name	Type	Description
status	string	Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	throughput	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

node

Node where the aggregate currently resides.

Name	Type	Description
_links	_links	
name	string	
uuid	string	

raid_group

Name	Type	Description
default	integer	Default number of disks in a RAID group.

Name	Type	Description
max	integer	Maximum number of disks allowed in a RAID group.
min	integer	Minimum number of disks allowed in a RAID group.

layout_requirement

Name	Type	Description
aggregate_min_disks	integer	Minimum number of disks to create an aggregate.
default	boolean	Indicates if this RAID type is the default.
raid_group	raid_group	
raid_type	string	RAID type.

node

Node where the spares are assigned.

Name	Type	Description
_links	_links	
name	string	
uuid	string	

aggregate_spare

Name	Type	Description
checksum_style	string	The checksum type that has been assigned to the spares.
disk_class	string	Disk class of spares.
disk_type	string	Type of disk.
is_partition	boolean	Indicates whether a disk is partitioned (true) or whole (false)
layout_requirements	array[layout_requirement]	Available RAID protections and their restrictions.

Name	Type	Description
node	node	Node where the spares are assigned.
size	integer	Usable size of each spare, in bytes.
syncmirror_pool	string	SyncMirror spare pool.
total	integer	Total number of spares in the bucket. The total spare count for each class of spares also includes reserved spare capacity recommended by ONTAP best practices. • example: 10 • readOnly: 1 • Introduced in: 9.11 • x-nullable: true
usable	integer	Total number of usable spares in the bucket. The usable count for each class of spares does not include reserved spare capacity recommended by ONTAP best practices. • example: 9 • readOnly: 1 • Introduced in: 9.6 • x-nullable: true

snapshot

Name	Type	Description
files_total	integer	Total files allowed in Snapshot copies
files_used	integer	Total files created in Snapshot copies
max_files_available	integer	Maximum files available for Snapshot copies

Name	Type	Description
max_files_used	integer	Files in use by Snapshot copies

block_storage

Name	Type	Description
aggregate_metadata	integer	Space used by different metafiles and internal operations inside the aggregate, in bytes.
aggregate_metadata_percent	integer	Aggregate metadata as a percentage.
available	integer	Space available in bytes.
data_compacted_count	integer	Amount of compacted data in bytes.
data_compaction_space_saved	integer	Space saved in bytes by compacting the data.
data_compaction_space_saved_percent	integer	Percentage saved by compacting the data.
full_threshold_percent	integer	The aggregate used percentage at which 'monitor.volume.full' EMS is generated.
inactive_user_data	integer	The size that is physically used in the block storage and has a cold temperature, in bytes. This property is only supported if the aggregate is either attached to a cloud store or can be attached to a cloud store. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either block_storage.inactive_user_data or **.

Name	Type	Description
inactive_user_data_percent	integer	The percentage of inactive user data in the block storage. This property is only supported if the aggregate is either attached to a cloud store or can be attached to a cloud store. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either block_storage.inactive_user_data_percent or **.
physical_used	integer	Total physical used size of an aggregate in bytes.
physical_used_percent	integer	Physical used percentage.
size	integer	Total usable space in bytes, not including WAFL reserve and aggregate Snapshot copy reserve.
used	integer	Space used or reserved in bytes. Includes volume guarantees and aggregate metadata.
used_including_snapshot_reserve	integer	Total used including the Snapshot copy reserve, in bytes.
used_including_snapshot_reserve_percent	integer	Total used including the Snapshot reserve as a percentage.
used_percent	integer	Aggregate used percentage.
volume_deduplication_shared_count	integer	Amount of shared bytes counted by storage efficiency.
volume_deduplication_space_saved	integer	Amount of space saved in bytes by storage efficiency.
volume_deduplication_space_saved_percent	integer	Percentage of space saved by storage efficiency.

Name	Type	Description
volume_footprints_percent	integer	A summation of volume footprints inside the aggregate, as a percentage. A volume's footprint is the amount of space being used for the volume in the aggregate.

cloud_storage

Name	Type	Description
used	integer	Used space in bytes in the cloud store. Only applicable for aggregates with a cloud store tier.

efficiency

Storage efficiency.

Name	Type	Description
auto_adaptive_compression_savings	boolean	Indicates whether or not aggregate has auto adaptive compression savings.
cross_volume_background_dedupe	boolean	Indicates whether or not cross volume background deduplication is enabled.
cross_volume_dedupe_savings	boolean	Indicates whether or not aggregate has cross volume deduplication savings.
cross_volume_inline_dedupe	boolean	Indicates whether or not cross volume inline deduplication is enabled.
logical_used	integer	Logical used
ratio	number	Data reduction ratio (logical_used / used)
savings	integer	Space saved by storage efficiencies (logical_used - used)

efficiency_without_snapshots

Storage efficiency that does not include the savings provided by Snapshot copies.

Name	Type	Description
logical_used	integer	Logical used
ratio	number	Data reduction ratio (logical_used / used)
savings	integer	Space saved by storage efficiencies (logical_used - used)

efficiency_without_snapshots_flexclones

Storage efficiency that does not include the savings provided by Snapshot copies and Flexclone volumes.

Name	Type	Description
logical_used	integer	Logical used
ratio	number	Data reduction ratio (logical_used / used)
savings	integer	Space saved by storage efficiencies (logical_used - used)

snapshot

Name	Type	Description
available	integer	Available space for Snapshot copies in bytes
reserve_percent	integer	Percentage of space reserved for Snapshot copies
total	integer	Total space for Snapshot copies in bytes
used	integer	Space used by Snapshot copies in bytes
used_percent	integer	Percentage of disk space used by Snapshot copies

space

Name	Type	Description
block_storage	block_storage	

Name	Type	Description
cloud_storage	cloud_storage	
efficiency	efficiency	Storage efficiency.
efficiency_without_snapshots	efficiency_without_snapshots	Storage efficiency that does not include the savings provided by Snapshot copies.
efficiency_without_snapshots_flex clones	efficiency_without_snapshots_flex clones	Storage efficiency that does not include the savings provided by Snapshot copies and Flexclone volumes.
footprint	integer	A summation of volume footprints (including volume guarantees), in bytes. This includes all of the volume footprints in the block_storage tier and the cloud_storage tier. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.
snapshot	snapshot	

iops_raw

The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.

Name	Type	Description
write	integer	Peformance metric for write I/O operations.

latency_raw

The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

throughput_raw

Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

statistics

The real time I/O statistics for the aggregate.

Name	Type	Description
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Update an aggregate specified by the UUID

PATCH /storage/aggregates/{uuid}

Introduced In: 9.6

Updates the aggregate specified by the UUID with the properties in the body. This request starts a job and returns a link to that job.

Related ONTAP commands

- storage aggregate add-disks
- storage aggregate mirror
- storage aggregate modify
- storage aggregate relocation start
- storage aggregate rename

Parameters

Name	Type	In	Required	Description
uuid	string	path	True	Aggregate UUID

Name	Type	In	Required	Description
auto_provision_policy	string	query	False	If set to expand, the PATCH operation runs the recommended expansion of the aggregate. • Introduced in: 9.8
simulate	boolean	query	False	If set to true, the PATCH operation runs a simulated aggregate expansion with the provided input disk count and returns the proposed size of the new aggregate along with any associated warnings. • Introduced in: 9.8
disk_size	integer	query	False	If set, PATCH only selects disks of the specified size.

Name	Type	In	Required	Description
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202. • Default value: 1 • Max value: 120 • Min value: 0

Request Body

Name	Type	Description
_links	_links	
_tags	array[string]	Tags are an optional way to track the uses of a resource. Tag values must be formatted as key:value strings.
block_storage	block_storage	Configuration information for the locally attached portion of the aggregate. When a cloud store is also used by this aggregate, this is referred to as the performance tier.

Name	Type	Description
cloud_storage	cloud_storage	Configuration information for the cloud storage portion of the aggregate. This is referred to as the capacity tier.
create_time	string	Timestamp of aggregate creation.
data_encryption	data_encryption	
dr_home_node	dr_home_node	Node where the aggregate resides after disaster recovery. The value for this field might differ from the 'node' field during switchover.
home_node	home_node	Node where the aggregate resides after giveback. The value for this field might differ from the value of the 'node' field during takeover.
inactive_data_reporting	inactive_data_reporting	
is_spare_low	boolean	Specifies whether the aggregate is in a spares low condition on any of the RAID groups. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.
name	string	Aggregate name.
node	node	Node where the aggregate currently resides.
recommendation_spares	array[aggregate_spare]	Information on the aggregate's remaining hot spare disks.
sidl_enabled	boolean	Specifies whether or not SIDL is enabled on the aggregate.
snapshot	snapshot	
space	space	
state	string	Operational state of the aggregate.

Name	Type	Description
uuid	string	Aggregate UUID.
volume-count	integer	Number of volumes in the aggregate.

Example request

```json
{
  "_tags": [
    "team:csi",
    "environment:test"
  ],
  "block_storage": {
    "hybrid_cache": {
      "disk_count": 6,
      "disk_type": "string",
      "raid_size": 24,
      "raid_type": "string",
      "simulated_raid_groups": [
        {
          "name": "string"
        }
      ],
      "size": 1612709888,
      "storage_pools": [
        {
          "storage_pool": {
            "name": "storage_pool_1",
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          }
        }
      ],
      "used": 26501122
    },
    "mirror": {
      "enabled": "",
      "state": "string"
    },
    "plexes": [
      {
        "name": "plex0"
      }
    ],
    "primary": {
      "disk_count": 8,
      "disk_type": "string",
      "raid_size": 16,
      "raid_type": "string",
      "simulated_raid_groups": [
        {
          "name": "string",
```

```
            "raid_type": "string"
          }
        ]
      },
      "storage_type": "string"
    },
    "cloud_storage": {
      "stores": [
        {
          "cloud_store": {
            "name": "store1",
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          },
          "used": 0
        }
      ]
    },
    "create_time": "2018-01-01 16:00:00 +0000",
    "dr_home_node": {
      "name": "node1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "home_node": {
      "name": "node1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "inactive_data_reporting": {
      "start_time": "2019-12-12 16:00:00 +0000"
    },
    "is_spare_low": "",
    "name": "node1_aggr_1",
    "node": {
      "name": "node1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "recommendation_spares": [
      {
        "checksum_style": "string",
        "disk_class": "solid_state",
        "disk_type": "string",
        "is_partition": 1,
        "layout_requirements": [
          {
            "aggregate_min_disks": 6,
            "raid_group": {
              "default": 16,
```

```
          "max": 28,
          "min": 5
        },
        "raid_type": "string"
      }
    ],
    "node": {
      "name": "node1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "size": 10156769280,
    "syncmirror_pool": "string",
    "total": 10,
    "usable": 9
  }
],
"snapshot": {
  "files_total": 10,
  "files_used": 3,
  "max_files_available": 5,
  "max_files_used": 50
},
"space": {
  "block_storage": {
    "aggregate_metadata": 2655,
    "aggregate_metadata_percent": 8,
    "available": 10156560384,
    "data_compacted_count": 1990000,
    "data_compaction_space_saved": 1996000,
    "data_compaction_space_saved_percent": 27,
    "full_threshold_percent": 0,
    "inactive_user_data": 304448,
    "inactive_user_data_percent": 0,
    "physical_used": 2461696,
    "physical_used_percent": 50,
    "size": 10156769280,
    "used": 2088960,
    "used_including_snapshot_reserve": 674685,
    "used_including_snapshot_reserve_percent": 35,
    "used_percent": 50,
    "volume_deduplication_shared_count": 1990000,
    "volume_deduplication_space_saved": 1996000,
    "volume_deduplication_space_saved_percent": 27,
    "volume_footprints_percent": 14
  },
  "cloud_storage": {
```

```
        "used": 402743264
      },
      "efficiency": {
        "logical_used": 0,
        "ratio": 0,
        "savings": 0
      },
      "efficiency_without_snapshots": {
        "logical_used": 0,
        "ratio": 0,
        "savings": 0
      },
      "efficiency_without_snapshots_flexclones": {
        "logical_used": 0,
        "ratio": 0,
        "savings": 0
      },
      "footprint": 608896,
      "snapshot": {
        "available": 2000,
        "reserve_percent": 20,
        "total": 5000,
        "used": 3000,
        "used_percent": 45
      }
    },
    "state": "string",
    "uuid": "string",
    "volume-count": 0
  }
```

Response

```
Status: 200, Ok
```

Name	Type	Description
job	job_link	
num_records	integer	Number of records
records	array[aggregate]	

Name	Type	Description
warnings	array[aggregate_warning]	List of validation warnings and remediation advice for the aggregate simulate behavior.

Example response

```
{
  "job": {
    "uuid": "string"
  },
  "num_records": 1,
  "records": [
    {
      "_tags": [
        "team:csi",
        "environment:test"
      ],
      "block_storage": {
        "hybrid_cache": {
          "disk_count": 6,
          "disk_type": "string",
          "raid_size": 24,
          "raid_type": "string",
          "simulated_raid_groups": [
            {
              "name": "string"
            }
          ],
          "size": 1612709888,
          "storage_pools": [
            {
              "storage_pool": {
                "name": "storage_pool_1",
                "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
              }
            }
          ],
          "used": 26501122
        },
        "mirror": {
          "enabled": "",
          "state": "string"
        },
        "plexes": [
          {
            "name": "plex0"
          }
        ],
        "primary": {
          "disk_count": 8,
```

```
          "disk_type": "string",
          "raid_size": 16,
          "raid_type": "string",
          "simulated_raid_groups": [
            {
              "name": "string",
              "raid_type": "string"
            }
          ]
        },
        "storage_type": "string"
      },
      "cloud_storage": {
        "stores": [
          {
            "cloud_store": {
              "name": "store1",
              "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
            },
            "used": 0
          }
        ]
      },
      "create_time": "2018-01-01 16:00:00 +0000",
      "dr_home_node": {
        "name": "node1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "home_node": {
        "name": "node1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "inactive_data_reporting": {
        "start_time": "2019-12-12 16:00:00 +0000"
      },
      "is_spare_low": "",
      "name": "node1_aggr_1",
      "node": {
        "name": "node1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "recommendation_spares": [
        {
          "checksum_style": "string",
          "disk_class": "solid_state",
          "disk_type": "string",
```

```
        "is_partition": 1,
        "layout_requirements": [
          {
            "aggregate_min_disks": 6,
            "raid_group": {
              "default": 16,
              "max": 28,
              "min": 5
            },
            "raid_type": "string"
          }
        ],
        "node": {
          "name": "node1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        },
        "size": 10156769280,
        "syncmirror_pool": "string",
        "total": 10,
        "usable": 9
      }
    ],
    "snapshot": {
      "files_total": 10,
      "files_used": 3,
      "max_files_available": 5,
      "max_files_used": 50
    },
    "space": {
      "block_storage": {
        "aggregate_metadata": 2655,
        "aggregate_metadata_percent": 8,
        "available": 10156560384,
        "data_compacted_count": 1990000,
        "data_compaction_space_saved": 1996000,
        "data_compaction_space_saved_percent": 27,
        "full_threshold_percent": 0,
        "inactive_user_data": 304448,
        "inactive_user_data_percent": 0,
        "physical_used": 2461696,
        "physical_used_percent": 50,
        "size": 10156769280,
        "used": 2088960,
        "used_including_snapshot_reserve": 674685,
        "used_including_snapshot_reserve_percent": 35,
        "used_percent": 50,
```

```
            "volume_deduplication_shared_count": 1990000,
            "volume_deduplication_space_saved": 1996000,
            "volume_deduplication_space_saved_percent": 27,
            "volume_footprints_percent": 14
          },
          "cloud_storage": {
            "used": 402743264
          },
          "efficiency": {
            "logical_used": 0,
            "ratio": 0,
            "savings": 0
          },
          "efficiency_without_snapshots": {
            "logical_used": 0,
            "ratio": 0,
            "savings": 0
          },
          "efficiency_without_snapshots_flexclones": {
            "logical_used": 0,
            "ratio": 0,
            "savings": 0
          },
          "footprint": 608896,
          "snapshot": {
            "available": 2000,
            "reserve_percent": 20,
            "total": 5000,
            "used": 3000,
            "used_percent": 45
          }
        },
        "state": "string",
        "uuid": "string",
        "volume-count": 0
      }
    ],
    "warnings": [
      {
        "action": {
          "arguments": [
            "string"
          ],
          "message": "string"
        },
        "name": "string",
```

```
        "warning": {
          "arguments": [
            "string"
          ],
          "message": "string"
        }
      }
    ]
  }
```

Response

```
Status: 202, Accepted
```

Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
262247	The value is invalid for the field.
460777	Failed to get information on the job.
786434	Cannot connect to node where the aggregate resides.
786435	Internal Error. Failed to create a communication handle.
786439	An aggregate already uses the specified name.
786447	Failed to modify the aggregate.
786456	Failed to add disks to the aggregate.
786458	Failed to rename aggregate.
786465	Failed to take aggregate offline because of the reason outlined in the message.
786467	Failed to take aggregate online because of the reason outlined in the message.
786468	VLDB is offline.
786472	Node that hosts the aggregate is offline.
786479	Cannot find node ID for the node.
786491	Not enough spares on the node.

Error Code	Description
786730	Internal Error
786771	Aggregate does not exist.
786787	Aggregate is not online.
786808	Aggregate mirror failed.
786867	Specified aggregate resides on the remote cluster.
786911	Not every node in the cluster has the Data ONTAP version required for the feature.
786923	This operation is not allowed during the pre-commit phase of a 7-mode to clustered Data ONTAP transition.
786924	Internal error for an aggregate that is in the pre-commit phase of a 7-mode to clustered Data ONTAP transition.
786955	Modifying raidtype to raid_tec requires a minimum of six disks in the RAID Group.
786956	Modifying raidtype to raid_dp requires a minimum of four disks in the RAID Group.
786965	Spare Selection in userspace failed.
787046	Mirroring of a FabricPool is not allowed.
787092	The target field cannot be specified for this operation.
787144	Aggregate is not a FabricPool.
787156	Modifying the attributes of mirror object store is not allowed.
787169	Only one field can be modified per operation.
787170	Failed to patch the "block_storage.primary.disk_count" because the disk count specified is smaller than existing disk count.
787172	This query is only allowed during the modification of the specified field.
787178	Unmirroring an aggregate with a PATCH operation is not supported.
787187	Internal error. Failed to check if the aggregate is a FabricPool.
787266	Invalid aggregate state. This state is not supported for a PATCH operation.
787273	Allocation unit count is not valid.
787274	Raidtype is not valid.
787275	Patch request with multiple records is not valid.

Error Code	Description
787276	Storage pool name and uuid do not match.
787277	Storage pool name and uuid are empty.
787278	Incorrect storage pool name specified.
787279	Incorrect storage pool uuid specified.
787280	Allocation unit count specified is smaller than the existing allocation unit count.
787281	Cannot modify RAID type of aggregate hybrid cache tier.
787282	RAID group must be specified on a disk addition to an aggregate with mixed RAID types.
787283	RAID group must be specified on a disk addition to a Flash Pool aggregate.
787284	The specified RAID group uses capacity from one or more storage pools.
787287	Cannot add physical SSD cache because aggregate uses cache capacity from a storage pool.
787288	Cannot add storage pool units because aggregate uses physical SSD cache.
787289	Incorrect raid_group specified during first time addition of physical SSD cache to an aggregate.
787291	Cannot specify RAID group which is located on the aggregate primary tier.
787293	Cannot specify RAID group which is located on the aggregate cache tier.
787294	This query is only allowed during the modification of the specific fields.
787295	The storage pool allocation units count is required.
1258699	Cannot use all the disks specified for the requested operation.
1263500	Operation will lead to creation of new raid group.
1263501	Operation will exceed half of the maximum volume sizes allowed on the node.
1263502	One spare data partition from at least one of the chosen root-data1-data2 disks will not be used.
1263503	Operation will lead to downsizing of one or more disks.
1263504	Operation will lead to a spares low condition.
1263598	One or more selected disks will be partitioned.
1263624	Operation will lead to a no sparecore condition.

Error Code	Description
2425736	No matching node found for the UUID provided.
7208962	Aggregate in an inconsistent state.
7208993	Failed to offline as the volume is being used.
7209033	CIFS open files prevent operation.
7209049	Cannot perform the operation because the aggregate is currently expanding.
7209075	Cannot perform the operation because the volume size limit for this system type would be exceeded.
7209090	Inconsistent state.
7209183	Volume is a partial volume.
7209229	This version of ONTAP does not recognize the filesystem. It is probably from a later version of the software and is being left offline.
7209246	The specified operation could not be completed as the volume is currently busy.
7209247	The volume was not found.
7209263	Container has failed.
7209271	wafliron is currently active.
7209275	Container was created in an unclustered ONTAP deployment.
7209463	Nvfile replay pending.
7209966	Another online request is already in progress for aggregate. The previous online request is waiting on a response from the licensing manager.
11206666	Storage pool is not healthy.
11210659	Aggregate is not online.
11210662	Adding capacity from storage pool to a mirrored aggregate is not supported.
11210667	Storage pool does not have enough spare allocation units.
11210670	Cannot add capacity from storage pool to aggregate, because currently allocated capacity to the aggregate does not span across all disks belonging to the storage pool.
11210672	Cannot grow aggregate as no capacity is allocated to it from storage pool.
11210673	Mixing of physical SSDs and capacity from a storage pool is not allowed in same aggregate.

Error Code	Description
11210675	Capacity in storage pool belongs to different fault isolation domain than aggregate.
11210678	Storage pool does not have enough disks to create RAID groups of same raid type as that of already allocated cache tier.
11210679	Storage pool does not have enough disks to create RAID groups of type RAID-DP.
11210680	Storage pool does not have enough disks to create RAID groups of same raid type as that of the aggregate.
11210685	Storage pool does not have enough disks to create RAID groups of type RAID-TEC.
11210688	Capacity from storage pool cannot be added to an SSD aggregate and a Flash Pool.
13108106	Cannot run aggregate relocation because volume expand is in progress.
19726347	There are a number of unassigned disks visible to the node that owns this aggregate.
19726382	Another provisioning operation is in progress on this cluster. Wait a few minutes, and try the operation again.
19726390	Unable to automatically expand this aggregate.
19726391	Too many unassigned disks visible to the node that owns this aggregate.
19726392	Layout of this aggregate is not a supported configuration.
19726393	Failed to expand the aggregate. Aggregate expansion is not supported on this system.
19726394	Automatic aggregate expansion is not supported on systems with multiple data aggregates.
19726395	Automatic aggregate expansion is not supported when MetroCluster is not configured.
19726396	Automatic aggregate expansion is not supported when the DR group is not in a normal state.
19726397	Aggregates must contain disks with identical disk-types and disk-sizes.
19726402	Internal error. Unable to determine the MetroCluster configuration state.
19726538	Cannot perform the operation because the aggregate is not in a healthy state.

Error Code	Description
26542083	Destination node is at higher Data ONTAP version than source node.
26542084	Source node is at higher Data ONTAP version than destination node.
26542097	Unable to get D-blade ID of destination.
26542101	Unable to contact the source node.
26542102	Unable to contact the destination node.
26542120	An SVM migrate operation is in progress. When the migrate operation completes, try the operation again.
26542121	A MetroCluster disaster recovery operation is in progress. When the recovery operation completes, try the operation again.
196608334	Failed to modify the aggregate because it contains NAE volumes.
196608335	Failed to modify the aggregate because it contains non-encrypted volumes.

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

simulated_raid_groups

Name	Type	Description
added_data_disk_count	integer	Number of added data disks in RAID group.
added_parity_disk_count	integer	Number of added parity disks in RAID group.
existing_data_disk_count	integer	Number of existing data disks in the RAID group.
existing_parity_disk_count	integer	Number of existing parity disks in the RAID group.
is_partition	boolean	Indicates whether the disk is partitioned (true) or whole (false).
name	string	Name of the raid group.
usable_size	integer	Usable size of each disk, in bytes.

storage_pool_reference

Shared Storage Pool

Name	Type	Description
name	string	
uuid	string	

storage_pools

Name	Type	Description
allocation_units_count	integer	Allocation count of storage pool.

Name	Type	Description
storage_pool	storage_pool_reference	Shared Storage Pool

hybrid_cache

Contains the configuration for the hybrid cache. The hybrid cache is made up of either whole SSDs or storage pool SSDs.

Name	Type	Description
disk_count	integer	Number of disks used in the cache tier of the aggregate. Only provided when hybrid_cache.enabled is 'true'.
disk_type	string	Type of disk being used by the aggregate's cache tier.
enabled	boolean	Specifies whether the aggregate uses HDDs with SSDs as a cache.
raid_size	integer	Option to specify the maximum number of disks that can be included in a RAID group.
raid_type	string	RAID type for SSD cache of the aggregate. Only provided when hybrid_cache.enabled is 'true'.
simulated_raid_groups	array[simulated_raid_groups]	
size	integer	Total usable space in bytes of SSD cache. Only provided when hybrid_cache.enabled is 'true'.
storage_pools	array[storage_pools]	List of storage pool properties and allocation_units_count for aggregate.
used	integer	Space used in bytes of SSD cache. Only provided when hybrid_cache.enabled is 'true'.

mirror

Name	Type	Description
enabled	boolean	Aggregate is SyncMirror protected
state	string	

plex_reference

Plex

Name	Type	Description
name	string	

simulated_raid_groups

Name	Type	Description
added_data_disk_count	integer	Number of added data disks in RAID group.
added_parity_disk_count	integer	Number of added parity disks in RAID group.
data_disk_count	integer	Number of data disks in RAID group.
existing_data_disk_count	integer	Number of existing data disks in the RAID group.
existing_parity_disk_count	integer	Number of existing parity disks in the RAID group.
is_partition	boolean	Indicates whether the disk is partitioned (true) or whole (false).
name	string	Name of the raid group.
parity_disk_count	integer	Number of parity disks in RAID group.
raid_type	string	RAID type of the aggregate.
usable_size	integer	Usable size of each disk, in bytes.

primary

Configuration information for the primary storage portion of the aggregate. This excludes the hybrid cache

details.

Name	Type	Description
disk_count	integer	Number of disks used in the aggregate. This includes parity disks, but excludes disks in the hybrid cache.
disk_type	string	The type of disk being used by the aggregate.
raid_size	integer	Option to specify the maximum number of disks that can be included in a RAID group.
raid_type	string	RAID type of the aggregate.
simulated_raid_groups	array[simulated_raid_groups]	

block_storage

Configuration information for the locally attached portion of the aggregate. When a cloud store is also used by this aggregate, this is referred to as the performance tier.

Name	Type	Description
hybrid_cache	hybrid_cache	Contains the configuration for the hybrid cache. The hybrid cache is made up of either whole SSDs or storage pool SSDs.
mirror	mirror	
plexes	array[plex_reference]	Plex reference for each plex in the aggregate.
primary	primary	Configuration information for the primary storage portion of the aggregate. This excludes the hybrid cache details.
storage_type	string	Type of aggregate.
uses_partitions	boolean	If true, aggregate is using shared disks.

cloud_store

Cloud store

Name	Type	Description
_links	_links	
name	string	
uuid	string	

cloud_storage_tier

Name	Type	Description
cloud_store	cloud_store	Cloud store
used	integer	Capacity used in bytes in the cloud store by this aggregate. This is a cached value calculated every 5 minutes.

cloud_storage

Configuration information for the cloud storage portion of the aggregate. This is referred to as the capacity tier.

Name	Type	Description
attach_eligible	boolean	Specifies whether the aggregate is eligible for a cloud store to be attached.
migrate_threshold	integer	Specifies the minimum percentage of performance tier free space that must exist in order for migration of data from the capacity tier to performance tier to be allowed. Only valid for PATCH operations.
stores	array[cloud_storage_tier]	Configuration information for each cloud storage portion of the aggregate.
tiering_fullness_threshold	integer	The percentage of space in the performance tier that must be used before data is tiered out to the cloud store. Only valid for PATCH operations.

data_encryption

Name	Type	Description
drive_protection_enabled	boolean	Specifies whether the aggregate uses self-encrypting drives with data protection enabled.
software_encryption_enabled	boolean	Specifies whether NetApp aggregate encryption is enabled. All data in the aggregate is encrypted.

dr_home_node

Node where the aggregate resides after disaster recovery. The value for this field might differ from the 'node' field during switchover.

Name	Type	Description
name	string	
uuid	string	

home_node

Node where the aggregate resides after giveback. The value for this field might differ from the value of the 'node' field during takeover.

Name	Type	Description
_links	_links	
name	string	
uuid	string	

inactive_data_reporting

Name	Type	Description
enabled	boolean	Specifes whether or not inactive data reporting is enabled on the aggregate.
start_time	string	Timestamp at which inactive data reporting was enabled on the aggregate.

inode_attributes

Name	Type	Description
file_private_capacity	integer	Number of files that can currently be stored on disk for system metadata files. This number will dynamically increase as more system files are created. This is an advanced property; there is an added computationl cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.
file_public_capacity	integer	Number of files that can currently be stored on disk for user-visible files. This number will dynamically increase as more user-visible files are created. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.
files_private_used	integer	Number of system metadata files used. If the referenced file system is restricted or offline, a value of 0 is returned. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.
files_total	integer	Maximum number of user-visible files that this referenced file system can currently hold. If the referenced file system is restricted or offline, a value of 0 is returned.

Name	Type	Description
files_used	integer	Number of user-visible files used in the referenced file system. If the referenced file system is restricted or offline, a value of 0 is returned.
max_files_available	integer	The count of the maximum number of user-visible files currently allowable on the referenced file system.
max_files_possible	integer	The largest value to which the maxfiles-available parameter can be increased by reconfiguration, on the referenced file system.
max_files_used	integer	The number of user-visible files currently in use on the referenced file system.
used_percent	integer	The percentage of disk space currently in use based on user-visible file count on the referenced file system.
version	integer	The inofile-version of the aggregate. If the referenced file system is restricted or offline, a value of 0 is returned. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.

iops

The rate of I/O operations observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

latency

The round trip latency in microseconds observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

throughput

The rate of throughput bytes per second observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.

Name	Type	Description
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

metric

The most recent sample of I/O metrics for the aggregate.

Name	Type	Description
_links	_links	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	iops	The rate of I/O operations observed at the storage object.
latency	latency	The round trip latency in microseconds observed at the storage object.

Name	Type	Description
status	string	Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	throughput	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

node

Node where the aggregate currently resides.

Name	Type	Description
_links	_links	
name	string	
uuid	string	

raid_group

Name	Type	Description
default	integer	Default number of disks in a RAID group.

Name	Type	Description
max	integer	Maximum number of disks allowed in a RAID group.
min	integer	Minimum number of disks allowed in a RAID group.

layout_requirement

Name	Type	Description
aggregate_min_disks	integer	Minimum number of disks to create an aggregate.
default	boolean	Indicates if this RAID type is the default.
raid_group	raid_group	
raid_type	string	RAID type.

node

Node where the spares are assigned.

Name	Type	Description
_links	_links	
name	string	
uuid	string	

aggregate_spare

Name	Type	Description
checksum_style	string	The checksum type that has been assigned to the spares.
disk_class	string	Disk class of spares.
disk_type	string	Type of disk.
is_partition	boolean	Indicates whether a disk is partitioned (true) or whole (false)
layout_requirements	array[layout_requirement]	Available RAID protections and their restrictions.

Name	Type	Description
node	node	Node where the spares are assigned.
size	integer	Usable size of each spare, in bytes.
syncmirror_pool	string	SyncMirror spare pool.
total	integer	Total number of spares in the bucket. The total spare count for each class of spares also includes reserved spare capacity recommended by ONTAP best practices. • example: 10 • readOnly: 1 • Introduced in: 9.11 • x-nullable: true
usable	integer	Total number of usable spares in the bucket. The usable count for each class of spares does not include reserved spare capacity recommended by ONTAP best practices. • example: 9 • readOnly: 1 • Introduced in: 9.6 • x-nullable: true

snapshot

Name	Type	Description
files_total	integer	Total files allowed in Snapshot copies
files_used	integer	Total files created in Snapshot copies
max_files_available	integer	Maximum files available for Snapshot copies

Name	Type	Description
max_files_used	integer	Files in use by Snapshot copies

block_storage

Name	Type	Description
aggregate_metadata	integer	Space used by different metafiles and internal operations inside the aggregate, in bytes.
aggregate_metadata_percent	integer	Aggregate metadata as a percentage.
available	integer	Space available in bytes.
data_compacted_count	integer	Amount of compacted data in bytes.
data_compaction_space_saved	integer	Space saved in bytes by compacting the data.
data_compaction_space_saved_percent	integer	Percentage saved by compacting the data.
full_threshold_percent	integer	The aggregate used percentage at which 'monitor.volume.full' EMS is generated.
inactive_user_data	integer	The size that is physically used in the block storage and has a cold temperature, in bytes. This property is only supported if the aggregate is either attached to a cloud store or can be attached to a cloud store. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either block_storage.inactive_user_data or **.

Name	Type	Description
inactive_user_data_percent	integer	The percentage of inactive user data in the block storage. This property is only supported if the aggregate is either attached to a cloud store or can be attached to a cloud store. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either block_storage.inactive_user_data_percent or **.
physical_used	integer	Total physical used size of an aggregate in bytes.
physical_used_percent	integer	Physical used percentage.
size	integer	Total usable space in bytes, not including WAFL reserve and aggregate Snapshot copy reserve.
used	integer	Space used or reserved in bytes. Includes volume guarantees and aggregate metadata.
used_including_snapshot_reserve	integer	Total used including the Snapshot copy reserve, in bytes.
used_including_snapshot_reserve_percent	integer	Total used including the Snapshot reserve as a percentage.
used_percent	integer	Aggregate used percentage.
volume_deduplication_shared_count	integer	Amount of shared bytes counted by storage efficiency.
volume_deduplication_space_saved	integer	Amount of space saved in bytes by storage efficiency.
volume_deduplication_space_saved_percent	integer	Percentage of space saved by storage efficiency.

Name	Type	Description
volume_footprints_percent	integer	A summation of volume footprints inside the aggregate, as a percentage. A volume's footprint is the amount of space being used for the volume in the aggregate.

cloud_storage

Name	Type	Description
used	integer	Used space in bytes in the cloud store. Only applicable for aggregates with a cloud store tier.

efficiency

Storage efficiency.

Name	Type	Description
auto_adaptive_compression_savings	boolean	Indicates whether or not aggregate has auto adaptive compression savings.
cross_volume_background_dedupe	boolean	Indicates whether or not cross volume background deduplication is enabled.
cross_volume_dedupe_savings	boolean	Indicates whether or not aggregate has cross volume deduplication savings.
cross_volume_inline_dedupe	boolean	Indicates whether or not cross volume inline deduplication is enabled.
logical_used	integer	Logical used
ratio	number	Data reduction ratio (logical_used / used)
savings	integer	Space saved by storage efficiencies (logical_used - used)

efficiency_without_snapshots

Storage efficiency that does not include the savings provided by Snapshot copies.

Name	Type	Description
logical_used	integer	Logical used
ratio	number	Data reduction ratio (logical_used / used)
savings	integer	Space saved by storage efficiencies (logical_used - used)

efficiency_without_snapshots_flexclones

Storage efficiency that does not include the savings provided by Snapshot copies and Flexclone volumes.

Name	Type	Description
logical_used	integer	Logical used
ratio	number	Data reduction ratio (logical_used / used)
savings	integer	Space saved by storage efficiencies (logical_used - used)

snapshot

Name	Type	Description
available	integer	Available space for Snapshot copies in bytes
reserve_percent	integer	Percentage of space reserved for Snapshot copies
total	integer	Total space for Snapshot copies in bytes
used	integer	Space used by Snapshot copies in bytes
used_percent	integer	Percentage of disk space used by Snapshot copies

space

Name	Type	Description
block_storage	block_storage	

Name	Type	Description
cloud_storage	cloud_storage	
efficiency	efficiency	Storage efficiency.
efficiency_without_snapshots	efficiency_without_snapshots	Storage efficiency that does not include the savings provided by Snapshot copies.
efficiency_without_snapshots_flex clones	efficiency_without_snapshots_flex clones	Storage efficiency that does not include the savings provided by Snapshot copies and Flexclone volumes.
footprint	integer	A summation of volume footprints (including volume guarantees), in bytes. This includes all of the volume footprints in the block_storage tier and the cloud_storage tier. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.
snapshot	snapshot	

iops_raw

The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.

Name	Type	Description
write	integer	Peformance metric for write I/O operations.

latency_raw

The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

throughput_raw

Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Peformance metric for write I/O operations.

statistics

The real time I/O statistics for the aggregate.

Name	Type	Description
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

aggregate

Name	Type	Description
_links	_links	
_tags	array[string]	Tags are an optional way to track the uses of a resource. Tag values must be formatted as key:value strings.
block_storage	block_storage	Configuration information for the locally attached portion of the aggregate. When a cloud store is also used by this aggregate, this is referred to as the performance tier.
cloud_storage	cloud_storage	Configuration information for the cloud storage portion of the aggregate. This is referred to as the capacity tier.
create_time	string	Timestamp of aggregate creation.
data_encryption	data_encryption	
dr_home_node	dr_home_node	Node where the aggregate resides after disaster recovery. The value for this field might differ from the 'node' field during switchover.
home_node	home_node	Node where the aggregate resides after giveback. The value for this field might differ from the value of the 'node' field during takeover.
inactive_data_reporting	inactive_data_reporting	
is_spare_low	boolean	Specifies whether the aggregate is in a spares low condition on any of the RAID groups. This is an advanced property; there is an added computational cost to retrieving its value. The field is not populated for either a collection GET or an instance GET unless it is explicitly requested using the *fields* query parameter containing either footprint or **.

Name	Type	Description
name	string	Aggregate name.
node	node	Node where the aggregate currently resides.
recommendation_spares	array[aggregate_spare]	Information on the aggregate's remaining hot spare disks.
sidl_enabled	boolean	Specifies whether or not SIDL is enabled on the aggregate.
snapshot	snapshot	
space	space	
state	string	Operational state of the aggregate.
uuid	string	Aggregate UUID.
volume-count	integer	Number of volumes in the aggregate.

job_link

Name	Type	Description
uuid	string	The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation.

action

Name	Type	Description
arguments	array[string]	Arguments present in the specified action message.
code	integer	Corrective action code of the specified action.
message	string	Specifies the corrective action to be taken to resolve the issue.

warning

Name	Type	Description
arguments	array[string]	Arguments present in the warning message encountered.
code	integer	Warning code of the warning encountered.
message	string	Details of the warning encountered by the aggregate simulate query.

aggregate_warning

Name	Type	Description
action	action	
name	string	Name of the entity that returns the warning.
warning	warning	

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.