

# Manage application consistency groups

**ONTAP 9.10.1 REST API Documentation** 

NetApp May 08, 2024

This PDF was generated from https://docs.netapp.com/us-en/ontap-restapi-9101/ontap/application\_consistency-groups\_endpoint\_overview.html on May 08, 2024. Always check docs.netapp.com for the latest.

# **Table of Contents**

Manage application consistency groups	1
Application consistency-groups endpoint overview	1
Retrieve details of a collection or consistency group	18
Create a consistency group	65
Delete a consistency group	. 109
Retrieve a consistency group	. 112
Update a consistency group.	. 160

# Manage application consistency groups

# Application consistency-groups endpoint overview

### Overview

A consistency group is a group of volumes that supports capabilities such as creating a snapshot of all of its member volumes at the same point-in-time with a write-fence, thus ensuring a consistent image of the volumes at that time.

Applications with datasets scoped to a single volume can have its contents saved to a Snapshot copy, replicated, or cloned in a crash-consistent manner implicitly with corresponding native ONTAP volume-granular operations. Applications with datasets spanning a group of multiple volumes must have such operations performed on the group. Typically, by first fencing writes to all the volumes in the group, flushing any writes pending in queues, executing the intended operation, that is, take Snapshot copy of every volume in the group and when that is complete, unfence and resume writes. A consistency group is the conventional mechanism for providing such group semantics.

### **Consistency group APIs**

The following APIs are used to perform operations related to consistency groups:

– GET /api/application/consistency-groups

– POST /api/application/consistency-groups

– GET /api/application/consistency-groups/{uuid}

– PATCH /api/application/consistency-groups/{uuid}

– DELETE /api/application/consistency-groups/{uuid}

### Examples

#### Retrieving all consistency groups of an SVM

```
# The API:
/api/application/consistency-groups
# The call:
curl -X GET "https://netapp-
cluster.netapp.com/api/application/consistency-groups?svm.name=vs1" -H
"accept: application/hal+json"
# The response:
{
    "records": [
    {
        "uuid": "6f48d798-0a7f-11ec-a449-005056bbcf9f",
```

```
"name": "vol1",
    " links": {
     "self": {
        "href": "/api/application/consistency-groups/6f48d798-0a7f-11ec-
a449-005056bbcf9f"
     }
   }
  },
  {
    "uuid": "c1b22c85-0a82-11ec-a449-005056bbcf9f",
    "name": "parent cg",
    " links": {
     "self": {
        "href": "/api/application/consistency-groups/c1b22c85-0a82-11ec-
a449-005056bbcf9f"
      }
   }
  },
  {
    "uuid": "c1b270b1-0a82-11ec-a449-005056bbcf9f",
    "name": "child 1",
    " links": {
      "self": {
        "href": "/api/application/consistency-groups/c1b270b1-0a82-11ec-
a449-005056bbcf9f"
     }
   }
  },
  {
    "uuid": "c1b270c3-0a82-11ec-a449-005056bbcf9f",
    "name": "child 2",
    " links": {
     "self": {
        "href": "/api/application/consistency-groups/c1b270c3-0a82-11ec-
a449-005056bbcf9f"
     }
   }
 }
],
"num records": 4,
" links": {
 "self": {
    "href": "/api/application/consistency-groups"
 }
}
}
```

#### Retrieving details of all consistency groups of an SVM

Retrieving details of the consistency groups for a specified SVM. These details are considered to be performant and will return within 1 second when 40 records or less are requested.

```
curl -X GET -k -u admin:netapp1! "https://netapp-
cluster.netapp.com/api/application/consistency-
groups?svm.name=vs1&fields=*&max records=40"
#### Response:
{
"records": [
  {
    "uuid": "6f48d798-0a7f-11ec-a449-005056bbcf9f",
    "name": "vol1",
    "svm": {
      "uuid": "4853f97a-0a63-11ec-a449-005056bbcf9f",
      "name": "vs1",
      " links": {
        "self": {
          "href": "/api/svm/svms/4853f97a-0a63-11ec-a449-005056bbcf9f"
        }
      }
    },
    "space": {
      "size": 108003328,
      "available": 107704320,
      "used": 299008
    },
    "replicated": false,
    " links": {
      "self": {
        "href": "/api/application/consistency-groups/6f48d798-0a7f-11ec-
a449-005056bbcf9f"
      }
    }
  },
  {
    "uuid": "c1b22c85-0a82-11ec-a449-005056bbcf9f",
    "name": "parent cg",
    "svm": {
      "uuid": "4853f97a-0a63-11ec-a449-005056bbcf9f",
      "name": "vs1",
      " links": {
        "self": {
          "href": "/api/svm/svms/4853f97a-0a63-11ec-a449-005056bbcf9f"
        }
```

```
}
    },
    "snapshot policy": {
      "name": "default-1weekly",
      "uuid": "a30bd0fe-067d-11ec-a449-005056bbcf9f",
      " links": {
        "self": {
          "href": "/api/storage/snapshot-policies/a30bd0fe-067d-11ec-a449-
005056bbcf9f"
       }
      }
    },
    "consistency groups": [
      {
        "uuid": "c1b270b1-0a82-11ec-a449-005056bbcf9f",
        "name": "child 1",
        "space": {
          "size": 41943040,
          "available": 39346176,
          "used": 499712
        },
        " links": {
          "self": {
            "href": "/api/application/consistency-groups/c1b270b1-0a82-
11ec-a449-005056bbcf9f"
          }
        }
      },
      {
        "uuid": "c1b270c3-0a82-11ec-a449-005056bbcf9f",
        "name": "child 2",
        "space": {
          "size": 41943040,
          "available": 39350272,
          "used": 495616
        },
        " links": {
          "self": {
            "href": "/api/application/consistency-groups/c1b270c3-0a82-
11ec-a449-005056bbcf9f"
          }
        }
      }
    ],
    "space": {
      "size": 83886080,
```

```
"available": 78696448,
      "used": 995328
    },
    "replicated": false,
    " links": {
      "self": {
        "href": "/api/application/consistency-groups/c1b22c85-0a82-11ec-
a449-005056bbcf9f"
      }
    }
  },
  {
    "uuid": "c1b270b1-0a82-11ec-a449-005056bbcf9f",
    "name": "child 1",
    "parent consistency group": {
      "uuid": "c1b22c85-0a82-11ec-a449-005056bbcf9f",
      "name": "parent cg",
      " links": {
        "self": {
          "href": "/api/application/consistency-groups/c1b22c85-0a82-11ec-
a449-005056bbcf9f"
        }
      }
    },
    "svm": {
      "uuid": "4853f97a-0a63-11ec-a449-005056bbcf9f",
      "name": "vs1",
      " links": {
        "self": {
          "href": "/api/svm/svms/4853f97a-0a63-11ec-a449-005056bbcf9f"
        }
      }
    },
    "snapshot_policy": {
      "name": "default",
      "uuid": "a30b60a4-067d-11ec-a449-005056bbcf9f",
      " links": {
        "self": {
          "href": "/api/storage/snapshot-policies/a30b60a4-067d-11ec-a449-
005056bbcf9f"
        }
      }
    },
    "space": {
      "size": 41943040,
      "available": 39346176,
```

```
"used": 499712
    },
    " links": {
      "self": {
        "href": "/api/application/consistency-groups/c1b270b1-0a82-11ec-
a449-005056bbcf9f"
      }
    }
  },
  {
    "uuid": "c1b270c3-0a82-11ec-a449-005056bbcf9f",
    "name": "child 2",
    "parent consistency group": {
      "uuid": "c1b22c85-0a82-11ec-a449-005056bbcf9f",
      "name": "parent cg",
      " links": {
       "self": {
          "href": "/api/application/consistency-groups/c1b22c85-0a82-11ec-
a449-005056bbcf9f"
       }
      }
    },
    "svm": {
      "uuid": "4853f97a-0a63-11ec-a449-005056bbcf9f",
      "name": "vs1",
      " links": {
       "self": {
          "href": "/api/svm/svms/4853f97a-0a63-11ec-a449-005056bbcf9f"
        }
      }
    },
    "snapshot policy": {
      "name": "default",
      "uuid": "a30b60a4-067d-11ec-a449-005056bbcf9f",
      " links": {
        "self": {
          "href": "/api/storage/snapshot-policies/a30b60a4-067d-11ec-a449-
005056bbcf9f"
       }
      }
    },
    "space": {
      "size": 41943040,
      "available": 39350272,
      "used": 495616
    },
```

```
" links": {
      "self": {
        "href": "/api/application/consistency-groups/c1b270c3-0a82-11ec-
a449-005056bbcf9f"
      }
    }
  }
],
"num_records": 4,
" links": {
  "self": {
    "href": "/api/application/consistency-
groups?svm.name=vs1&fields=*&max records=40"
  }
}
}
```

#### Retrieving details of non-nested consistency groups

Retrieves details of the consistency groups without nested consistency groups, or only the parent consistency group for a number of consistency groups of a specified SVM.

```
curl -X GET -k -u admin:netapp1! "https://netapp-
cluster.netapp.com/api/application/consistency-
groups?svm.name=vs1&parent consistency group.uuid=null"
#### Response:
{
"records": [
  {
    "uuid": "6f48d798-0a7f-11ec-a449-005056bbcf9f",
    "name": "vol1",
    "svm": {
      "name": "vs1"
    },
    " links": {
      "self": {
        "href": "/api/application/consistency-groups/6f48d798-0a7f-11ec-
a449-005056bbcf9f"
      }
    }
  },
  {
    "uuid": "c1b22c85-0a82-11ec-a449-005056bbcf9f",
    "name": "parent cg",
    "svm": {
      "name": "vs1"
    },
    " links": {
      "self": {
        "href": "/api/application/consistency-groups/c1b22c85-0a82-11ec-
a449-005056bbcf9f"
     }
    }
 }
],
"num records": 2,
" links": {
 "self": {
    "href": "/api/application/consistency-
groups?svm.name=vs1&parent consistency group.uuid=null"
 }
}
}
```

#### Creating a single consistency group with a new SAN volume

Provisions an application with one consistency group, each with one new SAN volumes, with one LUN, an igroup and no explicit Snapshot copy policy, FabricPool tiering policy, storage service, and QoS policy specification. The igroup to map a LUN to is specified at LUN-granularity.

```
curl -X POST -k -u admin:netapp1! https://netapp-
cluster.netapp.com/api/application/consistency-groups?return records=true
-d '{ "svm": { "name": "vs1" }, "luns": [ { "name": "/vol/vol1/lun1",
"space": { "size": "100mb" }, "os type": "linux", "lun maps": [ {
"igroup": { "name": "igroup1", "initiators": [ { "name": "ign.2021-
07.com.netapp.englab.gdl:scspr2429998001" } ] } ] } ] }'
#### Response:
{
"num records": 1,
"records": [
  {
    "uuid": "6f48d798-0a7f-11ec-a449-005056bbcf9f",
    "name": "vol1",
    "svm": {
    "uuid": "4853f97a-0a63-11ec-a449-005056bbcf9f",
    "name": "vs1",
    " links": {
     "self": {
        "href": "/api/svm/svms/4853f97a-0a63-11ec-a449-005056bbcf9f"
     }
    }
    },
    "luns": [
    {
      "lun maps": [
        {
          "igroup": {
            "name": "igroup1",
            "initiators": [
              {
                "name": "iqn.2021-
07.com.netapp.englab.gdl:scspr2429998001"
              }
            ]
          }
        }
      ],
      "name": "/vol/vol1/lun1",
      "os type": "linux",
      "space": {
```

```
"size": 104857600
}
}
],
],
"job": {
"uuid": "6f4907ae-0a7f-11ec-a449-005056bbcf9f",
"_links": {
    "self": {
    "href": "/api/cluster/jobs/6f4907ae-0a7f-11ec-a449-005056bbcf9f"
    }
}
```

#### Creating an Application with two consistency groups with existing SAN volumes

Provisions an application with two consistency groups, each with two existing SAN volumes, a Snapshot copy policy at application-granularity, and a distinct consistency group granular Snapshot copy policy.

```
curl -X POST -k -u admin:netapp1! https://netapp-
cluster.netapp.com/api/application/consistency-groups?return records=true
-d '{ "svm": { "name": "vs1" }, "name": "parent cg", "snapshot policy": {
"name": "default-1weekly" }, "consistency groups": [ { "name": "child 1",
"snapshot policy": { "name": "default" }, "volumes": [ { "name":
"existing vol1", "provisioning options": { "action": "add" } }, { "name":
"existing vol2", "provisioning options": { "action": "add" } } ] }, {
"name": "child 2", "snapshot policy": { "name": "default" }, "volumes": [
{ "name": "existing vol3", "provisioning_options": { "action": "add" } },
{ "name": "existing vol4", "provisioning options": { "action": "add" } } ]
} ] }'
#### Response:
{
"num records": 1,
"records": [
  {
    "uuid": "c1b22c85-0a82-11ec-a449-005056bbcf9f",
    "name": "parent cg",
    "svm": {
      "uuid": "4853f97a-0a63-11ec-a449-005056bbcf9f",
      "name": "vs1",
      " links": {
        "self": {
          "href": "/api/svm/svms/4853f97a-0a63-11ec-a449-005056bbcf9f"
```

```
}
      }
    },
    "snapshot policy": {
      "name": "default-1weekly"
    },
    "consistency groups": [
      {
        "uuid": "c1b270b1-0a82-11ec-a449-005056bbcf9f",
        "name": "child 1",
        "snapshot policy": {
          "name": "default"
        },
        "volumes": [
          {
            "name": "existing vol1"
          },
          {
            "name": "existing_vol2"
          }
        1
      },
      {
        "uuid": "c1b270c3-0a82-11ec-a449-005056bbcf9f",
        "name": "child 2",
        "snapshot_policy": {
          "name": "default"
        },
        "volumes": [
          {
            "name": "existing vol3"
          },
          {
            "name": "existing_vol4"
          }
        1
      }
    ]
  }
],
"job": {
  "uuid": "c1b272b9-0a82-11ec-a449-005056bbcf9f",
  " links": {
    "self": {
      "href": "/api/cluster/jobs/c1b272b9-0a82-11ec-a449-005056bbcf9f"
    }
```

} }

#### Retrieving specific details of an existing consistency group

Retrieves the details of an existing consistency group.

```
curl -X GET -k -u admin:netapp1! https://netapp-
cluster.netapp.com/api/application/consistency-groups/6f48d798-0a7f-11ec-
a449-005056bbcf9f
#### Response:
{
"uuid": "6f48d798-0a7f-11ec-a449-005056bbcf9f",
"name": "vol1",
"svm": {
  "uuid": "4853f97a-0a63-11ec-a449-005056bbcf9f",
  "name": "vs1",
  " links": {
    "self": {
      "href": "/api/svm/svms/4853f97a-0a63-11ec-a449-005056bbcf9f"
    }
  }
},
"space": {
 "size": 108003328,
  "available": 107724800,
  "used": 278528
},
"replicated": false,
" links": {
 "self": {
    "href": "/api/application/consistency-groups/6f48d798-0a7f-11ec-a449-
005056bbcf9f"
  }
}
}
```

#### Retrieving all details of an existing consistency group

Retrieves all details of an existing consistency group. These details are not considered to be performant and are not guaranteed to return within one second.

curl -X GET -k -u admin:netapp1! https://netapp-

```
cluster.netapp.com/api/application/consistency-groups/6f48d798-0a7f-11ec-
a449-005056bbcf9f?fields=**
#### Response:
{
"uuid": "6f48d798-0a7f-11ec-a449-005056bbcf9f",
"name": "vol1",
"svm": {
  "uuid": "4853f97a-0a63-11ec-a449-005056bbcf9f",
 "name": "vs1",
 " links": {
    "self": {
      "href": "/api/svm/svms/4853f97a-0a63-11ec-a449-005056bbcf9f"
   }
 }
},
"qos": {
 "policy": {
    "uuid": "b7189398-e572-48ab-8f69-82cd46580812",
    "name": "extreme-fixed",
    " links": {
      "self": {
        "href": "/api/storage/gos/policies/b7189398-e572-48ab-8f69-
82cd46580812"
     }
   }
 }
},
"tiering": {
 "policy": "none"
},
"create time": "2021-08-31T13:18:24-04:00",
"volumes": [
  {
    "uuid": "6f516c6c-0a7f-11ec-a449-005056bbcf9f",
    "qos": {
      "policy": {
        "uuid": "b7189398-e572-48ab-8f69-82cd46580812",
        "name": "extreme-fixed",
        " links": {
          "self": {
            "href": "/api/storage/gos/policies/b7189398-e572-48ab-8f69-
82cd46580812"
          }
        }
      }
```

```
},
    "tiering": {
     "policy": "none"
    },
    "comment": "",
    "create time": "2021-08-31T13:18:22-04:00",
    "name": "vol1",
    "snapshot policy": {
      "name": "default",
      "uuid": "a30b60a4-067d-11ec-a449-005056bbcf9f"
    },
    "space": {
      "size": 108003328,
      "available": 107569152,
      "used": 434176,
      "snapshot": {
        "used": 151552,
        "reserve percent": 0,
        "autodelete enabled": false
     }
    },
    "activity tracking": {
      "supported": false,
      "unsupported reason": {
        "message": "Volume activity tracking is not supported on volumes
that contain LUNs.",
        "code": "124518405"
      },
     "state": "off"
    },
    " links": {
      "self": {
        "href": "/api/storage/volumes/6f516c6c-0a7f-11ec-a449-
005056bbcf9f"
     }
    }
 }
],
"luns": [
  {
    "uuid": "6f51748a-0a7f-11ec-a449-005056bbcf9f",
    "location": {
      "logical unit": "lun1",
      "node": {
        "name": "johnhil-vsim1",
        "uuid": "6eb682f2-067d-11ec-a449-005056bbcf9f",
```

```
" links": {
          "self": {
            "href": "/api/cluster/nodes/6eb682f2-067d-11ec-a449-
005056bbcf9f"
          }
        }
      },
      "volume": {
        "uuid": "6f516c6c-0a7f-11ec-a449-005056bbcf9f",
        "name": "vol1",
        " links": {
          "self": {
            "href": "/api/storage/volumes/6f516c6c-0a7f-11ec-a449-
005056bbcf9f"
          }
        }
      }
    },
    "lun maps": [
      {
        "igroup": {
          "uuid": "6f4a4b86-0a7f-11ec-a449-005056bbcf9f",
          "name": "igroup1",
          "os type": "linux",
          "protocol": "mixed",
          "initiators": [
            {
              "name": "iqn.2021-07.com.netapp.englab.gdl:scspr2429998001"
            }
          ],
          " links": {
            "self": {
              "href": "/api/protocols/san/igroups/6f4a4b86-0a7f-11ec-a449-
005056bbcf9f"
           }
          }
        },
        "logical unit number": 0
      }
    ],
    "name": "/vol/vol1/lun1",
    "auto delete": false,
    "class": "regular",
    "create time": "2021-08-31T13:18:24-04:00",
    "os type": "linux",
    "serial number": "wIqM6]RfQK3t",
```

```
"space": {
      "size": 104857600,
      "used": 0,
      "guarantee": {
        "requested": false,
        "reserved": false
      }
    },
    "status": {
      "container state": "online",
      "mapped": true,
      "read only": false,
      "state": "online"
    },
    " links": {
      "self": {
        "href": "/api/storage/luns/6f51748a-0a7f-11ec-a449-005056bbcf9f"
      }
    }
  }
],
"space": {
  "size": 108003328,
  "available": 107569152,
  "used": 434176
},
"replicated": false,
" links": {
  "self": {
    "href": "/api/application/consistency-groups/6f48d798-0a7f-11ec-a449-
005056bbcf9f?fields=**"
  }
}
}
```

#### Adding LUNs to an existing volume in an existing consistency group

Adds two NVMe namespaces to an existing volume in an existing consistency group, creates a new subsystem, and binds the new namespaces to it.

```
curl -X PATCH -k -u admin:netapp1! 'https://netapp-
cluster.netapp.com/api/application/consistency-groups/6f48d798-0a7f-11ec-
a449-005056bbcf9f' -d '{ "luns": [ { "name": "/vol/vol1/new luns",
"provisioning options": { "count": 2, "action": "create" }, "space": {
"size": "100mb" }, "os type": "linux", "lun maps": [ { "igroup": { "name":
"igroup2", "initiators": [ { "name": "01:02:03:04:05:06:07:01" } ] } } ]
1 }'
#### Response:
{
"job": {
  "uuid": "5306ea44-0a87-11ec-a449-005056bbcf9f",
  " links": {
    "self": {
      "href": "/api/cluster/jobs/5306ea44-0a87-11ec-a449-005056bbcf9f"
    }
  }
}
}
```

#### Restoring a consistency group to the contents of an existing snapshot

Restores an existing consistency group to the contents of an existing snapshot of the consistency group.

```
curl -X PATCH -k -u admin:netapp1! 'https://netapp-
cluster.netapp.com/api/application/consistency-groups/6f51748a-0a7f-11ec-
a449-005056bbcf9f' -d '{ "restore_to": { "snapshot": { "uuid": "92c6c770-
17a1-11eb-b141-005056acd498"} } }' -H "Accept: Application/hal+json"
##### Response:
{
    "job": {
    "uuid": "8907bd9e-1463-11eb-a719-005056ac70af",
    "_links": {
        "self": {
            "href": "/api/cluster/jobs/8907bd9e-1463-11eb-a719-005056ac70af"
        }
    }
}
```

#### Deleting a consistency group

Deletes a consistency group, where all storage originally associated with that consistency group remains in place.

```
curl -X DELETE -k -u admin:netapp1! 'https://netapp-
cluster.netapp.com/api/application/consistency-groups/6f48d798-0a7f-11ec-
a449-005056bbcf9f'
##### Response:
{
}
```

## Retrieve details of a collection or consistency group

**GET** /application/consistency-groups

#### Introduced In: 9.10

Retrieve details of a collection or a specific consistency group.

#### **Notes**

When volume granular properties, such as, the storage SLC, Fabric Pool tiering are not the same for all the existing volumes of a consistency group, the corresponding property is not reported at consistency group granularity. It is only reported if all the volumes of the consistency group have the same value for that property.

If this consistency group instance is part of a replication relationship, the "replicated" parameter will be true. Otherwise, it is false. Also, the "replicated" parameter will not be present in the output for Nested-consistency groups, it is included only for single and top-level consistency groups. If this consistency group instance is the source of a replication relationship, the "replication\_source" parameter will be true. Otherwise, it is false.

#### **Expensive properties**

There is an added 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 DOC Requesting specific fields to learn more.

- volumes
- luns
- namespaces

#### **Parameters**

Name	Туре	In	Required	Description
luns.space.size	integer	query	False	Filter by luns.space.size
luns.space.used	integer	query	False	Filter by luns.space.used

Name	Туре	In	Required	Description
luns.enabled	boolean	query	False	Filter by luns.enabled
luns.name	string	query	False	Filter by luns.name
luns.lun_maps.logica I_unit_number	integer	query	False	Filter by luns.lun_maps.logic al_unit_number
luns.lun_maps.igrou p.uuid	string	query	False	Filter by luns.lun_maps.igrou p.uuid
luns.lun_maps.igrou p.os_type	string	query	False	Filter by luns.lun_maps.igrou p.os_type
luns.lun_maps.igrou p.igroups.uuid	string	query	False	Filter by luns.lun_maps.igrou p.igroups.uuid
luns.lun_maps.igrou p.igroups.name	string	query	False	Filter by luns.lun_maps.igrou p.igroups.name
luns.lun_maps.igrou p.name	string	query	False	Filter by luns.lun_maps.igrou p.name
luns.lun_maps.igrou p.protocol	string	query	False	Filter by luns.lun_maps.igrou p.protocol
luns.lun_maps.igrou p.initiators.comment	string	query	False	Filter by luns.lun_maps.igrou p.initiators.comment
luns.lun_maps.igrou p.initiators.name	string	query	False	Filter by luns.lun_maps.igrou p.initiators.name
luns.create_time	string	query	False	Filter by luns.create_time
luns.os_type	string	query	False	Filter by luns.os_type

Name	Туре	In	Required	Description
luns.qos.policy.uuid	string	query	False	Filter by luns.qos.policy.uuid
luns.qos.policy.min_t hroughput_iops	integer	query	False	Filter by luns.qos.policy.min_ throughput_iops
luns.qos.policy.max_ throughput_iops	integer	query	False	Filter by luns.qos.policy.max _throughput_iops
luns.qos.policy.min_t hroughput_mbps	integer	query	False	Filter by luns.qos.policy.min_ throughput_mbps
luns.qos.policy.name	string	query	False	Filter by luns.qos.policy.nam e
luns.qos.policy.max_ throughput_mbps	integer	query	False	Filter by luns.qos.policy.max _throughput_mbps
luns.uuid	string	query	False	Filter by luns.uuid
luns.serial_number	string	query	False	Filter by luns.serial_number
luns.comment	string	query	False	Filter by luns.comment
replicated	boolean	query	False	Filter by replicated
qos.policy.min_throu ghput_mbps	integer	query	False	Filter by qos.policy.min_throu ghput_mbps
qos.policy.name	string	query	False	Filter by qos.policy.name
qos.policy.max_thro ughput_mbps	integer	query	False	Filter by qos.policy.max_thro ughput_mbps
qos.policy.uuid	string	query	False	Filter by qos.policy.uuid

Name	Туре	In	Required	Description
qos.policy.min_throu ghput_iops	integer	query	False	Filter by qos.policy.min_throu ghput_iops
qos.policy.max_thro ughput_iops	integer	query	False	Filter by qos.policy.max_thro ughput_iops
svm.uuid	string	query	False	Filter by svm.uuid
svm.name	string	query	False	Filter by svm.name
space.used	integer	query	False	Filter by space.used
space.available	integer	query	False	Filter by space.available
space.size	integer	query	False	Filter by space.size
tiering.policy	string	query	False	Filter by tiering.policy
uuid	string	query	False	Filter by uuid
parent_consistency_ group.uuid	string	query	False	Filter by parent_consistency_ group.uuid
parent_consistency_ group.name	string	query	False	Filter by parent_consistency_ group.name
snapshot_policy.uuid	string	query	False	Filter by snapshot_policy.uui d
snapshot_policy.na me	string	query	False	Filter by snapshot_policy.na me
name	string	query	False	Filter by name
consistency_groups. qos.policy.min_throu ghput_mbps	integer	query	False	Filter by consistency_groups. qos.policy.min_throu ghput_mbps

Name	Туре	In	Required	Description
consistency_groups. qos.policy.name	string	query	False	Filter by consistency_groups. qos.policy.name
consistency_groups. qos.policy.max_thro ughput_mbps	integer	query	False	Filter by consistency_groups. qos.policy.max_thro ughput_mbps
consistency_groups. qos.policy.uuid	string	query	False	Filter by consistency_groups. qos.policy.uuid
consistency_groups. qos.policy.min_throu ghput_iops	integer	query	False	Filter by consistency_groups. qos.policy.min_throu ghput_iops
consistency_groups. qos.policy.max_thro ughput_iops	integer	query	False	Filter by consistency_groups. qos.policy.max_thro ughput_iops
consistency_groups. uuid	string	query	False	Filter by consistency_groups. uuid
consistency_groups. parent_consistency_ group.uuid	string	query	False	Filter by consistency_groups. parent_consistency_ group.uuid
consistency_groups. parent_consistency_ group.name	string	query	False	Filter by consistency_groups. parent_consistency_ group.name
consistency_groups. snapshot_policy.uuid	string	query	False	Filter by consistency_groups. snapshot_policy.uui d
consistency_groups. snapshot_policy.na me	string	query	False	Filter by consistency_groups. snapshot_policy.na me

Name	Туре	In	Required	Description
consistency_groups.l uns.space.size	integer	query	False	Filter by consistency_groups. luns.space.size
consistency_groups.l uns.space.used	integer	query	False	Filter by consistency_groups. luns.space.used
consistency_groups.l uns.enabled	boolean	query	False	Filter by consistency_groups. luns.enabled
consistency_groups.l uns.name	string	query	False	Filter by consistency_groups. luns.name
consistency_groups.l uns.lun_maps.logical _unit_number	integer	query	False	Filter by consistency_groups. luns.lun_maps.logic al_unit_number
consistency_groups.l uns.lun_maps.igroup .uuid	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.uuid
consistency_groups.l uns.lun_maps.igroup .os_type	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.os_type
consistency_groups.l uns.lun_maps.igroup .igroups.uuid	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.igroups.uuid
consistency_groups.l uns.lun_maps.igroup .igroups.name	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.igroups.name
consistency_groups.l uns.lun_maps.igroup .name	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.name

Name	Туре	In	Required	Description
consistency_groups.l uns.lun_maps.igroup .protocol	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.protocol
consistency_groups.l uns.lun_maps.igroup .initiators.comment	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.initiators.comment
consistency_groups.l uns.lun_maps.igroup .initiators.name	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.initiators.name
consistency_groups.l uns.create_time	string	query	False	Filter by consistency_groups. luns.create_time
consistency_groups.l uns.os_type	string	query	False	Filter by consistency_groups. luns.os_type
consistency_groups.l uns.qos.policy.uuid	string	query	False	Filter by consistency_groups. luns.qos.policy.uuid
consistency_groups.l uns.qos.policy.min_t hroughput_iops	integer	query	False	Filter by consistency_groups. luns.qos.policy.min_ throughput_iops
consistency_groups.l uns.qos.policy.max_t hroughput_iops	integer	query	False	Filter by consistency_groups. luns.qos.policy.max _throughput_iops
consistency_groups.l uns.qos.policy.min_t hroughput_mbps	integer	query	False	Filter by consistency_groups. luns.qos.policy.min_ throughput_mbps
consistency_groups.l uns.qos.policy.name	string	query	False	Filter by consistency_groups. luns.qos.policy.nam e

Name	Туре	In	Required	Description
consistency_groups.l uns.qos.policy.max_t hroughput_mbps	integer	query	False	Filter by consistency_groups. luns.qos.policy.max _throughput_mbps
consistency_groups.l uns.uuid	string	query	False	Filter by consistency_groups. luns.uuid
consistency_groups.l uns.serial_number	string	query	False	Filter by consistency_groups. luns.serial_number
consistency_groups.l uns.comment	string	query	False	Filter by consistency_groups. luns.comment
consistency_groups. tiering.policy	string	query	False	Filter by consistency_groups. tiering.policy
consistency_groups. volumes.uuid	string	query	False	Filter by consistency_groups. volumes.uuid
consistency_groups. volumes.qos.policy. min_throughput_mb ps	integer	query	False	Filter by consistency_groups. volumes.qos.policy. min_throughput_mb ps
consistency_groups. volumes.qos.policy.n ame	string	query	False	Filter by consistency_groups. volumes.qos.policy. name
consistency_groups. volumes.qos.policy. max_throughput_mb ps	integer	query	False	Filter by consistency_groups. volumes.qos.policy. max_throughput_mb ps
consistency_groups. volumes.qos.policy.u uid	string	query	False	Filter by consistency_groups. volumes.qos.policy. uuid

Name	Туре	In	Required	Description
consistency_groups. volumes.qos.policy. min_throughput_iops	integer	query	False	Filter by consistency_groups. volumes.qos.policy. min_throughput_iop s
consistency_groups. volumes.qos.policy. max_throughput_iop s	integer	query	False	Filter by consistency_groups. volumes.qos.policy. max_throughput_iop s
consistency_groups. volumes.snapshot_p olicy.uuid	string	query	False	Filter by consistency_groups. volumes.snapshot_p olicy.uuid
consistency_groups. volumes.snapshot_p olicy.name	string	query	False	Filter by consistency_groups. volumes.snapshot_p olicy.name
consistency_groups. volumes.comment	string	query	False	Filter by consistency_groups. volumes.comment
consistency_groups. volumes.tiering.polic y	string	query	False	Filter by consistency_groups. volumes.tiering.polic y
consistency_groups. volumes.language	string	query	False	Filter by consistency_groups. volumes.language
consistency_groups. volumes.space.size	integer	query	False	Filter by consistency_groups. volumes.space.size
consistency_groups. volumes.space.avail able	integer	query	False	Filter by consistency_groups. volumes.space.avail able
consistency_groups. volumes.space.used	integer	query	False	Filter by consistency_groups. volumes.space.used

Name	Туре	In	Required	Description
consistency_groups. volumes.name	string	query	False	Filter by consistency_groups. volumes.name
consistency_groups. space.used	integer	query	False	Filter by consistency_groups. space.used
consistency_groups. space.size	integer	query	False	Filter by consistency_groups. space.size
consistency_groups. space.available	integer	query	False	Filter by consistency_groups. space.available
consistency_groups. name	string	query	False	Filter by consistency_groups. name
consistency_groups. svm.uuid	string	query	False	Filter by consistency_groups. svm.uuid
consistency_groups. svm.name	string	query	False	Filter by consistency_groups. svm.name
volumes.uuid	string	query	False	Filter by volumes.uuid
volumes.qos.policy. min_throughput_mb ps	integer	query	False	Filter by volumes.qos.policy. min_throughput_mb ps
volumes.qos.policy.n ame	string	query	False	Filter by volumes.qos.policy. name
volumes.qos.policy. max_throughput_mb ps	integer	query	False	Filter by volumes.qos.policy. max_throughput_mb ps

Name	Туре	In	Required	Description
volumes.qos.policy.u uid	string	query	False	Filter by volumes.qos.policy. uuid
volumes.qos.policy. min_throughput_iops	integer	query	False	Filter by volumes.qos.policy. min_throughput_iop s
volumes.qos.policy. max_throughput_iop s	integer	query	False	Filter by volumes.qos.policy. max_throughput_iop s
volumes.snapshot_p olicy.uuid	string	query	False	Filter by volumes.snapshot_p olicy.uuid
volumes.snapshot_p olicy.name	string	query	False	Filter by volumes.snapshot_p olicy.name
volumes.comment	string	query	False	Filter by volumes.comment
volumes.tiering.polic y	string	query	False	Filter by volumes.tiering.polic y
volumes.language	string	query	False	Filter by volumes.language
volumes.space.size	integer	query	False	Filter by volumes.space.size
volumes.space.avail able	integer	query	False	Filter by volumes.space.avail able
volumes.space.used	integer	query	False	Filter by volumes.space.used
volumes.name	string	query	False	Filter by volumes.name
replication_source	boolean	query	False	Filter by replication_source

Name	Туре	In	Required	Description
fields	array[string]	query	False	Specify the fields to return.
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned. • Default value: 1
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When iterating over a collection, the default is 15 seconds. ONTAP returns earlier if either max records or the end of the collection is reached. • Default value: 1 • Max value: 120 • Min value: 0
order_by	array[string]	query	False	Order results by specified fields and optional [asc

## Response

Status: 200, Ok

Name	Туре	Description
_links	collection_links	
num_records	integer	Number of records.
records	array[records]	

**Example response** 

```
{
 " links": {
   "next": {
     "href": "/api/resourcelink"
   },
   "self": {
     "href": "/api/resourcelink"
   }
 },
 "records": {
   " links": {
     "self": {
       "href": "/api/resourcelink"
     }
   },
   "consistency groups": {
     " links": {
       "self": {
         "href": "/api/resourcelink"
       }
     },
     "luns": {
       "clone": {
         "source": {
           "name": "/vol/volume1/lun1",
           "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
         }
        },
        "comment": "string",
        "create time": "2018-06-04T19:00:00Z",
        "lun maps": {
          "igroup": {
            "igroups": {
              " links": {
                "self": {
                  "href": "/api/resourcelink"
                }
              },
              "name": "igroup1",
             "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
            },
            "initiators": {
              "comment": "my comment",
              "name": "iqn.1998-01.com.corp.iscsi:name1"
```

```
},
      "name": "igroup1",
      "os type": "aix",
      "protocol": "fcp",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "name": "/vol/volume1/lun1",
  "os type": "aix",
  "provisioning options": {
    "action": "create"
  },
  "qos": {
    "policy": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "serial number": "string",
  "space": {
    "size": 1073741824,
   "used": 0
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"namespaces": {
  "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "name": "/vol/volume1/qtree1/namespace1",
  "os type": "aix",
  "provisioning options": {
    "action": "create"
  },
  "subsystem map": {
   " links": {
      "self": {
        "href": "/api/resourcelink"
```

```
}
    },
    "anagrpid": "00103050h",
    "nsid": "00000001h",
    "subsystem": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
 },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"parent consistency group": {
  " links": {
   "self": {
     "href": "/api/resourcelink"
   }
 },
  "name": "my consistency group",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"provisioning options": {
 "action": "create",
 "storage service": {
   "name": "extreme"
 }
},
"qos": {
 "policy": {
   " links": {
     "self": {
       "href": "/api/resourcelink"
     }
    },
    "max throughput iops": 10000,
    "max throughput mbps": 500,
    "min throughput iops": 2000,
    "min throughput mbps": 500,
   "name": "performance",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
 }
},
"snapshot policy": {
```

```
" links": {
   "self": {
     "href": "/api/resourcelink"
   }
  },
 "name": "default",
 "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"space": {
 "available": 5737418,
 "size": 1073741824,
 "used": 5737418
},
"svm": {
 " links": {
   "self": {
     "href": "/api/resourcelink"
   }
 },
 "name": "svm1",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"tiering": {
 "control": "allowed",
 "policy": "all"
},
"uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
"volumes": {
 "comment": "string",
 "language": "ar",
 "name": "vol cs dept",
 "provisioning options": {
   "action": "create",
   "storage service": {
     "name": "extreme"
   }
  },
  "qos": {
   "policy": {
      " links": {
        "self": {
        "href": "/api/resourcelink"
        }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
```

```
"min throughput iops": 2000,
        "min throughput mbps": 500,
        "name": "performance",
       "uuid": "lcd8a442-86d1-11e0-ae1c-123478563412"
     }
    },
    "snapshot policy": {
     " links": {
       "self": {
         "href": "/api/resourcelink"
       }
     },
     "name": "default",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "space": {
     "available": 0,
     "used": 0
    },
    "tiering": {
     "control": "allowed",
     "policy": "all"
   },
   "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
 }
},
"luns": {
 "clone": {
   "source": {
     "name": "/vol/volume1/lun1",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
 },
 "comment": "string",
 "create time": "2018-06-04T19:00:00Z",
 "lun maps": {
   "igroup": {
      "igroups": {
        " links": {
          "self": {
           "href": "/api/resourcelink"
         }
        },
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
```
```
"initiators": {
        "comment": "my comment",
        "name": "ign.1998-01.com.corp.iscsi:name1"
      },
      "name": "igroup1",
      "os type": "aix",
      "protocol": "fcp",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "name": "/vol/volume1/lun1",
  "os type": "aix",
  "provisioning options": {
    "action": "create"
  },
  "qos": {
    "policy": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
       }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "serial number": "string",
  "space": {
   "size": 1073741824,
   "used": 0
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"namespaces": {
  "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "name": "/vol/volume1/qtree1/namespace1",
  "os type": "aix",
  "provisioning options": {
   "action": "create"
  },
  "subsystem map": {
```

```
" links": {
      "self": {
       "href": "/api/resourcelink"
     }
    },
    "anagrpid": "00103050h",
    "nsid": "00000001h",
   "subsystem": {
      " links": {
       "self": {
         "href": "/api/resourcelink"
       }
     },
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
 },
 "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"parent consistency group": {
 " links": {
   "self": {
     "href": "/api/resourcelink"
   }
 },
 "name": "my consistency group",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"provisioning options": {
 "action": "create",
 "storage service": {
   "name": "extreme"
 }
},
"qos": {
 "policy": {
    " links": {
     "self": {
       "href": "/api/resourcelink"
     }
   },
    "max throughput iops": 10000,
   "max throughput mbps": 500,
    "min throughput iops": 2000,
    "min throughput mbps": 500,
    "name": "performance",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
```

```
}
},
"snapshot policy": {
 " links": {
   "self": {
     "href": "/api/resourcelink"
   }
 },
 "name": "default",
 "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"space": {
 "available": 5737418,
 "size": 1073741824,
 "used": 5737418
},
"svm": {
 " links": {
   "self": {
     "href": "/api/resourcelink"
   }
 },
 "name": "svm1",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"tiering": {
"control": "allowed",
"policy": "all"
},
"uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
"volumes": {
 "comment": "string",
 "language": "ar",
 "name": "vol cs dept",
 "provisioning options": {
   "action": "create",
   "storage service": {
     "name": "extreme"
   }
 },
  "qos": {
    "policy": {
      " links": {
       "self": {
         "href": "/api/resourcelink"
        }
```

```
},
          "max throughput iops": 10000,
          "max throughput mbps": 500,
          "min throughput iops": 2000,
          "min throughput mbps": 500,
          "name": "performance",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
       }
      },
      "snapshot policy": {
       " links": {
         "self": {
           "href": "/api/resourcelink"
         }
       },
        "name": "default",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "space": {
       "available": 0,
       "used": 0
      },
      "tiering": {
       "control": "allowed",
       "policy": "all"
      },
      "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
   }
 }
}
```

## Error

```
Status: Default, Error
```

Name	Туре	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	Туре	Description
href	string	

collection\_links

Name	Туре	Description
next	href	
self	href	

self\_link

Name	Туре	Description
self	href	

#### source

The source LUN for a LUN clone operation. This can be specified using property clone.source.uuid or clone.source.name. If both properties are supplied, they must refer to the same LUN.

Valid in POST to create a new LUN as a clone of the source.

Valid in PATCH to overwrite an existing LUN's data as a clone of another.

Name	Туре	Description
name	string	The fully qualified path name of the clone source LUN composed of a "/vol" prefix, the volume name, the (optional) qtree name, and base name of the LUN. Valid in POST and PATCH.
uuid	string	The unique identifier of the clone source LUN. Valid in POST and PATCH.

#### clone

This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: auto\_delete,
gos\_policy, space.guarantee.requested and
space.scsi thin provisioning support enabled.

When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following

properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: class, auto\_delete, lun\_maps, serial\_number, status.state, and uuid.

Persistent reservations for the patched LUN are also preserved.

Name	Туре	Description
source	source	The source LUN for a LUN clone operation. This can be specified using property clone.source.uuid Or clone.source.name. If both properties are supplied, they must refer to the same LUN. Valid in POST to create a new LUN as a clone of the source. Valid in PATCH to overwrite an existing LUN's data as a clone of another.

#### igroups

Name	Туре	Description
_links	self_link	
name	string	The name of the initiator group.
uuid	string	The unique identifier of the initiator group.

#### initiators

The initiators that are members of the initiator group.

Name	Туре	Description
comment	string	A comment available for use by the administrator.
name	string	Name of initiator that is a member of the initiator group.

#### igroup

The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

Name	Туре	Description
igroups	array[igroups]	Separate igroup definitions to include in this igroup.
initiators	array[initiators]	The initiators that are members of the group.
name	string	The name of the initiator group. Required in POST; optional in PATCH.
os_type	string	The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. Required in POST; optional in PATCH.
protocol	string	The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to <i>mixed</i> . The protocol of an initiator group cannot be changed after creation of the group.
uuid	string	The unique identifier of the initiator group.

#### lun\_maps

A LUN map is an association between a LUN and an initiator group.

When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

Name	Туре	Description
igroup	igroup	The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

Name	Туре	Description
logical_unit_number	integer	The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through the Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value. • Introduced in: 9.6 • readCreate: 1

# provisioning\_options

Options that are applied to the operation.

Name	Туре	Description
action	string	Operation to perform
count	integer	Number of elements to perform the operation on.

## policy

# The QoS policy

Name	Туре	Description
_links	self_link	
max_throughput_iops	integer	Specifies the maximum throughput in IOPS, 0 means none. This is mutually exclusive with name and UUID during POST and PATCH.
max_throughput_mbps	integer	Specifies the maximum throughput in Megabytes per sec, 0 means none. This is mutually exclusive with name and UUID during POST and PATCH.

Name	Туре	Description
min_throughput_iops	integer	Specifies the minimum throughput in IOPS, 0 means none. Setting "min_throughput" is supported on AFF platforms only, unless FabricPool tiering policies are set. This is mutually exclusive with name and UUID during POST and PATCH.
min_throughput_mbps	integer	Specifies the minimum throughput in Megabytes per sec, 0 means none. This is mutually exclusive with name and UUID during POST and PATCH.
name	string	The QoS policy group name. This is mutually exclusive with UUID and other QoS attributes during POST and PATCH.
uuid	string	The QoS policy group UUID. This is mutually exclusive with name and other QoS attributes during POST and PATCH.

### qos

Name	Туре	Description
policy	policy	The QoS policy

## space

The storage space related properties of the LUN.

size integer The total provisioned size of the LUN. The LUN size can be increased but not reduced using the REST interface. The maximum and minimum sizes listed here are the absolute minimum sizes, in bytes. The actual minimum and maxiumum sizes, in bytes. The actual minimum and maximum sizes vary depending on the ONTAP version, ONTAP platform, and the available space in the containing volume and aggregate. For more information, see <i>Size properties</i> in the <i>docs</i> section of the ONTAP REST API documentation. • example: 1073741824 • format: int64 • Max value: 140737488355328 • Min value: 4096 • Introduced in: 9.6	Name	Туре	Description
	size	integer	The total provisioned size of the LUN. The LUN size can be increased but not reduced using the REST interface. The maximum and minimum sizes listed here are the absolute maximum and absolute minimum sizes, in bytes. The actual minimum and maxiumum sizes vary depending on the ONTAP version, ONTAP platform, and the available space in the containing volume and aggregate. For more information, see <i>Size properties</i> in the <i>docs</i> section of the ONTAP REST API documentation. • example: 1073741824 • format: int64 • Max value: 140737488355328 • Min value: 4096 • Introduced in: 9.6

Name	Туре	Description
used	integer	The amount of space consumed by the main data stream of the LUN.
		This value is the total space consumed in the volume by the LUN, including filesystem overhead, but excluding prefix and suffix streams. Due to internal filesystem overhead and the many ways SAN filesystems and applications utilize blocks within a LUN, this value does not necessarily reflect actual consumption/availability from the perspective of the filesystem or application. Without specific knowledge of how the LUN blocks are utilized outside of ONTAP, this property should not be used as an indicator for an out-of-space condition. For more information, see <i>Size</i> <i>properties</i> in the <i>docs</i> section of the ONTAP REST API documentation.
		<ul> <li>format: int64</li> <li>roodOphy: 1</li> </ul>
		• readUnity. I
		• Introduced In: 9.6

#### luns

A LUN is the logical representation of storage in a storage area network (SAN).

In ONTAP, a LUN is located within a volume. Optionally, it can be located within a qtree in a volume.

A LUN can be created to a specified size using thin or thick provisioning. A LUN can then be renamed, resized, cloned, and moved to a different volume. LUNs support the assignment of a quality of service (QoS) policy for performance management or a QoS policy can be assigned to the volume containing the LUN. See the LUN object model to learn more about each of the properties supported by the LUN REST API.

A LUN must be mapped to an initiator group to grant access to the initiator group's initiators (client hosts). Initiators can then access the LUN and perform I/O over a Fibre Channel (FC) fabric using the Fibre Channel Protocol or a TCP/IP network using iSCSI.

Name	Туре	Description
clone	clone	This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: auto_delete, qos_policy, space.guarantee.requested and space.scsi_thin_provision ing_support_enabled. When used in a PATCH, the patched LUN's data is over- written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: class, auto_delete, lun_maps, serial_number, status.state, and uuid. Persistent reservations for the patched LUN are also preserved.
comment	string	A configurable comment available for use by the administrator. Valid in POST and PATCH.
create_time	string	The time the LUN was created.

enabledbooleanThe enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Cortain error conditions also cause the LUN containerror conditions also cause the LUN is disabled. If the LUN is disabled (offline) or has become disabled as arrow to determine if the enabled property to the enabled property to the enabled property to the enabled property to the LUN is enabled by default. Valid in PATCH.lun_mapsarray[un_maps]An array of LUN maps. A LUN map is an association between a LUN and an initiator group. The initiator group, the initiator group, the initiator group is many LUNs to many initiator groups.namestringThe fully qualified path name of the LUN is ename (optional), and the ase name of the LUN. The refers the volume name, the qree name (optional), and the ase name of the LUN. Nadie in POST and PATCH.os_lypestringThe operating system type of the LUN. Required in POST when creating a LUN done.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.qosqosoptions that are applied to the operation.	Name	Туре	Description
lun_mapsarray[lun_maps]An array of LUN maps.LUN mapsA LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group is initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.namestringThe fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the qtree name (optional), and the base name of the LUN. Valid in POST and PATCH.os_typestringThe operating system type of the LUN. Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.	enabled	boolean	The enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Certain error conditions also cause the LUN to become disabled. If the LUN is disabled, you can consult the state property to determine if the LUN is administratively disabled ( <i>offline</i> ) or has become disabled as a result of an error. A LUN in an error condition can be brought online by setting the enabled property to <i>true</i> or brought administratively offline by setting the enabled property to <i>false</i> . Upon creation, a LUN is enabled by default. Valid in PATCH.
namestringThe fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the qtree name (optional), and the base name of the LUN. Valid in POST and PATCH.os_typestringThe operating system type of the LUN. Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.qosqosstringImage: composite the string	lun_maps	array[lun_maps]	An array of LUN maps. A LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.
os_typestringThe operating system type of the LUN.Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.qosqosImage: constant options	name	string	The fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the qtree name (optional), and the base name of the LUN. Valid in POST and PATCH.
provisioning_optionsprovisioning_optionsOptions that are applied to the operation.qosqos	os_type	string	The operating system type of the LUN. Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.
qos qos	provisioning_options	provisioning_options	Options that are applied to the operation.
	qos	qos	

Name	Туре	Description
serial_number	string	The LUN serial number. The serial number is generated by ONTAP when the LUN is created. • maxLength: 12 • minLength: 12 • readOnly: 1 • Introduced in: 9.10
space	space	The storage space related properties of the LUN.
uuid	string	<ul> <li>The unique identifier of the LUN.</li> <li>The UUID is generated by ONTAP when the LUN is created.</li> <li>example: 1cd8a442-86d1- 11e0-ae1c-123478563412</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>

#### links

Name	Туре	Description
self	href	

nvme\_subsystem\_reference

An NVMe subsystem maintains configuration state and NVMe namespace access control for a set of NVMe-connected hosts.

Name	Туре	Description
_links	_links	
name	string	The name of the NVMe subsystem.
uuid	string	The unique identifier of the NVMe subsystem.

#### subsystem\_map

The NVMe subsystem with which the NVMe namespace is associated. A namespace can be mapped to zero (0) or one (1) subsystems.

There is an added cost to retrieving property values for subsystem\_map. They are not populated for

either a collection GET or an instance GET unless explicitly requested using the fields query parameter.

Name	Туре	Description
_links	self_link	
anagrpid	string	The Asymmetric Namespace Access Group ID (ANAGRPID) of the NVMe namespace. The format for an ANAGRPID is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
nsid	string	The NVMe namespace identifier. This is an identifier used by an NVMe controller to provide access to the NVMe namespace. The format for an NVMe namespace identifier is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
subsystem	nvme_subsystem_reference	An NVMe subsystem maintains configuration state and NVMe namespace access control for a set of NVMe-connected hosts.

#### namespaces

An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the storage virtual machine using the NVMe over Fabrics protocol.

In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.

An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API.

An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol.

Name	Туре	Description
auto_delete	boolean	This property marks the NVMe namespace for auto deletion when the volume containing the namespace runs out of space. This is most commonly set on namespace clones. When set to <i>true</i> , the NVMe namespace becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the namespace is also configured for auto deletion and free space in the volume decreases below a particular threshold. This property is optional in POST and PATCH. The default value for a new NVMe namespace is <i>false</i> . There is an added cost to retrieving this property's value. It is not populated for either a collection GET or an instance GET unless it is explicitly requested using the fields query parameter. See Requesting specific fields to learn more.
comment	string	A configurable comment available for use by the administrator. Valid in POST and PATCH.
create_time	string	The time the NVMe namespace was created.
enabled	boolean	The enabled state of the NVMe namespace. Certain error conditions cause the namespace to become disabled. If the namespace is disabled, you can check the state property to determine what error disabled the namespace. An NVMe namespace is enabled automatically when it is created.

Name	Туре	Description
name	string	The fully qualified path name of the NVMe namespace composed of a "/vol" prefix, the volume name, the (optional) qtree name and base name of the namespace. Valid in POST. NVMe namespaces do not support rename, or movement between volumes.
os_type	string	The operating system type of the NVMe namespace. Required in POST when creating an NVMe namespace that is not a clone of another. Disallowed in POST when creating a namespace clone.
provisioning_options	provisioning_options	Options that are applied to the operation.
subsystem_map	array[subsystem_map]	The NVMe subsystem with which the NVMe namespace is associated. A namespace can be mapped to zero (0) or one (1) subsystems. There is an added cost to retrieving property values for subsystem_map. They are not populated for either a collection GET or an instance GET unless explicitly requested using the fields query parameter.
uuid	string	The unique identifier of the NVMe namespace.

## parent\_consistency\_group

The parent consistency group.

Name	Туре	Description
_links	self_link	

Name	Туре	Description
name	string	The name of the consistency group.
uuid	string	The unique identifier of the consistency group.

storage\_service

Determines the placement of any storage object created during this operation.

Name	Туре	Description
name	string	Storage service name. If not specified, the default value is the most performant for the platform.

#### provisioning\_options

Options that are applied to the operation.

Name	Туре	Description
action	string	Operation to perform
storage_service	storage_service	Determines the placement of any storage object created during this operation.

### snapshot

A consistency group's Snapshot copy

Name	Туре	Description
name	string	The name of the consistency group's Snapshot copy to restore to.
uuid	string	The UUID of the consistency group's Snapshot copy to restore to.

## restore\_to

Use to restore a consistency group to a previous Snapshot copy

Name	Туре	Description
snapshot	snapshot	A consistency group's Snapshot copy

## snapshot\_policy\_reference

This is a reference to the Snapshot copy policy.

Name	Туре	Description
_links	_links	
name	string	
uuid	string	

space

Space information for the consistency group.

Name	Туре	Description
available	integer	The amount of space available in the consistency group, in bytes.
size	integer	The total provisioned size of the consistency group, in bytes.
used	integer	The amount of space consumed in the consistency group, in bytes.

#### svm\_reference

SVM, applies only to SVM-scoped objects.

Name	Туре	Description
_links	_links	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

## tiering

The tiering placement and policy definitions for volumes in this consistency group.

Туре	Description
string	Storage tiering placement rules for the object.
string	<ul> <li>Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold.</li> <li>FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH.</li> <li>all ‐ Allows tiering of both Snapshot copies and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.</li> <li>auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store.</li> <li>none ‐ Volume blocks are not be tiered to the cloud store.</li> <li>snapshot_only ‐ Allows tiering of only the volume Snapshot copies not associated with the active file system.</li> <li>The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering nolicy is 2</li> </ul>
	days and for the "auto" tiering policy it is 31 days.
	Type         string         string         Image: Imag

provisioning\_options

Options that are applied to the operation.

Name	Туре	Description
action	string	Operation to perform
count	integer	Number of elements to perform the operation on.
storage_service	storage_service	Determines the placement of any storage object created during this operation.

qos

The QoS policy for this volume.

Name	Туре	Description
policy	policy	The QoS policy

## space

Name	Туре	Description
available	integer	The available space, in bytes.
size	integer	Total provisioned size, in bytes.
used	integer	The virtual space used (includes volume reserves) before storage efficiency, in bytes.

## tiering

The tiering placement and policy definitions for this volume.

Name	Туре	Description
control	string	Storage tiering placement rules for the object.

string	Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold.
	FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH.
	all ‐ Allows tiering of both Snapshot copies and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.
	auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store
	none ‐ Volume blocks are not be tiered to the cloud store.
	snapshot_only ‐ Allows tiering of only the volume Snapshot copies not associated with the active file system.
	The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering policy is 2 days and for the "auto" tiering policy it is 31 days.
	string

#### volumes

Name	Туре	Description
comment	string	A comment for the volume. Valid in POST or PATCH.

Name	Туре	Description
language	string	Language encoding setting for volume. If no language is specified, the volume inherits its SVM language encoding setting.
name	string	Volume name. The name of volume must start with an alphabetic character (a to z or A to Z) or an underscore (_). The name must be 197 or fewer characters in length for FlexGroups, and 203 or fewer characters in length for all other types of volumes. Volume names must be unique within an SVM. Required on POST.
provisioning_options	provisioning_options	Options that are applied to the operation.
qos	qos	The QoS policy for this volume.
snapshot_policy	snapshot_policy_reference	This is a reference to the Snapshot copy policy.
space	space	
tiering	tiering	The tiering placement and policy definitions for this volume.
uuid	string	<ul> <li>Unique identifier for the volume. This corresponds to the instance- uuid that is exposed in the CLI and ONTAPI. It does not change due to a volume move.</li> <li>example: 028baa66-41bd- 11e9-81d5-00a0986138f7</li> <li>readOnly: 1</li> <li>Introduced in: 9.8</li> </ul>

## consistency\_groups

Name	Туре	Description
_links	self_link	

Name	Туре	Description
luns	array[luns]	The LUNs array can be used to create or modify LUNs in a consistency group on a new or existing volume that is a member of the consistency group. LUNs are considered members of a consistency group if they are located on a volume that is a member of the consistency group.
name	string	Name of the consistency group. The consistency group name must be unique within an SVM. If not provided and the consistency group contains only one volume, the name will be generated based on the volume name. If the consistency group contains more than one volume, the name is required.

Name	Туре	Description
namespaces	array[namespaces]	An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the SVM using the NVMe over Fabrics protocol. In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.
		An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API. An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe
		using the NVMe over Fabrics protocol.
		• minitems: U
		• uniqueitems: 1
		Introduced In: 9.10
		<ul> <li>x-ntap-modifyOnly: true</li> </ul>
parent_consistency_group	parent_consistency_group	The parent consistency group.
provisioning_options	provisioning_options	Options that are applied to the operation.

Name	Туре	Description
qos	qos	
restore_to	restore_to	Use to restore a consistency group to a previous Snapshot copy
snapshot_policy	snapshot_policy_reference	This is a reference to the Snapshot copy policy.
space	space	Space information for the consistency group.
svm	svm_reference	SVM, applies only to SVM- scoped objects.
tiering	tiering	The tiering placement and policy definitions for volumes in this consistency group.
uuid	string	<ul> <li>The unique identifier of the consistency group. The UUID is generated by ONTAP when the consistency group is created.</li> <li>example: 1cd8a442-86d1-11e0-ae1c-123478563412</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>
volumes	array[volumes]	A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A volume can only be associated with one direct parent consistency group. The volumes array can be used to create new volumes in the consistency group, add existing volumes to the consistency group, or modify existing volumes that are already members of the consistency group. The total number of volumes across all child consistency groups contained in a consistency group is constrained by the same limit.

#### records

Name	Туре	Description
_links	self_link	
consistency_groups	array[consistency_groups]	A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A consistency group can only be associated with one direct parent consistency group.
luns	array[luns]	The LUNs array can be used to create or modify LUNs in a consistency group on a new or existing volume that is a member of the consistency group. LUNs are considered members of a consistency group if they are located on a volume that is a member of the consistency group.
name	string	Name of the consistency group. The consistency group name must be unique within an SVM. If not provided and the consistency group contains only one volume, the name will be generated based on the volume name. If the consistency group contains more than one volume, the name is required.

Name	Туре	Description
namespaces	array[namespaces]	An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the SVM using the NVMe over Fabrics protocol. In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.
		An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API. An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol.
		• maxItems: 16
		• minItems: 0
		• uniqueltems: 1
		Introduced in: 9.10
		<ul> <li>x-ntap-modifyOnly: true</li> </ul>
parent_consistency_group	parent_consistency_group	The parent consistency group.
provisioning_options	provisioning_options	Options that are applied to the operation.

Name	Туре	Description
qos	qos	
replicated	boolean	Indicates whether or not replication has been enabled on this consistency group.
replication_source	boolean	Indicates whether or not this consistency group is the source for replication.
restore_to	restore_to	Use to restore a consistency group to a previous Snapshot copy
snapshot_policy	snapshot_policy_reference	This is a reference to the Snapshot copy policy.
space	space	Space information for the consistency group.
svm	svm_reference	SVM, applies only to SVM- scoped objects.
tiering	tiering	The tiering placement and policy definitions for volumes in this consistency group.
uuid	string	The unique identifier of the consistency group. The UUID is generated by ONTAP when the consistency group is created. • example: 1cd8a442-86d1- 11e0-ae1c-123478563412 • readOnly: 1 • Introduced in: 9.6

Name	Туре	Description
volumes	array[volumes]	DescriptionA consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A volume can only be associated with one direct parent consistency group.The volumes array can be used to create new volumes in the consistency group, add existing volumes to the consistency group, or modify existing volumes that are already members of the consistency group.The total number of volumes across all child consistency group is constrained
		by the same limit.

## error\_arguments

Name	Туре	Description
code	string	Argument code
message	string	Message argument

error

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

# Create a consistency group

POST /application/consistency-groups

Introduced In: 9.10

Creates a consistency group with one or more consistency groups having:

- new SAN volumes,
- existing SAN, NVMe or NAS FlexVol volumes in a new or existing consistency group

# **Required properties**

- svm.uuid or svm.name Existing SVM in which to create the group.
- volumes, luns **or** namespaces

# **Naming Conventions**

## **Consistency groups**

- name or consistency\_groups[].name, if specified
- derived from volumes[0].name, if only one volume is specified, same as volume name

## Volume

- volumes[].name, if specified
- derived from volume prefix in luns[].name
- derived from cg[].name, suffixed by "\_#" where "#" is a system generated unique number
- suffixed by "\_#" where "#" is a system generated unique number, if provisioning\_options.count is provided

# LUN

- luns[].name, if specified
- derived from volumes[].name, suffixed by "\_#" where "#" is a system generated unique number
- suffixed by "\_#" where "#" is a system generated unique number, if provisioning\_options.count is provided

## **NVMe Namespace**

- namespaces[].name, if specified
- derived from volumes[].name, suffixed by "\_#" where "#" is a system generated unique number
- suffixed by "\_#" where "#" is a system generated unique number, if provisioning\_options.count is provided

# **Related ONTAP commands**

There are no ONTAP commands for managing consistency group.

## **Parameters**

Name	Туре	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
return_records	boolean	query	False	The default is false. If set to true, the records are returned. • Default value:

# **Request Body**

Name	Туре	Description
_links	self_link	

Name	Туре	Description
consistency_groups	array[consistency_groups]	A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A consistency group can only be associated with one direct parent consistency group.
luns	array[luns]	The LUNs array can be used to create or modify LUNs in a consistency group on a new or existing volume that is a member of the consistency group. LUNs are considered members of a consistency group if they are located on a volume that is a member of the consistency group.
name	string	Name of the consistency group. The consistency group name must be unique within an SVM. If not provided and the consistency group contains only one volume, the name will be generated based on the volume name. If the consistency group contains more than one volume, the name is required.

Name	Туре	Description
namespaces	array[namespaces]	An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the SVM using the NVMe over Fabrics protocol. In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.
		An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API. An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol. • maxItems: 16 • minItems: 0 • uniqueItems: 1 • Introduced in: 9.10 • x-ntap-modifyOnly: true
parent_consistency_group	parent_consistency_group	The parent consistency group.
provisioning_options	provisioning_options	Options that are applied to the operation.
qos	qos	

Name	Туре	Description
replicated	boolean	Indicates whether or not replication has been enabled on this consistency group.
replication_source	boolean	Indicates whether or not this consistency group is the source for replication.
restore_to	restore_to	Use to restore a consistency group to a previous Snapshot copy
snapshot_policy	snapshot_policy_reference	This is a reference to the Snapshot copy policy.
space	space	Space information for the consistency group.
svm	svm_reference	SVM, applies only to SVM-scoped objects.
tiering	tiering	The tiering placement and policy definitions for volumes in this consistency group.
uuid	string	<ul> <li>The unique identifier of the consistency group. The UUID is generated by ONTAP when the consistency group is created.</li> <li>example: 1cd8a442-86d1-11e0-ae1c-123478563412</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>
Name	Туре	Description
---------	----------------	---
volumes	array[volumes]	A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A volume can only be associated with one direct parent consistency group. The volumes array can be used to create new volumes in the consistency group, add existing volumes to the consistency group, or modify existing volumes that are already members of the consistency group. The total number of volumes across all child consistency groups contained in a consistency group is
		constrained by the same limit.

Example request

```
{
 " links": {
   "self": {
     "href": "/api/resourcelink"
   }
 },
 "consistency groups": {
   " links": {
    "self": {
       "href": "/api/resourcelink"
     }
   },
   "luns": {
     "clone": {
       "source": {
         "name": "/vol/volume1/lun1",
         "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
       }
     },
     "comment": "string",
     "create time": "2018-06-04T19:00:00Z",
     "lun maps": {
       "igroup": {
         "igroups": {
            " links": {
             "self": {
                "href": "/api/resourcelink"
             }
            },
            "name": "igroup1",
           "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
          },
         "initiators": {
           "comment": "my comment",
           "name": "iqn.1998-01.com.corp.iscsi:name1"
         },
         "name": "igroup1",
         "os type": "aix",
         "protocol": "fcp",
         "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
       }
     },
     "name": "/vol/volume1/lun1",
     "os type": "aix",
```

```
"provisioning options": {
    "action": "create"
 },
  "qos": {
    "policy": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
      "uuid": "lcd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "serial number": "string",
  "space": {
    "size": 1073741824,
   "used": 0
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"namespaces": {
  "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "name": "/vol/volume1/qtree1/namespace1",
  "os type": "aix",
  "provisioning options": {
    "action": "create"
  },
  "subsystem map": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "anagrpid": "00103050h",
    "nsid": "00000001h",
    "subsystem": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
        }
```

```
},
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"parent consistency group": {
  " links": {
   "self": {
      "href": "/api/resourcelink"
    }
  },
 "name": "my consistency group",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"provisioning options": {
  "action": "create",
 "storage service": {
   "name": "extreme"
 }
},
"qos": {
  "policy": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
     }
    },
    "max throughput iops": 10000,
    "max throughput mbps": 500,
    "min throughput iops": 2000,
    "min throughput mbps": 500,
    "name": "performance",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
 }
},
"snapshot policy": {
  " links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "name": "default",
 "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"space": {
```

```
"available": 5737418,
  "size": 1073741824,
 "used": 5737418
},
"svm": {
  " links": {
   "self": {
      "href": "/api/resourcelink"
    }
  },
 "name": "svm1",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"tiering": {
 "control": "allowed",
 "policy": "all"
},
"uuid": "lcd8a442-86d1-11e0-ae1c-123478563412",
"volumes": {
 "comment": "string",
 "language": "ar",
  "name": "vol cs dept",
  "provisioning options": {
   "action": "create",
    "storage service": {
     "name": "extreme"
   }
  },
  "qos": {
    "policy": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "snapshot policy": {
    " links": {
      "self": {
```

```
"href": "/api/resourcelink"
       }
      },
      "name": "default",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "space": {
     "available": 0,
     "used": 0
    },
    "tiering": {
     "control": "allowed",
     "policy": "all"
    },
    "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
  }
},
"luns": {
  "clone": {
   "source": {
      "name": "/vol/volume1/lun1",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "lun maps": {
    "igroup": {
      "igroups": {
        " links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "initiators": {
        "comment": "my comment",
        "name": "iqn.1998-01.com.corp.iscsi:name1"
      },
      "name": "igroup1",
      "os type": "aix",
      "protocol": "fcp",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
```

```
},
  "name": "/vol/volume1/lun1",
  "os type": "aix",
  "provisioning options": {
   "action": "create"
  },
  "qos": {
    "policy": {
      " links": {
       "self": {
          "href": "/api/resourcelink"
       }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "serial number": "string",
  "space": {
   "size": 1073741824,
   "used": 0
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"namespaces": {
  "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "name": "/vol/volume1/qtree1/namespace1",
  "os type": "aix",
  "provisioning options": {
   "action": "create"
  },
  "subsystem map": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "anagrpid": "00103050h",
    "nsid": "00000001h",
    "subsystem": {
      " links": {
```

```
"self": {
          "href": "/api/resourcelink"
        }
      },
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"parent consistency group": {
  " links": {
   "self": {
     "href": "/api/resourcelink"
   }
  },
  "name": "my consistency group",
  "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"provisioning options": {
  "action": "create",
 "storage service": {
   "name": "extreme"
 }
},
"qos": {
  "policy": {
    " links": {
     "self": {
        "href": "/api/resourcelink"
     }
    },
    "max throughput iops": 10000,
    "max throughput mbps": 500,
    "min throughput iops": 2000,
    "min throughput mbps": 500,
    "name": "performance",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
 }
},
"snapshot policy": {
  " links": {
   "self": {
     "href": "/api/resourcelink"
   }
  },
  "name": "default",
```

```
"uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"space": {
 "available": 5737418,
 "size": 1073741824,
 "used": 5737418
},
"svm": {
  " links": {
   "self": {
     "href": "/api/resourcelink"
   }
  },
  "name": "svm1",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"tiering": {
 "control": "allowed",
 "policy": "all"
},
"uuid": "lcd8a442-86d1-11e0-ae1c-123478563412",
"volumes": {
  "comment": "string",
  "language": "ar",
  "name": "vol cs dept",
  "provisioning options": {
    "action": "create",
   "storage service": {
     "name": "extreme"
   }
  },
  "qos": {
    "policy": {
      " links": {
       "self": {
         "href": "/api/resourcelink"
       }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
```

```
"snapshot policy": {
      " links": {
       "self": {
         "href": "/api/resourcelink"
       }
     },
      "name": "default",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "space": {
     "available": 0,
    "used": 0
   },
    "tiering": {
     "control": "allowed",
    "policy": "all"
   },
    "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
 }
}
```

# Response

```
Status: 202, Accepted
```

Name	Туре	Description
job	job_link	

### Example response

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

# Response

Status: 201, Created

# Error

Status: Default

# ONTAP Error Response Codes

Error Code	Description
53411842	Consistency group does not exist.
53411843	A consistency group with specified UUID was not found.
53411844	Specified consistency group was not found in the specified SVM.
53411845	The specified UUID and name refer to different consistency groups.
53411846	Either name or UUID must be provided.
53411853	Fields provided in the request conflict with each other.
53411856	Field provided is only supported when provisioning new objects.
53411857	LUNs that are not members of the application are not supported by this API. LUNs can be added to an application by adding the volume containing the LUNs to the application.
53411860	An object with the same identifier in the same scope exists.
53411861	Volume specified does not exist in provided volume array.
53411862	Modifying existing igroups is not supported using this API.
53411864	Request content insufficient to add an existing volume to an application.
53411865	Volumes contained in one consistency group can not be added to a different consistency group.
53411866	LUNs are not supported on FlexGroups volumes.
53411867	LUN name is too long after appending a unique suffix.
53411869	Volume name is too long after appending a unique suffix.

Error Code	Description
53411870	When using the "round_robin" layout, the volume count must not be greater than the LUN count.

Name	Туре	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	Туре	Description
href	string	

self\_link

Name	Туре	Description
self	href	

source

The source LUN for a LUN clone operation. This can be specified using property clone.source.uuid or clone.source.name. If both properties are supplied, they must refer to the same LUN.

Valid in POST to create a new LUN as a clone of the source.

Valid in PATCH to overwrite an existing LUN's data as a clone of another.

Name	Туре	Description
name	string	The fully qualified path name of the clone source LUN composed of a "/vol" prefix, the volume name, the (optional) qtree name, and base name of the LUN. Valid in POST and PATCH.
uuid	string	The unique identifier of the clone source LUN. Valid in POST and PATCH.

#### clone

This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: auto\_delete, gos\_policy, space.guarantee.requested and space.scsi\_thin\_provisioning\_support\_enabled.

When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: class, auto\_delete, lun\_maps, serial\_number, status.state, and uuid.

Persistent reservations for the patched LUN are also preserved.

Name	Туре	Description
source	source	The source LUN for a LUN clone operation. This can be specified using property clone.source.uuid or clone.source.name. If both properties are supplied, they must refer to the same LUN. Valid in POST to create a new LUN as a clone of the source. Valid in PATCH to overwrite an existing LUN's data as a clone of another.

### igroups

Name	Туре	Description
_links	self_link	
name	string	The name of the initiator group.
uuid	string	The unique identifier of the initiator group.

### initiators

The initiators that are members of the initiator group.

Name	Туре	Description
comment	string	A comment available for use by the administrator.
name	string	Name of initiator that is a member of the initiator group.

## igroup

The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

Name	Туре	Description
igroups	array[igroups]	Separate igroup definitions to include in this igroup.

Name	Туре	Description
initiators	array[initiators]	The initiators that are members of the group.
name	string	The name of the initiator group. Required in POST; optional in PATCH.
os_type	string	The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. Required in POST; optional in PATCH.
protocol	string	The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to <i>mixed</i> . The protocol of an initiator group cannot be changed after creation of the group.
uuid	string	The unique identifier of the initiator group.

lun\_maps

A LUN map is an association between a LUN and an initiator group.

When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

Name	Туре	Description
igroup	igroup	The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

Name	Туре	Description
logical_unit_number	integer	The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through the Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value. • Introduced in: 9.6 • readCreate: 1

# provisioning\_options

Options that are applied to the operation.

Name	Туре	Description
action	string	Operation to perform
count	integer	Number of elements to perform the operation on.

# policy

# The QoS policy

Name	Туре	Description
_links	self_link	
max_throughput_iops	integer	Specifies the maximum throughput in IOPS, 0 means none. This is mutually exclusive with name and UUID during POST and PATCH.
max_throughput_mbps	integer	Specifies the maximum throughput in Megabytes per sec, 0 means none. This is mutually exclusive with name and UUID during POST and PATCH.

Name	Туре	Description
min_throughput_iops	integer	Specifies the minimum throughput in IOPS, 0 means none. Setting "min_throughput" is supported on AFF platforms only, unless FabricPool tiering policies are set. This is mutually exclusive with name and UUID during POST and PATCH.
min_throughput_mbps	integer	Specifies the minimum throughput in Megabytes per sec, 0 means none. This is mutually exclusive with name and UUID during POST and PATCH.
name	string	The QoS policy group name. This is mutually exclusive with UUID and other QoS attributes during POST and PATCH.
uuid	string	The QoS policy group UUID. This is mutually exclusive with name and other QoS attributes during POST and PATCH.

### qos

Name	Туре	Description
policy	policy	The QoS policy

### space

The storage space related properties of the LUN.

Name	Туре	Description
size	integer	The total provisioned size of the LUN. The LUN size can be increased but not reduced using the REST interface. The maximum and minimum sizes listed here are the absolute maximum and absolute minimum sizes, in bytes. The actual minimum and maxiumum sizes vary depending on the ONTAP version, ONTAP platform, and the available space in the containing volume and aggregate. For more information, see <i>Size properties</i> in the <i>docs</i> section of the ONTAP REST API documentation. • example: 1073741824 • format: int64 • Max value: 140737488355328 • Min value: 4096 • Introduced in: 9.6

Name	Туре	Description
used	integer	The amount of space consumed by the main data stream of the LUN.
		This value is the total space consumed in the volume by the LUN, including filesystem overhead, but excluding prefix and suffix streams. Due to internal filesystem overhead and the many ways SAN filesystems and applications utilize blocks within a LUN, this value does not necessarily reflect actual consumption/availability from the perspective of the filesystem or application. Without specific knowledge of how the LUN blocks are utilized outside of ONTAP, this property should not be used as an indicator for an out-of-space condition. For more information, see <i>Size</i> <i>properties</i> in the <i>docs</i> section of the ONTAP REST API documentation.
		• format: int64
		• readOnly: 1
		Introduced in: 9.6

#### luns

A LUN is the logical representation of storage in a storage area network (SAN).

In ONTAP, a LUN is located within a volume. Optionally, it can be located within a qtree in a volume.

A LUN can be created to a specified size using thin or thick provisioning. A LUN can then be renamed, resized, cloned, and moved to a different volume. LUNs support the assignment of a quality of service (QoS) policy for performance management or a QoS policy can be assigned to the volume containing the LUN. See the LUN object model to learn more about each of the properties supported by the LUN REST API.

A LUN must be mapped to an initiator group to grant access to the initiator group's initiators (client hosts). Initiators can then access the LUN and perform I/O over a Fibre Channel (FC) fabric using the Fibre Channel Protocol or a TCP/IP network using iSCSI.

Name	Туре	Description
clone	clone	This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: auto_delete, qos_policy, space.guarantee.requested and space.scsi_thin_provision ing_support_enabled. When used in a PATCH, the patched LUN's data is over- written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: class, auto_delete, lun_maps, serial_number, status.state, and uuid. Persistent reservations for the patched LUN are also preserved.
comment	string	A configurable comment available for use by the administrator. Valid in POST and PATCH.
create_time	string	The time the LUN was created.

enabledbooleanThe enabled state of the LUN. LUNs can be disabled to prevent access to the LUN Certain ere conditions also cause the LUN certain ere to determine if the LUN is disabled, you can consult the state property to determine if the LUN is disabled, you can consult the state property to determine if the LUN is disabled a competition can be brought administratively disabled (offline) or has become disabled (offline) or has become disabled approperty to determine if the competition can be brought administratively disabled (offline) or has become disabled approperty to the or brought administratively disabled approperty to the or able of property to the or brought administratively disabled approperty to the or brought administratively disabled approperty to the telsonship between a LUN and an initiator group is many LUNs to many initiator group is many LUNs to many initiator group is	Name	Туре	Description
lun_mapsarray[lun_maps]An array of LUN maps.LUN mapsA LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.namestringThe fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the qtree name (optional), and the base name of the LUN. Valid in POST and PATCH.os_typestringThe operating system type of the LUN.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.	enabled	boolean	The enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Certain error conditions also cause the LUN to become disabled. If the LUN is disabled, you can consult the state property to determine if the LUN is administratively disabled ( <i>offline</i> ) or has become disabled as a result of an error. A LUN in an error condition can be brought online by setting the enabled property to <i>true</i> or brought administratively offline by setting the enabled property to <i>false</i> . Upon creation, a LUN is enabled by default. Valid in PATCH.
namestringThe fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the qtree name (optional), and the base name of the LUN. Valid in POST and PATCH.os_typestringThe operating system type of the LUN. Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.qosqosprovisioning_optionsprovisioning_options	lun_maps	array[lun_maps]	An array of LUN maps. A LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.
os_typestringThe operating system type of the LUN.Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.qosqos	name	string	The fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the qtree name (optional), and the base name of the LUN. Valid in POST and PATCH.
provisioning_optionsProvisioning_optionsOptions that are applied to the operation.qosqos	os_type	string	The operating system type of the LUN. Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.
qos qos	provisioning_options	provisioning_options	Options that are applied to the operation.
	qos	qos	

Name	Туре	Description
serial_number	string	<ul> <li>The LUN serial number. The serial number is generated by ONTAP when the LUN is created.</li> <li>maxLength: 12</li> <li>minLength: 12</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>
space	space	The storage space related properties of the LUN.
uuid	string	<ul> <li>The unique identifier of the LUN.</li> <li>The UUID is generated by ONTAP when the LUN is created.</li> <li>example: 1cd8a442-86d1- 11e0-ae1c-123478563412</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>

#### \_links

Name	Туре	Description
self	href	

nvme\_subsystem\_reference

An NVMe subsystem maintains configuration state and NVMe namespace access control for a set of NVMe-connected hosts.

Name	Туре	Description
_links	_links	
name	string	The name of the NVMe subsystem.
uuid	string	The unique identifier of the NVMe subsystem.

### subsystem\_map

The NVMe subsystem with which the NVMe namespace is associated. A namespace can be mapped to zero (0) or one (1) subsystems.

There is an added cost to retrieving property values for subsystem\_map. They are not populated for

either a collection GET or an instance GET unless explicitly requested using the fields query parameter.

Name	Туре	Description
_links	self_link	
anagrpid	string	The Asymmetric Namespace Access Group ID (ANAGRPID) of the NVMe namespace. The format for an ANAGRPID is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
nsid	string	The NVMe namespace identifier. This is an identifier used by an NVMe controller to provide access to the NVMe namespace. The format for an NVMe namespace identifier is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
subsystem	nvme_subsystem_reference	An NVMe subsystem maintains configuration state and NVMe namespace access control for a set of NVMe-connected hosts.

#### namespaces

An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the storage virtual machine using the NVMe over Fabrics protocol.

In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.

An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API.

An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol.

Name	Туре	Description
auto_delete	boolean	This property marks the NVMe namespace for auto deletion when the volume containing the namespace runs out of space. This is most commonly set on namespace clones. When set to <i>true</i> , the NVMe namespace becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the namespace is also configured for auto deletion and free space in the volume decreases below a particular threshold. This property is optional in POST and PATCH. The default value for a new NVMe namespace is <i>false</i> . There is an added cost to retrieving this property's value. It is not populated for either a collection GET or an instance GET unless it is explicitly requested using the fields query parameter. See Requesting specific fields to learn more.
comment	string	A configurable comment available for use by the administrator. Valid in POST and PATCH.
create_time	string	The time the NVMe namespace was created.
enabled	boolean	The enabled state of the NVMe namespace. Certain error conditions cause the namespace to become disabled. If the namespace is disabled, you can check the state property to determine what error disabled the namespace. An NVMe namespace is enabled automatically when it is created.

Name	Туре	Description
name	string	The fully qualified path name of the NVMe namespace composed of a "/vol" prefix, the volume name, the (optional) qtree name and base name of the namespace. Valid in POST. NVMe namespaces do not support rename, or movement between volumes.
os_type	string	The operating system type of the NVMe namespace. Required in POST when creating an NVMe namespace that is not a clone of another. Disallowed in POST when creating a namespace clone.
provisioning_options	provisioning_options	Options that are applied to the operation.
subsystem_map	array[subsystem_map]	The NVMe subsystem with which the NVMe namespace is associated. A namespace can be mapped to zero (0) or one (1) subsystems. There is an added cost to retrieving property values for subsystem_map. They are not populated for either a collection GET or an instance GET unless explicitly requested using the fields query parameter.
uuid	string	The unique identifier of the NVMe namespace.

# parent\_consistency\_group

The parent consistency group.

Name	Туре	Description
_links	self_link	

Name	Туре	Description
name	string	The name of the consistency group.
uuid	string	The unique identifier of the consistency group.

storage\_service

Determines the placement of any storage object created during this operation.

Name	Туре	Description
name	string	Storage service name. If not specified, the default value is the most performant for the platform.

### provisioning\_options

Options that are applied to the operation.

Name	Туре	Description
action	string	Operation to perform
storage_service	storage_service	Determines the placement of any storage object created during this operation.

### snapshot

A consistency group's Snapshot copy

Name	Туре	Description
name	string	The name of the consistency group's Snapshot copy to restore to.
uuid	string	The UUID of the consistency group's Snapshot copy to restore to.

## restore\_to

Use to restore a consistency group to a previous Snapshot copy

Name	Туре	Description
snapshot	snapshot	A consistency group's Snapshot copy

### snapshot\_policy\_reference

This is a reference to the Snapshot copy policy.

Name	Туре	Description
_links	_links	
name	string	
uuid	string	

space

Space information for the consistency group.

Name	Туре	Description
available	integer	The amount of space available in the consistency group, in bytes.
size	integer	The total provisioned size of the consistency group, in bytes.
used	integer	The amount of space consumed in the consistency group, in bytes.

#### svm\_reference

SVM, applies only to SVM-scoped objects.

Name	Туре	Description
_links	_links	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

## tiering

The tiering placement and policy definitions for volumes in this consistency group.

Name	Туре	Description
control	string	Storage tiering placement rules for the object.
policy	string	Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold. FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH. all ‐ Allows tiering of both Snapshot copies and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks. auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store none ‐ Volume blocks are not be tiered to the cloud store. snapshot_only ‐ Allows tiering of only the volume Snapshot copies not associated with the active file system. The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering policy is 2
		average for the "auto" tiering policy is 2 days and for the "auto" tiering policy it is 31 days.

provisioning\_options

Options that are applied to the operation.

Name	Туре	Description
action	string	Operation to perform
count	integer	Number of elements to perform the operation on.
storage_service	storage_service	Determines the placement of any storage object created during this operation.

qos

The QoS policy for this volume.

Name	Туре	Description
policy	policy	The QoS policy

### space

Name	Туре	Description
available	integer	The available space, in bytes.
size	integer	Total provisioned size, in bytes.
used	integer	The virtual space used (includes volume reserves) before storage efficiency, in bytes.

## tiering

The tiering placement and policy definitions for this volume.

Name	Туре	Description
control	string	Storage tiering placement rules for the object.

Туре	Description
string	Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold.
	FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH.
	all ‐ Allows tiering of both Snapshot copies and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.
	auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store
	none ‐ Volume blocks are not be tiered to the cloud store.
	snapshot_only ‐ Allows tiering of only the volume Snapshot copies not associated with the active file system.
	The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering policy is 2 days and for the "auto" tiering policy it is 31 days.
	Type         string

#### volumes

Name	Туре	Description
comment	string	A comment for the volume. Valid in POST or PATCH.

Name	Туре	Description
language	string	Language encoding setting for volume. If no language is specified, the volume inherits its SVM language encoding setting.
name	string	Volume name. The name of volume must start with an alphabetic character (a to z or A to Z) or an underscore (_). The name must be 197 or fewer characters in length for FlexGroups, and 203 or fewer characters in length for all other types of volumes. Volume names must be unique within an SVM. Required on POST.
provisioning_options	provisioning_options	Options that are applied to the operation.
qos	qos	The QoS policy for this volume.
snapshot_policy	snapshot_policy_reference	This is a reference to the Snapshot copy policy.
space	space	
tiering	tiering	The tiering placement and policy definitions for this volume.
uuid	string	<ul> <li>Unique identifier for the volume. This corresponds to the instance- uuid that is exposed in the CLI and ONTAPI. It does not change due to a volume move.</li> <li>example: 028baa66-41bd- 11e9-81d5-00a0986138f7</li> <li>readOnly: 1</li> <li>Introduced in: 9.8</li> </ul>

# consistency\_groups

Name	Туре	Description
_links	self_link	

Name	Туре	Description
luns	array[luns]	The LUNs array can be used to create or modify LUNs in a consistency group on a new or existing volume that is a member of the consistency group. LUNs are considered members of a consistency group if they are located on a volume that is a member of the consistency group.
name	string	Name of the consistency group. The consistency group name must be unique within an SVM. If not provided and the consistency group contains only one volume, the name will be generated based on the volume name. If the consistency group contains more than one volume, the name is required.

Name	Туре	Description
namespaces	array[namespaces]	An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the SVM using the NVMe over Fabrics protocol. In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.
		An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API. An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol.
		• maxItems: 16
		• minItems: 0
		• uniqueltems: 1
		<ul> <li>Introduced in: 9.10</li> </ul>
		• x-ntap-modifyOnly: true
parent_consistency_group	parent_consistency_group	The parent consistency group.
provisioning_options	provisioning_options	Options that are applied to the operation.

Name	Туре	Description
qos	qos	
restore_to	restore_to	Use to restore a consistency group to a previous Snapshot copy
snapshot_policy	snapshot_policy_reference	This is a reference to the Snapshot copy policy.
space	space	Space information for the consistency group.
svm	svm_reference	SVM, applies only to SVM- scoped objects.
tiering	tiering	The tiering placement and policy definitions for volumes in this consistency group.
uuid	string	<ul> <li>The unique identifier of the consistency group. The UUID is generated by ONTAP when the consistency group is created.</li> <li>example: 1cd8a442-86d1-11e0-ae1c-123478563412</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>
volumes	array[volumes]	A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A volume can only be associated with one direct parent consistency group. The volumes array can be used to create new volumes in the consistency group, add existing volumes to the consistency group, or modify existing volumes that are already members of the consistency group. The total number of volumes across all child consistency groups contained in a consistency group is constrained by the same limit.

### consistency\_group

Name	Туре	Description
_links	self_link	
consistency_groups	array[consistency_groups]	A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A consistency group can only be associated with one direct parent consistency group.
luns	array[luns]	The LUNs array can be used to create or modify LUNs in a consistency group on a new or existing volume that is a member of the consistency group. LUNs are considered members of a consistency group if they are located on a volume that is a member of the consistency group.
name	string	Name of the consistency group. The consistency group name must be unique within an SVM. If not provided and the consistency group contains only one volume, the name will be generated based on the volume name. If the consistency group contains more than one volume, the name is required.

Name	Туре	Description
Name namespaces	Type         array[namespaces]	Description An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the SVM using the NVMe over Fabrics protocol. In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume. An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API. An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol. • maxItems: 16 • minItems: 0
		• uniqueltems: 1
		<ul> <li>Introduced in: 9.10</li> <li>x-ntap-modifyOnly: true</li> </ul>
parent_consistency_group	parent_consistency_group	The parent consistency group.
provisioning_options	provisioning_options	Options that are applied to the operation.
Indicates whether or not replication has been enabled on this consistency group. Indicates whether or not this consistency group is the source for replication.		
---		
Indicates whether or not replication has been enabled on this consistency group. Indicates whether or not this consistency group is the source for replication.		
Indicates whether or not this consistency group is the source for replication.		
Use to restore a consistency group to a previous Snapshot copy		
This is a reference to the Snapshot copy policy.		
Space information for the consistency group.		
SVM, applies only to SVM- scoped objects.		
The tiering placement and policy definitions for volumes in this consistency group.		
The unique identifier of the consistency group. The UUID is generated by ONTAP when the consistency group is created. • example: 1cd8a442-86d1- 11e0-ae1c-123478563412		

Name	Туре	Description
volumes	array[volumes]	A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A volume can only be associated with one direct parent consistency group.
		The volumes array can be used to create new volumes in the consistency group, add existing volumes to the consistency group, or modify existing volumes that are already members of the consistency group.
		The total number of volumes across all child consistency groups contained in a consistency group is constrained by the same limit.

# job\_link

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

### error\_arguments

Name	Туре	Description
code	string	Argument code
message	string	Message argument

#### error

Name	Туре	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message

Name	Туре	Description
target	string	The target parameter that caused the error.

# Delete a consistency group

DELETE /application/consistency-groups/{uuid}

### Introduced In: 9.10

Deletes a consistency group.



this will not delete any associated volumes or LUNs. To remove those elements, you can use the appropriate object endpoint.

## **Related ONTAP commands**

There are no ONTAP commands for managing consistency groups.

## **Parameters**

Name	Туре	In	Required	Description
uuid	string	path	True	The unique identifier of the consistency group to delete.
delete_data	boolean	query	False	Delete the underlying storage as well as the consistency group association. This parameter should be used with caution. • Default value:

Name	Туре	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

# Response

Status: 200, Ok

# Response

Status: 202, Accepted

# Error

Status: Default

Error Code	Description
53411842	Consistency group does not exist.
53411843	A consistency group with specified UUID was not found.
53411844	Specified consistency group was not found in the specified SVM.
53411845	The specified UUID and name refer to different consistency groups.
53411846	Either name or UUID must be provided.

Name	Туре	Description
error	error	

Example error

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

# Definitions

#### **See Definitions**

error\_arguments

Name	Туре	Description
code	string	Argument code
message	string	Message argument

error

Name	Туре	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 a consistency group

**GET** /application/consistency-groups/{uuid}

### Introduced In: 9.10

Retrieves a single consistency group.

## **Expensive properties**

There is an added 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 DOC Requesting specific fields to learn more.

- volumes
- luns
- namespaces

## **Related ONTAP commands**

There are no ONTAP commands for managing consistency groups.

## **Parameters**

Name	Туре	In	Required	Description
uuid	string	path	True	The unique identifier of the group to retrieve.
luns.space.size	integer	query	False	Filter by luns.space.size
luns.space.used	integer	query	False	Filter by luns.space.used
luns.enabled	boolean	query	False	Filter by luns.enabled
luns.name	string	query	False	Filter by luns.name
luns.lun_maps.logica l_unit_number	integer	query	False	Filter by luns.lun_maps.logic al_unit_number
luns.lun_maps.igrou p.uuid	string	query	False	Filter by luns.lun_maps.igrou p.uuid
luns.lun_maps.igrou p.os_type	string	query	False	Filter by luns.lun_maps.igrou p.os_type
luns.lun_maps.igrou p.igroups.uuid	string	query	False	Filter by luns.lun_maps.igrou p.igroups.uuid
luns.lun_maps.igrou p.igroups.name	string	query	False	Filter by luns.lun_maps.igrou p.igroups.name
luns.lun_maps.igrou p.name	string	query	False	Filter by luns.lun_maps.igrou p.name
luns.lun_maps.igrou p.protocol	string	query	False	Filter by luns.lun_maps.igrou p.protocol

Name	Туре	In	Required	Description
luns.lun_maps.igrou p.initiators.comment	string	query	False	Filter by luns.lun_maps.igrou p.initiators.comment
luns.lun_maps.igrou p.initiators.name	string	query	False	Filter by luns.lun_maps.igrou p.initiators.name
luns.create_time	string	query	False	Filter by luns.create_time
luns.os_type	string	query	False	Filter by luns.os_type
luns.qos.policy.uuid	string	query	False	Filter by luns.qos.policy.uuid
luns.qos.policy.min_t hroughput_iops	integer	query	False	Filter by luns.qos.policy.min_ throughput_iops
luns.qos.policy.max_ throughput_iops	integer	query	False	Filter by luns.qos.policy.max _throughput_iops
luns.qos.policy.min_t hroughput_mbps	integer	query	False	Filter by luns.qos.policy.min_ throughput_mbps
luns.qos.policy.name	string	query	False	Filter by luns.qos.policy.nam e
luns.qos.policy.max_ throughput_mbps	integer	query	False	Filter by luns.qos.policy.max _throughput_mbps
luns.uuid	string	query	False	Filter by luns.uuid
luns.serial_number	string	query	False	Filter by luns.serial_number
luns.comment	string	query	False	Filter by luns.comment
replicated	boolean	query	False	Filter by replicated

Name	Туре	In	Required	Description
qos.policy.min_throu ghput_mbps	integer	query	False	Filter by qos.policy.min_throu ghput_mbps
qos.policy.name	string	query	False	Filter by qos.policy.name
qos.policy.max_thro ughput_mbps	integer	query	False	Filter by qos.policy.max_thro ughput_mbps
qos.policy.uuid	string	query	False	Filter by qos.policy.uuid
qos.policy.min_throu ghput_iops	integer	query	False	Filter by qos.policy.min_throu ghput_iops
qos.policy.max_thro ughput_iops	integer	query	False	Filter by qos.policy.max_thro ughput_iops
svm.uuid	string	query	False	Filter by svm.uuid
svm.name	string	query	False	Filter by svm.name
space.used	integer	query	False	Filter by space.used
space.available	integer	query	False	Filter by space.available
space.size	integer	query	False	Filter by space.size
tiering.policy	string	query	False	Filter by tiering.policy
parent_consistency_ group.uuid	string	query	False	Filter by parent_consistency_ group.uuid
parent_consistency_ group.name	string	query	False	Filter by parent_consistency_ group.name

Name	Туре	In	Required	Description
snapshot_policy.uuid	string	query	False	Filter by snapshot_policy.uui d
snapshot_policy.na me	string	query	False	Filter by snapshot_policy.na me
name	string	query	False	Filter by name
consistency_groups. qos.policy.min_throu ghput_mbps	integer	query	False	Filter by consistency_groups. qos.policy.min_throu ghput_mbps
consistency_groups. qos.policy.name	string	query	False	Filter by consistency_groups. qos.policy.name
consistency_groups. qos.policy.max_thro ughput_mbps	integer	query	False	Filter by consistency_groups. qos.policy.max_thro ughput_mbps
consistency_groups. qos.policy.uuid	string	query	False	Filter by consistency_groups. qos.policy.uuid
consistency_groups. qos.policy.min_throu ghput_iops	integer	query	False	Filter by consistency_groups. qos.policy.min_throu ghput_iops
consistency_groups. qos.policy.max_thro ughput_iops	integer	query	False	Filter by consistency_groups. qos.policy.max_thro ughput_iops
consistency_groups. uuid	string	query	False	Filter by consistency_groups. uuid
consistency_groups. parent_consistency_ group.uuid	string	query	False	Filter by consistency_groups. parent_consistency_ group.uuid

Name	Туре	In	Required	Description
consistency_groups. parent_consistency_ group.name	string	query	False	Filter by consistency_groups. parent_consistency_ group.name
consistency_groups. snapshot_policy.uuid	string	query	False	Filter by consistency_groups. snapshot_policy.uui d
consistency_groups. snapshot_policy.na me	string	query	False	Filter by consistency_groups. snapshot_policy.na me
consistency_groups.l uns.space.size	integer	query	False	Filter by consistency_groups. luns.space.size
consistency_groups.l uns.space.used	integer	query	False	Filter by consistency_groups. luns.space.used
consistency_groups.l uns.enabled	boolean	query	False	Filter by consistency_groups. luns.enabled
consistency_groups.l uns.name	string	query	False	Filter by consistency_groups. luns.name
consistency_groups.l uns.lun_maps.logical _unit_number	integer	query	False	Filter by consistency_groups. luns.lun_maps.logic al_unit_number
consistency_groups.l uns.lun_maps.igroup .uuid	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.uuid
consistency_groups.l uns.lun_maps.igroup .os_type	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.os_type

Name	Туре	In	Required	Description
consistency_groups.l uns.lun_maps.igroup .igroups.uuid	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.igroups.uuid
consistency_groups.l uns.lun_maps.igroup .igroups.name	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.igroups.name
consistency_groups.l uns.lun_maps.igroup .name	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.name
consistency_groups.l uns.lun_maps.igroup .protocol	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.protocol
consistency_groups.l uns.lun_maps.igroup .initiators.comment	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.initiators.comment
consistency_groups.l uns.lun_maps.igroup .initiators.name	string	query	False	Filter by consistency_groups. luns.lun_maps.igrou p.initiators.name
consistency_groups.l uns.create_time	string	query	False	Filter by consistency_groups. luns.create_time
consistency_groups.l uns.os_type	string	query	False	Filter by consistency_groups. luns.os_type
consistency_groups.l uns.qos.policy.uuid	string	query	False	Filter by consistency_groups. luns.qos.policy.uuid
consistency_groups.l uns.qos.policy.min_t hroughput_iops	integer	query	False	Filter by consistency_groups. luns.qos.policy.min_ throughput_iops

Name	Туре	In	Required	Description
consistency_groups.l uns.qos.policy.max_t hroughput_iops	integer	query	False	Filter by consistency_groups. luns.qos.policy.max _throughput_iops
consistency_groups.l uns.qos.policy.min_t hroughput_mbps	integer	query	False	Filter by consistency_groups. luns.qos.policy.min_ throughput_mbps
consistency_groups.l uns.qos.policy.name	string	query	False	Filter by consistency_groups. luns.qos.policy.nam e
consistency_groups.l uns.qos.policy.max_t hroughput_mbps	integer	query	False	Filter by consistency_groups. luns.qos.policy.max _throughput_mbps
consistency_groups.l uns.uuid	string	query	False	Filter by consistency_groups. luns.uuid
consistency_groups.l uns.serial_number	string	query	False	Filter by consistency_groups. luns.serial_number
consistency_groups.l uns.comment	string	query	False	Filter by consistency_groups. luns.comment
consistency_groups. tiering.policy	string	query	False	Filter by consistency_groups. tiering.policy
consistency_groups. volumes.uuid	string	query	False	Filter by consistency_groups. volumes.uuid
consistency_groups. volumes.qos.policy. min_throughput_mb ps	integer	query	False	Filter by consistency_groups. volumes.qos.policy. min_throughput_mb ps

Name	Туре	In	Required	Description
consistency_groups. volumes.qos.policy.n ame	string	query	False	Filter by consistency_groups. volumes.qos.policy. name
consistency_groups. volumes.qos.policy. max_throughput_mb ps	integer	query	False	Filter by consistency_groups. volumes.qos.policy. max_throughput_mb ps
consistency_groups. volumes.qos.policy.u uid	string	query	False	Filter by consistency_groups. volumes.qos.policy. uuid
consistency_groups. volumes.qos.policy. min_throughput_iops	integer	query	False	Filter by consistency_groups. volumes.qos.policy. min_throughput_iop s
consistency_groups. volumes.qos.policy. max_throughput_iop s	integer	query	False	Filter by consistency_groups. volumes.qos.policy. max_throughput_iop s
consistency_groups. volumes.snapshot_p olicy.uuid	string	query	False	Filter by consistency_groups. volumes.snapshot_p olicy.uuid
consistency_groups. volumes.snapshot_p olicy.name	string	query	False	Filter by consistency_groups. volumes.snapshot_p olicy.name
consistency_groups. volumes.comment	string	query	False	Filter by consistency_groups. volumes.comment
consistency_groups. volumes.tiering.polic y	string	query	False	Filter by consistency_groups. volumes.tiering.polic y

Name	Туре	In	Required	Description
consistency_groups. volumes.language	string	query	False	Filter by consistency_groups. volumes.language
consistency_groups. volumes.space.size	integer	query	False	Filter by consistency_groups. volumes.space.size
consistency_groups. volumes.space.avail able	integer	query	False	Filter by consistency_groups. volumes.space.avail able
consistency_groups. volumes.space.used	integer	query	False	Filter by consistency_groups. volumes.space.used
consistency_groups. volumes.name	string	query	False	Filter by consistency_groups. volumes.name
consistency_groups. space.used	integer	query	False	Filter by consistency_groups. space.used
consistency_groups. space.size	integer	query	False	Filter by consistency_groups. space.size
consistency_groups. space.available	integer	query	False	Filter by consistency_groups. space.available
consistency_groups. name	string	query	False	Filter by consistency_groups. name
consistency_groups. svm.uuid	string	query	False	Filter by consistency_groups. svm.uuid
consistency_groups. svm.name	string	query	False	Filter by consistency_groups. svm.name
volumes.uuid	string	query	False	Filter by volumes.uuid

Name	Туре	In	Required	Description
volumes.qos.policy. min_throughput_mb ps	integer	query	False	Filter by volumes.qos.policy. min_throughput_mb ps
volumes.qos.policy.n ame	string	query	False	Filter by volumes.qos.policy. name
volumes.qos.policy. max_throughput_mb ps	integer	query	False	Filter by volumes.qos.policy. max_throughput_mb ps
volumes.qos.policy.u uid	string	query	False	Filter by volumes.qos.policy. uuid
volumes.qos.policy. min_throughput_iops	integer	query	False	Filter by volumes.qos.policy. min_throughput_iop s
volumes.qos.policy. max_throughput_iop s	integer	query	False	Filter by volumes.qos.policy. max_throughput_iop s
volumes.snapshot_p olicy.uuid	string	query	False	Filter by volumes.snapshot_p olicy.uuid
volumes.snapshot_p olicy.name	string	query	False	Filter by volumes.snapshot_p olicy.name
volumes.comment	string	query	False	Filter by volumes.comment
volumes.tiering.polic y	string	query	False	Filter by volumes.tiering.polic y
volumes.language	string	query	False	Filter by volumes.language

Name	Туре	In	Required	Description
volumes.space.size	integer	query	False	Filter by volumes.space.size
volumes.space.avail able	integer	query	False	Filter by volumes.space.avail able
volumes.space.used	integer	query	False	Filter by volumes.space.used
volumes.name	string	query	False	Filter by volumes.name
replication_source	boolean	query	False	Filter by replication_source
fields	array[string]	query	False	Specify the fields to return.
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned. • Default value: 1
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When iterating over a collection, the default is 15 seconds. ONTAP returns earlier if either max records or the end of the collection is reached. • Default value: 1 • Max value: 120 • Min value: 0

Name	Туре	In	Required	Description
order_by	array[string]	query	False	Order results by specified fields and optional [asc

# Response

Status: 200, Ok

Name	Туре	Description
_links	self_link	
consistency_groups	array[consistency_groups]	A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A consistency group can only be associated with one direct parent consistency group.
luns	array[luns]	The LUNs array can be used to create or modify LUNs in a consistency group on a new or existing volume that is a member of the consistency group. LUNs are considered members of a consistency group if they are located on a volume that is a member of the consistency group.
name	string	Name of the consistency group. The consistency group name must be unique within an SVM. If not provided and the consistency group contains only one volume, the name will be generated based on the volume name. If the consistency group contains more than one volume, the name is required.

Name	Туре	Description
namespaces	array[namespaces]	An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the SVM using the NVMe over Fabrics protocol. In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.
		An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API. An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol.
		maxitems: 16     minitems: 0
		• uniqueltems: 1
		Introduced in: 9.10
		<ul> <li>x-ntap-modifyOnly: true</li> </ul>
parent_consistency_group	parent_consistency_group	The parent consistency group.
provisioning_options	provisioning_options	Options that are applied to the operation.
qos	qos	

Name	Туре	Description
replicated	boolean	Indicates whether or not replication has been enabled on this consistency group.
replication_source	boolean	Indicates whether or not this consistency group is the source for replication.
restore_to	restore_to	Use to restore a consistency group to a previous Snapshot copy
snapshot_policy	snapshot_policy_reference	This is a reference to the Snapshot copy policy.
space	space	Space information for the consistency group.
svm	svm_reference	SVM, applies only to SVM-scoped objects.
tiering	tiering	The tiering placement and policy definitions for volumes in this consistency group.
uuid	string	<ul> <li>The unique identifier of the consistency group. The UUID is generated by ONTAP when the consistency group is created.</li> <li>example: 1cd8a442-86d1-11e0-ae1c-123478563412</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>

volumes array[volumes] A consider	consistency group is a mutually clusive aggregation of volumes other consistency groups. A lume can only be associated with e direct parent consistency oup. e volumes array can be used to eate new volumes in the nsistency group, add existing lumes to the consistency group, modify existing volumes that are eady members of the nsistency group. e total number of volumes ross all child consistency groups ntained in a consistency group is nstrained by the same limit

Example response

```
{
 " links": {
   "self": {
     "href": "/api/resourcelink"
   }
 },
 "consistency groups": {
   " links": {
    "self": {
       "href": "/api/resourcelink"
     }
   },
   "luns": {
     "clone": {
       "source": {
         "name": "/vol/volume1/lun1",
         "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
       }
     },
     "comment": "string",
     "create time": "2018-06-04T19:00:00Z",
     "lun maps": {
       "igroup": {
         "igroups": {
            " links": {
             "self": {
                "href": "/api/resourcelink"
             }
            },
            "name": "igroup1",
           "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
          },
         "initiators": {
           "comment": "my comment",
           "name": "iqn.1998-01.com.corp.iscsi:name1"
         },
         "name": "igroup1",
         "os type": "aix",
         "protocol": "fcp",
         "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
       }
     },
     "name": "/vol/volume1/lun1",
     "os type": "aix",
```

```
"provisioning options": {
    "action": "create"
 },
  "qos": {
    "policy": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
      "uuid": "lcd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "serial number": "string",
  "space": {
    "size": 1073741824,
   "used": 0
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"namespaces": {
  "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "name": "/vol/volume1/qtree1/namespace1",
  "os type": "aix",
  "provisioning options": {
    "action": "create"
  },
  "subsystem map": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "anagrpid": "00103050h",
    "nsid": "00000001h",
    "subsystem": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
        }
```

```
},
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"parent consistency group": {
  " links": {
   "self": {
      "href": "/api/resourcelink"
    }
  },
 "name": "my consistency group",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"provisioning options": {
 "action": "create",
 "storage service": {
   "name": "extreme"
 }
},
"qos": {
  "policy": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
     }
    },
    "max throughput iops": 10000,
    "max throughput mbps": 500,
    "min throughput iops": 2000,
    "min throughput mbps": 500,
    "name": "performance",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
 }
},
"snapshot policy": {
  " links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "name": "default",
 "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"space": {
```

```
"available": 5737418,
  "size": 1073741824,
 "used": 5737418
},
"svm": {
  " links": {
   "self": {
      "href": "/api/resourcelink"
    }
  },
 "name": "svm1",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"tiering": {
 "control": "allowed",
 "policy": "all"
},
"uuid": "lcd8a442-86d1-11e0-ae1c-123478563412",
"volumes": {
 "comment": "string",
 "language": "ar",
  "name": "vol cs dept",
  "provisioning options": {
   "action": "create",
    "storage service": {
     "name": "extreme"
   }
  },
  "qos": {
    "policy": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "snapshot policy": {
    " links": {
      "self": {
```

```
"href": "/api/resourcelink"
       }
      },
      "name": "default",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "space": {
     "available": 0,
     "used": 0
    },
    "tiering": {
     "control": "allowed",
     "policy": "all"
    },
    "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
  }
},
"luns": {
  "clone": {
   "source": {
      "name": "/vol/volume1/lun1",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "lun maps": {
    "igroup": {
      "igroups": {
        " links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "initiators": {
        "comment": "my comment",
        "name": "iqn.1998-01.com.corp.iscsi:name1"
      },
      "name": "igroup1",
      "os type": "aix",
      "protocol": "fcp",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
```

```
},
  "name": "/vol/volume1/lun1",
  "os type": "aix",
  "provisioning options": {
   "action": "create"
  },
  "qos": {
    "policy": {
      " links": {
       "self": {
          "href": "/api/resourcelink"
       }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "serial number": "string",
  "space": {
   "size": 1073741824,
   "used": 0
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"namespaces": {
  "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "name": "/vol/volume1/qtree1/namespace1",
  "os type": "aix",
  "provisioning options": {
   "action": "create"
  },
  "subsystem map": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "anagrpid": "00103050h",
    "nsid": "00000001h",
    "subsystem": {
      " links": {
```

```
"self": {
          "href": "/api/resourcelink"
       }
      },
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"parent consistency group": {
  " links": {
   "self": {
     "href": "/api/resourcelink"
   }
  },
  "name": "my consistency group",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"provisioning options": {
 "action": "create",
 "storage service": {
   "name": "extreme"
 }
},
"qos": {
 "policy": {
    " links": {
     "self": {
        "href": "/api/resourcelink"
     }
    },
    "max throughput iops": 10000,
    "max throughput mbps": 500,
    "min throughput iops": 2000,
    "min throughput mbps": 500,
    "name": "performance",
   "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
 }
},
"snapshot policy": {
  " links": {
   "self": {
     "href": "/api/resourcelink"
   }
  },
  "name": "default",
```

```
"uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"space": {
 "available": 5737418,
 "size": 1073741824,
 "used": 5737418
},
"svm": {
  " links": {
   "self": {
     "href": "/api/resourcelink"
   }
  },
  "name": "svm1",
  "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"tiering": {
 "control": "allowed",
 "policy": "all"
},
"uuid": "lcd8a442-86d1-11e0-ae1c-123478563412",
"volumes": {
  "comment": "string",
  "language": "ar",
  "name": "vol cs dept",
  "provisioning options": {
    "action": "create",
   "storage service": {
     "name": "extreme"
    }
  },
  "qos": {
    "policy": {
      " links": {
       "self": {
         "href": "/api/resourcelink"
       }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
```

```
"snapshot policy": {
      " links": {
       "self": {
         "href": "/api/resourcelink"
       }
     },
     "name": "default",
     "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "space": {
    "available": 0,
    "used": 0
   },
    "tiering": {
     "control": "allowed",
    "policy": "all"
   },
    "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
 }
}
```

## Error

Status: Default

### ONTAP Error Response Codes

Error Code	Description
53411842	Consistency group does not exist.
53411843	A consistency group with specified UUID was not found.
53411844	Specified consistency group was not found in the specified SVM.
53411845	The specified UUID and name refer to different consistency groups.
53411846	Either name or UUID must be provided.

Name	Туре	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	Туре	Description
href	string	

self\_link

Name	Туре	Description
self	href	

source

The source LUN for a LUN clone operation. This can be specified using property clone.source.uuid or clone.source.name. If both properties are supplied, they must refer to the same LUN.

Valid in POST to create a new LUN as a clone of the source.

Valid in PATCH to overwrite an existing LUN's data as a clone of another.

Name	Туре	Description
name	string	The fully qualified path name of the clone source LUN composed of a "/vol" prefix, the volume name, the (optional) qtree name, and base name of the LUN. Valid in POST and PATCH.
uuid	string	The unique identifier of the clone source LUN. Valid in POST and PATCH.

#### clone

This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: auto\_delete, gos\_policy, space.guarantee.requested and space.scsi\_thin\_provisioning\_support\_enabled.

When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: class, auto\_delete, lun\_maps, serial\_number, status.state, and uuid.

Persistent reservations for the patched LUN are also preserved.

Name	Туре	Description
source	SOURCE	The source LUN for a LUN clone operation. This can be specified using property clone.source.uuid or clone.source.name. If both properties are supplied, they must refer to the same LUN. Valid in POST to create a new LUN as a clone of the source. Valid in PATCH to overwrite an existing LUN's data as a clone of another.

### igroups

Name	Туре	Description
_links	self_link	
name	string	The name of the initiator group.
uuid	string	The unique identifier of the initiator group.

#### initiators

The initiators that are members of the initiator group.

Name	Туре	Description
comment	string	A comment available for use by the administrator.
name	string	Name of initiator that is a member of the initiator group.

## igroup

The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

Name	Туре	Description
igroups	array[igroups]	Separate igroup definitions to include in this igroup.

Name	Туре	Description
initiators	array[initiators]	The initiators that are members of the group.
name	string	The name of the initiator group. Required in POST; optional in PATCH.
os_type	string	The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. Required in POST; optional in PATCH.
protocol	string	The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to <i>mixed</i> . The protocol of an initiator group cannot be changed after creation of the group.
uuid	string	The unique identifier of the initiator group.

lun\_maps

A LUN map is an association between a LUN and an initiator group.

When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

Name	Туре	Description
igroup	igroup	The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

Name	Туре	Description
logical_unit_number	integer	The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through the Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value. • Introduced in: 9.6 • readCreate: 1

# provisioning\_options

Options that are applied to the operation.

Name	Туре	Description
action	string	Operation to perform
count	integer	Number of elements to perform the operation on.

### policy

# The QoS policy

Name	Туре	Description
_links	self_link	
max_throughput_iops	integer	Specifies the maximum throughput in IOPS, 0 means none. This is mutually exclusive with name and UUID during POST and PATCH.
max_throughput_mbps	integer	Specifies the maximum throughput in Megabytes per sec, 0 means none. This is mutually exclusive with name and UUID during POST and PATCH.

Name	Туре	Description
min_throughput_iops	integer	Specifies the minimum throughput in IOPS, 0 means none. Setting "min_throughput" is supported on AFF platforms only, unless FabricPool tiering policies are set. This is mutually exclusive with name and UUID during POST and PATCH.
min_throughput_mbps	integer	Specifies the minimum throughput in Megabytes per sec, 0 means none. This is mutually exclusive with name and UUID during POST and PATCH.
name	string	The QoS policy group name. This is mutually exclusive with UUID and other QoS attributes during POST and PATCH.
uuid	string	The QoS policy group UUID. This is mutually exclusive with name and other QoS attributes during POST and PATCH.

### qos

Name	Туре	Description
policy	policy	The QoS policy

### space

The storage space related properties of the LUN.
Name	Туре	Description
size	integer	The total provisioned size of the LUN. The LUN size can be increased but not reduced using the REST interface. The maximum and minimum sizes listed here are the absolute maximum and absolute minimum sizes, in bytes. The actual minimum and maxiumum sizes vary depending on the ONTAP version, ONTAP platform, and the available space in the containing volume and aggregate. For more information, see <i>Size properties</i> in the <i>docs</i> section of the ONTAP REST API documentation. • example: 1073741824 • format: int64 • Max value: 140737488355328 • Min value: 4096 • Introduced in: 9.6

Name	Туре	Description
used	integer	The amount of space consumed by the main data stream of the LUN.
		This value is the total space consumed in the volume by the LUN, including filesystem overhead, but excluding prefix and suffix streams. Due to internal filesystem overhead and the many ways SAN filesystems and applications utilize blocks within a LUN, this value does not necessarily reflect actual consumption/availability from the perspective of the filesystem or application. Without specific knowledge of how the LUN blocks are utilized outside of ONTAP, this property should not be used as an indicator for an out-of-space condition. For more information, see <i>Size</i> <i>properties</i> in the <i>docs</i> section of the ONTAP REST API documentation.
		<ul> <li>format: int64</li> <li>roodOphy: 1</li> </ul>
		• readUnity. I
		• Introduced In: 9.6

#### luns

A LUN is the logical representation of storage in a storage area network (SAN).

In ONTAP, a LUN is located within a volume. Optionally, it can be located within a qtree in a volume.

A LUN can be created to a specified size using thin or thick provisioning. A LUN can then be renamed, resized, cloned, and moved to a different volume. LUNs support the assignment of a quality of service (QoS) policy for performance management or a QoS policy can be assigned to the volume containing the LUN. See the LUN object model to learn more about each of the properties supported by the LUN REST API.

A LUN must be mapped to an initiator group to grant access to the initiator group's initiators (client hosts). Initiators can then access the LUN and perform I/O over a Fibre Channel (FC) fabric using the Fibre Channel Protocol or a TCP/IP network using iSCSI.

Name	Туре	Description
clone	clone	This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: auto_delete, qos_policy, space.guarantee.requested and space.scsi_thin_provision ing_support_enabled. When used in a PATCH, the patched LUN's data is over- written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: class, auto_delete, lun_maps, serial_number, status.state, and uuid. Persistent reservations for the patched LUN are also preserved.
comment	string	A configurable comment available for use by the administrator. Valid in POST and PATCH.
create_time	string	The time the LUN was created.

enabledbooleanThe enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Cortain error conditions also cause the LUN containerror conditions also cause the LUN is disabled. If the LUN is disabled (offline) or has become disabled as arrow to the lun instratively disabled (offline) or has become disabled as arrow to the lun instratively disabled (offline) or has become disabled as property to true or enabled property to the enabled property to the enabled property to the enabled property to the enabled property to the enabled property to the enabled property to the enabled property to the enabled property to the enabled property to the enabled property to the enabled property to the enabled property to the enable	Name	Туре	Description
lun_mapsarray[lun_maps]An array of LUN maps.LUN mapsA LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group is initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.namestringThe fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the qtree name (optional), and the base name of the LUN. Valid in POST and PATCH.os_typestringThe operating system type of the LUN. Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.	enabled	boolean	The enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Certain error conditions also cause the LUN to become disabled. If the LUN is disabled, you can consult the state property to determine if the LUN is administratively disabled ( <i>offline</i> ) or has become disabled as a result of an error. A LUN in an error condition can be brought online by setting the enabled property to <i>true</i> or brought administratively offline by setting the enabled property to <i>false</i> . Upon creation, a LUN is enabled by default. Valid in PATCH.
namestringThe fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the qtree name (optional), and the base name of the LUN. Valid in POST and PATCH.os_typestringThe operating system type of the LUN. Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.qosqosstringstring	lun_maps	array[lun_maps]	An array of LUN maps. A LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.
os_typestringThe operating system type of the LUN.Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.provisioning_optionsprovisioning_optionsOptions that are applied to the operation.qosqosImage: constant options	name	string	The fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the qtree name (optional), and the base name of the LUN. Valid in POST and PATCH.
provisioning_optionsprovisioning_optionsOptions that are applied to the operation.qosqos	os_type	string	The operating system type of the LUN. Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone.
qos qos	provisioning_options	provisioning_options	Options that are applied to the operation.
	qos	qos	

Name	Туре	Description
serial_number	string	<ul> <li>The LUN serial number. The serial number is generated by ONTAP when the LUN is created.</li> <li>maxLength: 12</li> <li>minLength: 12</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>
space	space	The storage space related properties of the LUN.
uuid	string	<ul> <li>The unique identifier of the LUN.</li> <li>The UUID is generated by ONTAP when the LUN is created.</li> <li>example: 1cd8a442-86d1- 11e0-ae1c-123478563412</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>

#### links

Name	Туре	Description
self	href	

nvme\_subsystem\_reference

An NVMe subsystem maintains configuration state and NVMe namespace access control for a set of NVMe-connected hosts.

Name	Туре	Description
_links	_links	
name	string	The name of the NVMe subsystem.
uuid	string	The unique identifier of the NVMe subsystem.

#### subsystem\_map

The NVMe subsystem with which the NVMe namespace is associated. A namespace can be mapped to zero (0) or one (1) subsystems.

There is an added cost to retrieving property values for subsystem\_map. They are not populated for

either a collection GET or an instance GET unless explicitly requested using the fields query parameter.

Name	Туре	Description
_links	self_link	
anagrpid	string	The Asymmetric Namespace Access Group ID (ANAGRPID) of the NVMe namespace. The format for an ANAGRPID is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
nsid	string	The NVMe namespace identifier. This is an identifier used by an NVMe controller to provide access to the NVMe namespace. The format for an NVMe namespace identifier is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
subsystem	nvme_subsystem_reference	An NVMe subsystem maintains configuration state and NVMe namespace access control for a set of NVMe-connected hosts.

#### namespaces

An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the storage virtual machine using the NVMe over Fabrics protocol.

In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.

An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API.

An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol.

Name	Туре	Description
auto_delete	boolean	This property marks the NVMe namespace for auto deletion when the volume containing the namespace runs out of space. This is most commonly set on namespace clones. When set to <i>true</i> , the NVMe namespace becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the namespace is also configured for auto deletion and free space in the volume decreases below a particular threshold. This property is optional in POST and PATCH. The default value for a new NVMe namespace is <i>false</i> . There is an added cost to retrieving this property's value. It is not populated for either a collection GET or an instance GET unless it is explicitly requested using the fields query parameter. See Requesting specific fields to learn more.
comment	string	A configurable comment available for use by the administrator. Valid in POST and PATCH.
create_time	string	The time the NVMe namespace was created.
enabled	boolean	The enabled state of the NVMe namespace. Certain error conditions cause the namespace to become disabled. If the namespace is disabled, you can check the state property to determine what error disabled the namespace. An NVMe namespace is enabled automatically when it is created.

Name	Туре	Description
name	string	The fully qualified path name of the NVMe namespace composed of a "/vol" prefix, the volume name, the (optional) qtree name and base name of the namespace. Valid in POST. NVMe namespaces do not support rename, or movement between volumes.
os_type	string	The operating system type of the NVMe namespace. Required in POST when creating an NVMe namespace that is not a clone of another. Disallowed in POST when creating a namespace clone.
provisioning_options	provisioning_options	Options that are applied to the operation.
subsystem_map	array[subsystem_map]	The NVMe subsystem with which the NVMe namespace is associated. A namespace can be mapped to zero (0) or one (1) subsystems. There is an added cost to retrieving property values for subsystem_map. They are not populated for either a collection GET or an instance GET unless explicitly requested using the fields query parameter.
uuid	string	The unique identifier of the NVMe namespace.

# parent\_consistency\_group

The parent consistency group.

Name	Туре	Description
_links	self_link	

Name	Туре	Description
name	string	The name of the consistency group.
uuid	string	The unique identifier of the consistency group.

storage\_service

Determines the placement of any storage object created during this operation.

Name	Туре	Description
name	string	Storage service name. If not specified, the default value is the most performant for the platform.

#### provisioning\_options

Options that are applied to the operation.

Name	Туре	Description
action	string	Operation to perform
storage_service	storage_service	Determines the placement of any storage object created during this operation.

#### snapshot

A consistency group's Snapshot copy

Name	Туре	Description
name	string	The name of the consistency group's Snapshot copy to restore to.
uuid	string	The UUID of the consistency group's Snapshot copy to restore to.

## restore\_to

Use to restore a consistency group to a previous Snapshot copy

Name	Туре	Description
snapshot	snapshot	A consistency group's Snapshot copy

## snapshot\_policy\_reference

This is a reference to the Snapshot copy policy.

Name	Туре	Description
_links	_links	
name	string	
uuid	string	

space

Space information for the consistency group.

Name	Туре	Description
available	integer	The amount of space available in the consistency group, in bytes.
size	integer	The total provisioned size of the consistency group, in bytes.
used	integer	The amount of space consumed in the consistency group, in bytes.

#### svm\_reference

SVM, applies only to SVM-scoped objects.

Name	Туре	Description
_links	_links	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

## tiering

The tiering placement and policy definitions for volumes in this consistency group.

Туре	Description
string	Storage tiering placement rules for the object.
string	<ul> <li>Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold.</li> <li>FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH.</li> <li>all ‐ Allows tiering of both Snapshot copies and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.</li> <li>auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.</li> <li>auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store</li> <li>none ‐ Volume blocks are not be tiered to the cloud store.</li> <li>snapshot_only ‐ Allows tiering of only the volume Snapshot copies not associated with the active file system.</li> <li>The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering nolicy is 2</li> </ul>
	days and for the "auto" tiering policy it is 31 days.
	Type         string         string         Image: Imag

provisioning\_options

Options that are applied to the operation.

Name	Туре	Description
action	string	Operation to perform
count	integer	Number of elements to perform the operation on.
storage_service	storage_service	Determines the placement of any storage object created during this operation.

qos

The QoS policy for this volume.

Name	Туре	Description
policy	policy	The QoS policy

## space

Name	Туре	Description
available	integer	The available space, in bytes.
size	integer	Total provisioned size, in bytes.
used	integer	The virtual space used (includes volume reserves) before storage efficiency, in bytes.

## tiering

The tiering placement and policy definitions for this volume.

Name	Туре	Description
control	string	Storage tiering placement rules for the object.

Туре	Description
string	Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold.
	FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH.
	all ‐ Allows tiering of both Snapshot copies and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.
	auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store
	none ‐ Volume blocks are not be tiered to the cloud store.
	snapshot_only ‐ Allows tiering of only the volume Snapshot copies not associated with the active file system.
	The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering policy is 2 days and for the "auto" tiering policy it is 31 days.
	Type         string

#### volumes

Name	Туре	Description
comment	string	A comment for the volume. Valid in POST or PATCH.

Name	Туре	Description
language	string	Language encoding setting for volume. If no language is specified, the volume inherits its SVM language encoding setting.
name	string	Volume name. The name of volume must start with an alphabetic character (a to z or A to Z) or an underscore (_). The name must be 197 or fewer characters in length for FlexGroups, and 203 or fewer characters in length for all other types of volumes. Volume names must be unique within an SVM. Required on POST.
provisioning_options	provisioning_options	Options that are applied to the operation.
qos	qos	The QoS policy for this volume.
snapshot_policy	snapshot_policy_reference	This is a reference to the Snapshot copy policy.
space	space	
tiering	tiering	The tiering placement and policy definitions for this volume.
uuid	string	<ul> <li>Unique identifier for the volume. This corresponds to the instance- uuid that is exposed in the CLI and ONTAPI. It does not change due to a volume move.</li> <li>example: 028baa66-41bd- 11e9-81d5-00a0986138f7</li> <li>readOnly: 1</li> <li>Introduced in: 9.8</li> </ul>

# consistency\_groups

Name	Туре	Description
_links	self_link	

Name	Туре	Description
luns	array[luns]	The LUNs array can be used to create or modify LUNs in a consistency group on a new or existing volume that is a member of the consistency group. LUNs are considered members of a consistency group if they are located on a volume that is a member of the consistency group.
name	string	Name of the consistency group. The consistency group name must be unique within an SVM. If not provided and the consistency group contains only one volume, the name will be generated based on the volume name. If the consistency group contains more than one volume, the name is required.

Name	Туре	Description
namespaces	array[namespaces]	An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the SVM using the NVMe over Fabrics protocol. In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.
		An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API. An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe
		namespace and perform I/O using the NVMe over Fabrics protocol.
		maxItems: 16
		• minItems: 0
		• uniqueltems: 1
		Introduced in: 9.10
		• x-ntap-modifyOnly: true
parent_consistency_group	parent_consistency_group	The parent consistency group.
provisioning_options	provisioning_options	Options that are applied to the operation.

Name	Туре	Description
qos	qos	
restore_to	restore_to	Use to restore a consistency group to a previous Snapshot copy
snapshot_policy	snapshot_policy_reference	This is a reference to the Snapshot copy policy.
space	space	Space information for the consistency group.
svm	svm_reference	SVM, applies only to SVM- scoped objects.
tiering	tiering	The tiering placement and policy definitions for volumes in this consistency group.
uuid	string	<ul> <li>The unique identifier of the consistency group. The UUID is generated by ONTAP when the consistency group is created.</li> <li>example: 1cd8a442-86d1-11e0-ae1c-123478563412</li> <li>readOnly: 1</li> <li>Introduced in: 9.10</li> </ul>
volumes	array[volumes]	A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A volume can only be associated with one direct parent consistency group. The volumes array can be used to create new volumes in the consistency group, add existing volumes to the consistency group, or modify existing volumes that are already members of the consistency group. The total number of volumes across all child consistency groups contained in a consistency group is constrained by the same limit.

error\_arguments

Name	Туре	Description
code	string	Argument code
message	string	Message argument

error

Name	Туре	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 a consistency group

PATCH /application/consistency-groups/{uuid}

#### Introduced In: 9.10

Updates a consistency group.



that this operation will never delete storage elements. You can specify only elements that should be added to the consistency group regardless of existing storage objects.

# **Related ONTAP commands**

N/A. There are no ONTAP commands for managing consistency groups.

# **Examples:**

# Adding namespaces to an existing volume in an existing consistency group

To add two NVMe Namespaces to an existing volume in an existing consistency group, create a new subsystem and bind the new namespaces to it.

```
curl -X PATCH -k -u admin:netapp1! 'https://netapp-
cluster.netapp.com/api/application/consistency-groups/6f51748a-0a7f-11ec-
a449-005056bbcf9f' -d '{ "namespaces": [ { "name":
    "/vol/vol1/new_namespace", "space": { "size": "10M" }, "os_type":
    "windows", "provisioning_options": { "count": 2 }, "subsystem_map": {
    "subsystem": { "name": "mySubsystem", "hosts": [ { "nqn": "nqn.1992-
08.com.netapp:sn.d04594ef915b4c73b642169e72e4c0b1:subsystem.host1" }, {
    "nqn": "nqn.1992-
08.com.netapp:sn.d04594ef915b4c73b642169e72e4c0b1:subsystem.host2" } ] } 
} ] }'
### Response:
```

{ "job": { "uuid": "8c9cabf3-0a88-11ec-a449-005056bbcf9f", "\_links": { "self": { "href": "/api/cluster/jobs/8c9cabf3-0a88-11ec-a449-005056bbcf9f" } } } }

```
== Parameters
[cols=5*, options=header]
|===
|Name
|Type
|In
|Required
|Description
|uuid
|string
|path
|True
a|The unique identifier of the consistency group to modify.
|return timeout
linteger
|query
|False
a|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
[cols=3*, options=header]
|===
|Name
|Type
|Description
| links
|link:#self link[self link]
a|
|consistency groups
|array[link:#consistency groups[consistency groups]]
a|A consistency group is a mutually exclusive aggregation of volumes or
other consistency groups. A consistency group can only be associated with
one direct parent consistency group.
lluns
|array[link:#luns[luns]]
a|The LUNs array can be used to create or modify LUNs in a consistency
group on a new or existing volume that is a member of the consistency
group. LUNs are considered members of a consistency group if they are
located on a volume that is a member of the consistency group.
Iname
|string
a | Name of the consistency group. The consistency group name must be unique
within an SVM.
If not provided and the consistency group contains only one volume, the
name will be generated based on the volume name. If the consistency group
contains more than one volume, the name is required.
```

|namespaces

|array[link:#namespaces[namespaces]]
a|An NVMe namespace is a collection of addressable logical blocks
presented to hosts connected to the SVM using the NVMe over Fabrics
protocol.
In ONTAP, an NVMe namespace is located within a volume. Optionally, it can

be located within a qtree in a volume.

An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API.

An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol.

- \* maxItems: 16
- \* minItems: 0
- \* uniqueItems: 1
- \* Introduced in: 9.10
- \* x-ntap-modifyOnly: true

```
|parent_consistency_group
|link:#parent_consistency_group[parent_consistency_group]
a|The parent consistency group.
```

```
|provisioning_options
|link:#provisioning_options[provisioning_options]
a|Options that are applied to the operation.
```

```
|qos
|link:#qos[qos]
a|
```

```
|replicated
|boolean
a|Indicates whether or not replication has been enabled on this
consistency group.
```

```
|replication source
|boolean
a|Indicates whether or not this consistency group is the source for
replication.
|restore to
|link:#restore_to[restore_to]
a|Use to restore a consistency group to a previous Snapshot copy
|snapshot policy
|link:#snapshot policy reference[snapshot policy reference]
a|This is a reference to the Snapshot copy policy.
|space
[link:#space[space]
a|Space information for the consistency group.
|svm
|link:#svm reference[svm reference]
a|SVM, applies only to SVM-scoped objects.
|tiering
[link:#tiering[tiering]
a|The tiering placement and policy definitions for volumes in this
consistency group.
|uuid
|string
a|The unique identifier of the consistency group. The UUID is generated by
ONTAP when the consistency group is created.
* example: 1cd8a442-86d1-11e0-ae1c-123478563412
* readOnly: 1
* Introduced in: 9.10
|volumes
|array[link:#volumes[volumes]]
a|A consistency group is a mutually exclusive aggregation of volumes or
other consistency groups. A volume can only be associated with one direct
parent consistency group.
```

```
The volumes array can be used to create new volumes in the consistency
group, add existing volumes to the consistency group, or modify existing
volumes that are already members of the consistency group.
The total number of volumes across all child consistency groups contained
in a consistency group is constrained by the same limit.
|===
.Example request
[%collapsible%closed]
====
[source, json, subs=+macros]
{
  " links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "consistency groups": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "luns": {
      "clone": {
        "source": {
          "name": "/vol/volume1/lun1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        }
      },
      "comment": "string",
      "create time": "2018-06-04T19:00:00Z",
      "lun maps": {
        "igroup": {
          "igroups": {
            " links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "name": "igroup1",
```

```
"uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "initiators": {
        "comment": "my comment",
        "name": "ign.1998-01.com.corp.iscsi:name1"
      },
      "name": "igroup1",
      "os type": "aix",
      "protocol": "fcp",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "name": "/vol/volume1/lun1",
  "os type": "aix",
  "provisioning options": {
    "action": "create"
  },
  "qos": {
    "policy": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
       }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
  },
  "serial number": "string",
  "space": {
    "size": 1073741824,
   "used": 0
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"namespaces": {
 "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "name": "/vol/volume1/qtree1/namespace1",
  "os type": "aix",
  "provisioning options": {
    "action": "create"
```

```
},
  "subsystem map": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
     }
    },
    "anagrpid": "00103050h",
    "nsid": "00000001h",
    "subsystem": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
       }
      },
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"parent consistency group": {
  " links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
 "name": "my consistency group",
 "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"provisioning options": {
 "action": "create",
 "storage service": {
   "name": "extreme"
  }
},
"qos": {
  "policy": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "max throughput iops": 10000,
    "max throughput mbps": 500,
    "min throughput iops": 2000,
    "min throughput mbps": 500,
```

```
"name": "performance",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  }
},
"snapshot policy": {
  " links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "name": "default",
  "uuid": "lcd8a442-86d1-11e0-ae1c-123478563412"
},
"space": {
  "available": 5737418,
  "size": 1073741824,
 "used": 5737418
},
"svm": {
  " links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "name": "svm1",
  "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"tiering": {
 "control": "allowed",
  "policy": "all"
},
"uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
"volumes": {
  "comment": "string",
  "language": "ar",
  "name": "vol cs dept",
  "provisioning options": {
    "action": "create",
    "storage service": {
      "name": "extreme"
    }
  },
  "qos": {
    "policy": {
      " links": {
        "self": {
```

```
"href": "/api/resourcelink"
         }
        },
        "max throughput iops": 10000,
        "max throughput mbps": 500,
        "min throughput iops": 2000,
        "min throughput mbps": 500,
        "name": "performance",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
     }
    },
    "snapshot policy": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "default",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "space": {
      "available": 0,
     "used": 0
    },
    "tiering": {
      "control": "allowed",
      "policy": "all"
    },
   "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
 }
},
"luns": {
  "clone": {
    "source": {
      "name": "/vol/volume1/lun1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
   }
 },
  "comment": "string",
  "create time": "2018-06-04T19:00:00Z",
  "lun maps": {
    "igroup": {
      "igroups": {
        " links": {
          "self": {
            "href": "/api/resourcelink"
```

```
}
        },
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "initiators": {
        "comment": "my comment",
        "name": "iqn.1998-01.com.corp.iscsi:name1"
      },
      "name": "igroup1",
      "os type": "aix",
      "protocol": "fcp",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
   }
 },
  "name": "/vol/volume1/lun1",
  "os type": "aix",
  "provisioning options": {
   "action": "create"
  },
  "qos": {
    "policy": {
      " links": {
       "self": {
          "href": "/api/resourcelink"
        }
      },
      "max throughput iops": 10000,
      "max throughput mbps": 500,
      "min throughput iops": 2000,
      "min throughput mbps": 500,
      "name": "performance",
      "uuid": "lcd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "serial number": "string",
  "space": {
   "size": 1073741824,
   "used": 0
 },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"namespaces": {
 "comment": "string",
 "create time": "2018-06-04T19:00:00Z",
  "name": "/vol/volume1/qtree1/namespace1",
```

```
"os_type": "aix",
  "provisioning options": {
    "action": "create"
  },
  "subsystem map": {
    " links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "anagrpid": "00103050h",
    "nsid": "00000001h",
    "subsystem": {
      " links": {
        "self": {
          "href": "/api/resourcelink"
       }
      },
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"parent consistency group": {
  " links": {
    "self": {
      "href": "/api/resourcelink"
   }
  },
  "name": "my consistency group",
  "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"provisioning options": {
  "action": "create",
 "storage service": {
   "name": "extreme"
 }
},
"qos": {
  "policy": {
    " links": {
     "self": {
        "href": "/api/resourcelink"
     }
    },
    "max throughput iops": 10000,
```

```
"max throughput mbps": 500,
    "min throughput iops": 2000,
    "min throughput mbps": 500,
    "name": "performance",
    "uuid": "lcd8a442-86d1-11e0-ae1c-123478563412"
 }
},
"snapshot policy": {
  " links": {
   "self": {
      "href": "/api/resourcelink"
   }
  },
  "name": "default",
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"space": {
 "available": 5737418,
 "size": 1073741824,
 "used": 5737418
},
"svm": {
  " links": {
   "self": {
      "href": "/api/resourcelink"
   }
  },
  "name": "svm1",
  "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
},
"tiering": {
  "control": "allowed",
  "policy": "all"
},
"uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
"volumes": {
  "comment": "string",
  "language": "ar",
  "name": "vol cs dept",
  "provisioning options": {
    "action": "create",
    "storage service": {
      "name": "extreme"
   }
  },
  "qos": {
```

```
"policy": {
        " links": {
          "self": {
            "href": "/api/resourcelink"
         }
        },
        "max throughput iops": 10000,
        "max throughput mbps": 500,
        "min throughput iops": 2000,
        "min throughput mbps": 500,
        "name": "performance",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    },
    "snapshot policy": {
      " links": {
       "self": {
          "href": "/api/resourcelink"
       }
      },
      "name": "default",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "space": {
      "available": 0,
      "used": 0
    },
    "tiering": {
      "control": "allowed",
      "policy": "all"
    },
    "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
  }
}
====
== Response
```

#### Status: 200, Ok

== Response

Status: 202, Accepted

```
== Error
```

Status: Default

```
ONTAP Error Response Codes
|===
| Error Code | Description
| 53411842
| Consistency group does not exist.
| 53411843
| A consistency group with specified UUID was not found.
| 53411844
| Specified consistency group was not found in the specified SVM.
| 53411845
| The specified UUID and name refer to different consistency groups.
| 53411846
| Either name or UUID must be provided.
| 53411852
| A consistency group with the same identifier in the same scope exists.
| 53411853
| Fields provided in the request conflict with each other.
| 53411856
| Field provided is only supported when provisioning new objects.
| 53411857
| LUNs that are not members of the application are not supported by this
API. LUNs can be added to an application by adding the volume containing
the LUNs to the application.
| 53411860
| An object with the same identifier in the same scope exists.
| 53411861
| Volume specified does not exist in provided volume array.
| 53411862
```

```
| Modifying existing igroups is not supported using this API.
| 53411864
| Request content insufficient to add an existing volume to an
application.
| 53411865
| Volumes contained in one consistency group cannot be added to a
different consistency group.
| 53411866
| LUNs are not supported on FlexGroup volumes.
| 53411867
| LUN name is too long after appending a unique suffix.
| 53411869
| Volume name is too long after appending a unique suffix.
| 53411870
| When using the "round_robin" layout, the volume count must not be
greater than the LUN count.
|===
[cols=3*, options=header]
|===
Name
|Type
|Description
error
|link:#error[error]
al
|===
.Example error
[%collapsible%closed]
====
[source, json, subs=+macros]
{
 "error": {
    "arguments": {
      "code": "string",
```

```
"message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
 }
}
====
== Definitions
[.api-def-first-level]
.See Definitions
[%collapsible%closed]
//Start collapsible Definitions block
____
[#href]
[.api-collapsible-fifth-title]
href
[cols=3*, options=header]
|===
|Name
|Type
|Description
|href
|string
a|
|===
[#self link]
[.api-collapsible-fifth-title]
self_link
[cols=3*, options=header]
|===
|Name
|Туре
|Description
|self
|link:#href[href]
a|
```

|=== [#source] [.api-collapsible-fifth-title] source The source LUN for a LUN clone operation. This can be specified using property `clone.source.uuid` or `clone.source.name`. If both properties are supplied, they must refer to the same LUN. Valid in POST to create a new LUN as a clone of the source. Valid in PATCH to overwrite an existing LUN's data as a clone of another. [cols=3\*, options=header] |=== Name |Type |Description |name string a|The fully qualified path name of the clone source LUN composed of a "/vol" prefix, the volume name, the (optional) gtree name, and base name of the LUN. Valid in POST and PATCH. luuid |string a|The unique identifier of the clone source LUN. Valid in POST and PATCH. |=== [#clone] [.api-collapsible-fifth-title] clone This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: `auto delete`, `qos policy`, `space.guarantee.requested` and `space.scsi thin provisioning support enabled`.

```
When used in a PATCH, the patched LUN's data is over-written as a clone of
the source and the following properties are preserved from the patched LUN
unless otherwise specified as part of the PATCH: `class`, `auto delete`,
`lun_maps`, `serial_number`, `status.state`, and `uuid`.
Persistent reservations for the patched LUN are also preserved.
[cols=3*, options=header]
|===
|Name
|Type
|Description
source
[link:#source[source]
a|The source LUN for a LUN clone operation. This can be specified using
property `clone.source.uuid` or `clone.source.name`. If both properties
are supplied, they must refer to the same LUN.
Valid in POST to create a new LUN as a clone of the source.
Valid in PATCH to overwrite an existing LUN's data as a clone of another.
|===
[#igroups]
[.api-collapsible-fifth-title]
igroups
[cols=3*, options=header]
|===
Name
|Type
|Description
| links
|link:#self link[self link]
a|
|name
|string
a|The name of the initiator group.
```
```
luuid
|string
a|The unique identifier of the initiator group.
|===
[#initiators]
[.api-collapsible-fifth-title]
initiators
The initiators that are members of the initiator group.
[cols=3*, options=header]
|===
|Name
|Type
|Description
|comment
|string
a|A comment available for use by the administrator.
Iname
|string
a | Name of initiator that is a member of the initiator group.
|===
[#igroup]
[.api-collapsible-fifth-title]
igroup
The initiator group that directly owns the initiator, which is where
modification of the initiator is supported. This property will only be
populated when the initiator is a member of a nested initiator group.
[cols=3*, options=header]
| ===
|Name
```

```
|Type
|Description
|igroups
|array[link:#igroups[igroups]]
a|Separate igroup definitions to include in this igroup.
linitiators
|array[link:#initiators[initiators]]
a|The initiators that are members of the group.
|name
|string
a|The name of the initiator group. Required in POST; optional in PATCH.
|os type
|string
a|The host operating system of the initiator group. All initiators in the
group should be hosts of the same operating system. Required in POST;
optional in PATCH.
|protocol
|string
a|The protocols supported by the initiator group. This restricts the type
of initiators that can be added to the initiator group. Optional in POST;
if not supplied, this defaults to mixed .
The protocol of an initiator group cannot be changed after creation of the
group.
|uuid
string
a|The unique identifier of the initiator group.
|===
[#lun maps]
[.api-collapsible-fifth-title]
lun maps
```

A LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups. [cols=3\*, options=header] |=== Name |Type |Description ligroup |link:#igroup[igroup] a|The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group. |logical unit number |integer a|The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through the Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value. \* Introduced in: 9.6 \* readCreate: 1 |=== [#provisioning\_options] [.api-collapsible-fifth-title] provisioning options Options that are applied to the operation. [cols=3\*, options=header] |=== Name |Type |Description

```
|action
string
a|Operation to perform
|count
|integer
a Number of elements to perform the operation on.
|===
[#policy]
[.api-collapsible-fifth-title]
policy
The QoS policy
[cols=3*, options=header]
|===
|Name
|Type
|Description
| links
|link:#self link[self link]
al
|max throughput iops
|integer
a|Specifies the maximum throughput in IOPS, 0 means none. This is mutually
exclusive with name and UUID during POST and PATCH.
|max_throughput_mbps
|integer
a|Specifies the maximum throughput in Megabytes per sec, 0 means none.
This is mutually exclusive with name and UUID during POST and PATCH.
|min throughput iops
|integer
a|Specifies the minimum throughput in IOPS, 0 means none. Setting
"min throughput" is supported on AFF platforms only, unless FabricPool
```

```
tiering policies are set. This is mutually exclusive with name and UUID
during POST and PATCH.
|min throughput mbps
linteger
a|Specifies the minimum throughput in Megabytes per sec, 0 means none.
This is mutually exclusive with name and UUID during POST and PATCH.
|name
|string
a|The QoS policy group name. This is mutually exclusive with UUID and
other QoS attributes during POST and PATCH.
|uuid
|string
a|The QoS policy group UUID. This is mutually exclusive with name and
other QoS attributes during POST and PATCH.
|===
[#qos]
[.api-collapsible-fifth-title]
qos
[cols=3*, options=header]
|===
Name
|Type
|Description
|policy
|link:#policy[policy]
a | The QoS policy
|===
[#space]
[.api-collapsible-fifth-title]
space
```

```
The storage space related properties of the LUN.
[cols=3*, options=header]
|===
|Name
|Type
|Description
|size
linteger
a|The total provisioned size of the LUN. The LUN size can be increased but
not reduced using the REST interface.
The maximum and minimum sizes listed here are the absolute maximum and
absolute minimum sizes, in bytes. The actual minimum and maxiumum sizes
vary depending on the ONTAP version, ONTAP platform, and the available
space in the containing volume and aggregate.
For more information, see Size properties in the docs section of the
ONTAP REST API documentation.
* example: 1073741824
* format: int64
* Max value: 140737488355328
* Min value: 4096
* Introduced in: 9.6
used
|integer
a|The amount of space consumed by the main data stream of the LUN.
This value is the total space consumed in the volume by the LUN, including
filesystem overhead, but excluding prefix and suffix streams. Due to
internal filesystem overhead and the many ways SAN filesystems and
applications utilize blocks within a LUN, this value does not necessarily
reflect actual consumption/availability from the perspective of the
filesystem or application. Without specific knowledge of how the LUN
blocks are utilized outside of ONTAP, this property should not be used as
an indicator for an out-of-space condition.
For more information, see _Size properties _ in the _docs _ section of the
ONTAP REST API documentation.
* format: int64
* readOnly: 1
* Introduced in: 9.6
```

|===

[#luns]
[.api-collapsible-fifth-title]
luns

A LUN is the logical representation of storage in a storage area network (SAN).

In ONTAP, a LUN is located within a volume. Optionally, it can be located within a qtree in a volume.

A LUN can be created to a specified size using thin or thick provisioning. A LUN can then be renamed, resized, cloned, and moved to a different volume. LUNs support the assignment of a quality of service (QoS) policy for performance management or a QoS policy can be assigned to the volume containing the LUN. See the LUN object model to learn more about each of the properties supported by the LUN REST API.

A LUN must be mapped to an initiator group to grant access to the initiator group's initiators (client hosts). Initiators can then access the LUN and perform I/O over a Fibre Channel (FC) fabric using the Fibre Channel Protocol or a TCP/IP network using iSCSI.

```
[cols=3*, options=header]
|===
|Name
|Type
|Description
```

|clone

|link:#clone[clone]
a|This sub-object is used in POST to create a new LUN as a clone of an
existing LUN, or PATCH to overwrite an existing LUN as a clone of another.
Setting a property in this sub-object indicates that a LUN clone is
desired. Consider the following other properties when cloning a LUN:
`auto\_delete`, `qos\_policy`, `space.guarantee.requested` and
`space.scsi\_thin\_provisioning\_support\_enabled`.

When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto\_delete`, `lun maps`, `serial\_number`, `status.state`, and `uuid`. Persistent reservations for the patched LUN are also preserved.

|comment
|string
a|A configurable comment available for use by the administrator. Valid in
POST and PATCH.

|create\_time
|string
a|The time the LUN was created.

|enabled

|boolean

a|The enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Certain error conditions also cause the LUN to become disabled. If the LUN is disabled, you can consult the `state` property to determine if the LUN is administratively disabled (\_offline\_) or has become disabled as a result of an error. A LUN in an error condition can be brought online by setting the `enabled` property to \_true\_ or brought administratively offline by setting the `enabled` property to \_false\_. Upon creation, a LUN is enabled by default. Valid in PATCH.

|lun\_maps
|array[link:#lun\_maps[lun\_maps]]
a|An array of LUN maps.

A LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

|name
|string
a|The fully qualified path name of the LUN composed of the "/vol" prefix,
the volume name, the qtree name (optional), and the base name of the LUN.
Valid in POST and PATCH.

|os\_type
|string
a|The operating system type of the LUN.

```
Required in POST when creating a LUN that is not a clone of another.
Disallowed in POST when creating a LUN clone.
|provisioning options
|link:#provisioning options[provisioning options]
a | Options that are applied to the operation.
qos
|link:#qos[qos]
a|
|serial number
|string
a|The LUN serial number. The serial number is generated by ONTAP when the
LUN is created.
* maxLength: 12
* minLength: 12
* readOnly: 1
* Introduced in: 9.10
|space
|link:#space[space]
a | The storage space related properties of the LUN.
∣uuid
|string
a|The unique identifier of the LUN. The UUID is generated by ONTAP when
the LUN is created.
* example: 1cd8a442-86d1-11e0-ae1c-123478563412
* readOnly: 1
* Introduced in: 9.10
|===
[# links]
[.api-collapsible-fifth-title]
links
[cols=3*, options=header]
```

```
|===
|Name
|Type
|Description
|self
|link:#href[href]
a|
|===
[#nvme_subsystem_reference]
[.api-collapsible-fifth-title]
nvme subsystem reference
An NVMe subsystem maintains configuration state and NVMe namespace access
control for a set of NVMe-connected hosts.
[cols=3*, options=header]
|===
|Name
|Type
|Description
| links
|link:#_links[_links]
a|
|name
|string
a | The name of the NVMe subsystem.
|uuid
|string
a|The unique identifier of the NVMe subsystem.
|===
[#subsystem map]
[.api-collapsible-fifth-title]
subsystem map
```

```
The NVMe subsystem with which the NVMe namespace is associated. A
namespace can be mapped to zero (0) or one (1) subsystems.
There is an added cost to retrieving property values for `subsystem map`.
They are not populated for either a collection GET or an instance GET
unless explicitly requested using the `fields` query parameter.
[cols=3*, options=header]
|===
Name
|Type
|Description
| links
|link:#self link[self link]
a|
|anagrpid
|string
a|The Asymmetric Namespace Access Group ID (ANAGRPID) of the NVMe
namespace.
The format for an ANAGRPID is 8 hexadecimal digits (zero-filled) followed
by a lower case "h".
Insid
string
a|The NVMe namespace identifier. This is an identifier used by an NVMe
controller to provide access to the NVMe namespace.
The format for an NVMe namespace identifier is 8 hexadecimal digits (zero-
filled) followed by a lower case "h".
|subsystem
|link:#nvme subsystem reference[nvme subsystem reference]
a|An NVMe subsystem maintains configuration state and NVMe namespace
access control for a set of NVMe-connected hosts.
|===
[#namespaces]
[.api-collapsible-fifth-title]
```

189

namespaces

An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the storage virtual machine using the NVMe over Fabrics protocol.

In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.

An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API.

An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol.

[cols=3\*, options=header]
|===
|Name
|Type
|Description

|auto\_delete
|boolean
a|This property marks the NVMe namespace for auto deletion when the volume
containing the namespace runs out of space. This is most commonly set on
namespace clones.

When set to \_true\_, the NVMe namespace becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the namespace is also configured for auto deletion and free space in the volume decreases below a particular threshold.

This property is optional in POST and PATCH. The default value for a new NVMe namespace is \_false\_.

There is an added cost to retrieving this property's value. It is not populated for either a collection GET or an instance GET unless it is explicitly requested using the `fields` query parameter. See xref:{relative path}getting started with the ontap rest api.html#Requestin g\_specific\_fields[Requesting specific fields] to learn more.

|comment
|string
a|A configurable comment available for use by the administrator. Valid in
POST and PATCH.

|create\_time
|string
a|The time the NVMe namespace was created.

|enabled

|boolean

a|The enabled state of the NVMe namespace. Certain error conditions cause the namespace to become disabled. If the namespace is disabled, you can check the `state` property to determine what error disabled the namespace. An NVMe namespace is enabled automatically when it is created.

|name

|string
a|The fully qualified path name of the NVMe namespace composed of a "/vol"
prefix, the volume name, the (optional) qtree name and base name of the
namespace. Valid in POST.

NVMe namespaces do not support rename, or movement between volumes.

|os\_type
|string
a|The operating system type of the NVMe namespace.

Required in POST when creating an NVMe namespace that is not a clone of another. Disallowed in POST when creating a namespace clone.

|provisioning\_options
|link:#provisioning\_options[provisioning\_options]
a|Options that are applied to the operation.

|subsystem\_map |array[link:#subsystem\_map[subsystem\_map]] a|The NVMe subsystem with which the NVMe namespace is associated. A

```
namespace can be mapped to zero (0) or one (1) subsystems.
There is an added cost to retrieving property values for `subsystem map`.
They are not populated for either a collection GET or an instance GET
unless explicitly requested using the `fields` query parameter.
|uuid
|string
a|The unique identifier of the NVMe namespace.
|===
[#parent_consistency_group]
[.api-collapsible-fifth-title]
parent consistency group
The parent consistency group.
[cols=3*, options=header]
|===
|Name
|Type
|Description
| links
|link:#self_link[self_link]
a |
|name
|string
a | The name of the consistency group.
|uuid
|string
a|The unique identifier of the consistency group.
|===
[#storage_service]
[.api-collapsible-fifth-title]
```

```
storage_service
Determines the placement of any storage object created during this
operation.
[cols=3*, options=header]
|===
|Name
|Type
|Description
|name
|string
a|Storage service name. If not specified, the default value is the most
performant for the platform.
|===
[#provisioning options]
[.api-collapsible-fifth-title]
provisioning options
Options that are applied to the operation.
[cols=3*, options=header]
|===
|Name
|Type
|Description
action
|string
a|Operation to perform
|storage service
|link:#storage_service[storage_service]
a|Determines the placement of any storage object created during this
operation.
|===
```

193

```
[#snapshot]
[.api-collapsible-fifth-title]
snapshot
A consistency group's Snapshot copy
[cols=3*, options=header]
|===
|Name
|Type
|Description
|name
|string
a|The name of the consistency group's Snapshot copy to restore to.
|uuid
|string
a|The UUID of the consistency group's Snapshot copy to restore to.
|===
[#restore to]
[.api-collapsible-fifth-title]
restore to
Use to restore a consistency group to a previous Snapshot copy
[cols=3*, options=header]
|===
|Name
|Type
|Description
|snapshot
|link:#snapshot[snapshot]
a | A consistency group's Snapshot copy
|===
```

```
[#snapshot_policy_reference]
[.api-collapsible-fifth-title]
snapshot policy reference
This is a reference to the Snapshot copy policy.
[cols=3*, options=header]
|===
Name
|Туре
|Description
| links
|link:#_links[_links]
a|
|name
|string
a|
|uuid
|string
a|
|===
[#space]
[.api-collapsible-fifth-title]
space
Space information for the consistency group.
[cols=3*, options=header]
|===
|Name
|Type
|Description
|available
|integer
a|The amount of space available in the consistency group, in bytes.
```

```
size
|integer
a|The total provisioned size of the consistency group, in bytes.
lused
|integer
a|The amount of space consumed in the consistency group, in bytes.
|===
[#svm reference]
[.api-collapsible-fifth-title]
svm reference
SVM, applies only to SVM-scoped objects.
[cols=3*, options=header]
|===
|Name
|Type
|Description
| links
|link:#_links[_links]
a|
|name
|string
a|The name of the SVM.
|uuid
|string
a|The unique identifier of the SVM.
|===
[#tiering]
[.api-collapsible-fifth-title]
tiering
```

The tiering placement and policy definitions for volumes in this consistency group. [cols=3\*, options=header] |=== |Name |Type |Description |control string a|Storage tiering placement rules for the object. |policy |string a Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold. FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH. all ‐ Allows tiering of both Snapshot copies and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks. auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store none ‐ Volume blocks are not be tiered to the cloud store. snapshot only ‐ Allows tiering of only the volume Snapshot copies not associated with the active file system. The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering policy is 2 days and for the "auto" tiering policy it is 31 days. |===

[#provisioning\_options] [.api-collapsible-fifth-title]

```
provisioning_options
Options that are applied to the operation.
[cols=3*, options=header]
|===
|Name
|Type
|Description
|action
|string
a|Operation to perform
|count
|integer
a Number of elements to perform the operation on.
|storage service
|link:#storage_service[storage_service]
a|Determines the placement of any storage object created during this
operation.
|===
[#qos]
[.api-collapsible-fifth-title]
qos
The QoS policy for this volume.
[cols=3*, options=header]
|===
Name
|Type
|Description
|policy
|link:#policy[policy]
a | The QoS policy
```

```
|===
[#space]
[.api-collapsible-fifth-title]
space
[cols=3*, options=header]
|===
|Name
|Type
|Description
|available
|integer
a|The available space, in bytes.
|size
linteger
a|Total provisioned size, in bytes.
used
|integer
a|The virtual space used (includes volume reserves) before storage
efficiency, in bytes.
|===
[#tiering]
[.api-collapsible-fifth-title]
tiering
The tiering placement and policy definitions for this volume.
[cols=3*, options=header]
|===
|Name
|Type
|Description
|control
```

|string
a|Storage tiering placement rules for the object.

|policy
|string
a|Policy that determines whether the user data blocks of a volume in a
FabricPool will be tiered to the cloud store when they become cold.

FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH.

all ‐ Allows tiering of both Snapshot copies and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.

auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store

none ‐ Volume blocks are not be tiered to the cloud store.

snapshot\_only ‐ Allows tiering of only the volume Snapshot copies not associated with the active file system.

The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering policy is 2 days and for the "auto" tiering policy it is 31 days.

```
|===
```

```
[#volumes]
[.api-collapsible-fifth-title]
volumes
[cols=3*,options=header]
|===
|Name
|Type
|Description
|comment
|string
a|A comment for the volume. Valid in POST or PATCH.
```

```
|language
|string
a|Language encoding setting for volume. If no language is specified, the
volume inherits its SVM language encoding setting.
|name
|string
a Volume name. The name of volume must start with an alphabetic character
(a to z or A to Z) or an underscore ( ). The name must be 197 or fewer
characters in length for FlexGroups, and 203 or fewer characters in length
for all other types of volumes. Volume names must be unique within an SVM.
Required on POST.
|provisioning options
|link:#provisioning options[provisioning options]
a | Options that are applied to the operation.
qos
|link:#qos[qos]
a|The QoS policy for this volume.
|snapshot policy
|link:#snapshot policy reference[snapshot policy reference]
a|This is a reference to the Snapshot copy policy.
|space
|link:#space[space]
a|
|tiering
|link:#tiering[tiering]
a|The tiering placement and policy definitions for this volume.
|uuid
string
a|Unique identifier for the volume. This corresponds to the instance-uuid
that is exposed in the CLI and ONTAPI. It does not change due to a volume
move.
```

```
* example: 028baa66-41bd-11e9-81d5-00a0986138f7
* readOnly: 1
* Introduced in: 9.8
|===
[#consistency groups]
[.api-collapsible-fifth-title]
consistency groups
[cols=3*, options=header]
|===
|Name
|Type
|Description
| links
|link:#self link[self link]
al
lluns
|array[link:#luns[luns]]
a|The LUNs array can be used to create or modify LUNs in a consistency
group on a new or existing volume that is a member of the consistency
group. LUNs are considered members of a consistency group if they are
located on a volume that is a member of the consistency group.
|name
string
a | Name of the consistency group. The consistency group name must be unique
within an SVM.
If not provided and the consistency group contains only one volume, the
name will be generated based on the volume name. If the consistency group
contains more than one volume, the name is required.
|namespaces
|array[link:#namespaces[namespaces]]
a|An NVMe namespace is a collection of addressable logical blocks
presented to hosts connected to the SVM using the NVMe over Fabrics
protocol.
In ONTAP, an NVMe namespace is located within a volume. Optionally, it can
be located within a qtree in a volume.
```

An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS policy can be assigned to the volume containing the namespace. See the NVMe namespace object model to learn more about each of the properties supported by the NVMe namespace REST API.

An NVMe namespace must be mapped to an NVMe subsystem to grant access to the subsystem's hosts. Hosts can then access the NVMe namespace and perform I/O using the NVMe over Fabrics protocol.

- \* maxItems: 16
- \* minItems: 0
- \* uniqueItems: 1
- \* Introduced in: 9.10
- \* x-ntap-modifyOnly: true

```
|parent_consistency_group
|link:#parent_consistency_group[parent_consistency_group]
a|The parent consistency group.
```

```
|provisioning_options
|link:#provisioning_options[provisioning_options]
a|Options that are applied to the operation.
```

```
|qos
|link:#qos[qos]
a|
```

|restore\_to
|link:#restore\_to[restore\_to]
a|Use to restore a consistency group to a previous Snapshot copy

|snapshot\_policy
|link:#snapshot\_policy\_reference[snapshot\_policy\_reference]
a|This is a reference to the Snapshot copy policy.

|space
|link:#space[space]

```
a|Space information for the consistency group.
svm
|link:#svm reference[svm reference]
a|SVM, applies only to SVM-scoped objects.
|tiering
[link:#tiering[tiering]
a|The tiering placement and policy definitions for volumes in this
consistency group.
|uuid
string
a|The unique identifier of the consistency group. The UUID is generated by
ONTAP when the consistency group is created.
* example: 1cd8a442-86d1-11e0-ae1c-123478563412
* readOnly: 1
* Introduced in: 9.10
|volumes
|array[link:#volumes[volumes]]
a|A consistency group is a mutually exclusive aggregation of volumes or
other consistency groups. A volume can only be associated with one direct
parent consistency group.
The volumes array can be used to create new volumes in the consistency
group, add existing volumes to the consistency group, or modify existing
volumes that are already members of the consistency group.
The total number of volumes across all child consistency groups contained
in a consistency group is constrained by the same limit.
|===
[#consistency group]
[.api-collapsible-fifth-title]
consistency group
[cols=3*, options=header]
|===
```

|Name |Type |Description | links |link:#self link[self link] a| |consistency groups |array[link:#consistency groups[consistency groups]] a | A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A consistency group can only be associated with one direct parent consistency group. |luns |array[link:#luns[luns]] a|The LUNs array can be used to create or modify LUNs in a consistency group on a new or existing volume that is a member of the consistency group. LUNs are considered members of a consistency group if they are located on a volume that is a member of the consistency group. |name string a | Name of the consistency group. The consistency group name must be unique within an SVM. If not provided and the consistency group contains only one volume, the name will be generated based on the volume name. If the consistency group contains more than one volume, the name is required. |namespaces |array[link:#namespaces[namespaces]] a|An NVMe namespace is a collection of addressable logical blocks presented to hosts connected to the SVM using the NVMe over Fabrics protocol. In ONTAP, an NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume. An NVMe namespace is created to a specified size using thin or thick provisioning as determined by the volume on which it is created. NVMe namespaces support being cloned. An NVMe namespace cannot be renamed, resized, or moved to a different volume. NVMe namespaces do not support the assignment of a QoS policy for performance management, but a QoS

policy can be assigned to the volume containing the namespace. See the

```
NVMe namespace object model to learn more about each of the properties
supported by the NVMe namespace REST API.
An NVMe namespace must be mapped to an NVMe subsystem to grant access to
the subsystem's hosts. Hosts can then access the NVMe namespace and
perform I/O using the NVMe over Fabrics protocol.
* maxItems: 16
* minItems: 0
* uniqueItems: 1
* Introduced in: 9.10
* x-ntap-modifyOnly: true
|parent consistency group
|link:#parent consistency group[parent consistency group]
a | The parent consistency group.
|provisioning options
[link:#provisioning options[provisioning options]
a | Options that are applied to the operation.
qos
|link:#qos[qos]
al
|replicated
|boolean
a|Indicates whether or not replication has been enabled on this
consistency group.
|replication source
Iboolean
a|Indicates whether or not this consistency group is the source for
replication.
|restore to
|link:#restore to[restore to]
a|Use to restore a consistency group to a previous Snapshot copy
|snapshot policy
```

|link:#snapshot\_policy\_reference[snapshot\_policy\_reference]

a|This is a reference to the Snapshot copy policy. |space |link:#space[space] a|Space information for the consistency group. svm |link:#svm reference[svm reference] a|SVM, applies only to SVM-scoped objects. |tiering |link:#tiering[tiering] a|The tiering placement and policy definitions for volumes in this consistency group. luuid string a|The unique identifier of the consistency group. The UUID is generated by ONTAP when the consistency group is created. \* example: 1cd8a442-86d1-11e0-ae1c-123478563412 \* readOnly: 1 \* Introduced in: 9.10 |volumes |array[link:#volumes[volumes]] a | A consistency group is a mutually exclusive aggregation of volumes or other consistency groups. A volume can only be associated with one direct parent consistency group. The volumes array can be used to create new volumes in the consistency group, add existing volumes to the consistency group, or modify existing volumes that are already members of the consistency group. The total number of volumes across all child consistency groups contained in a consistency group is constrained by the same limit. |===

[#error\_arguments]

```
[.api-collapsible-fifth-title]
error arguments
[cols=3*, options=header]
|===
|Name
|Type
|Description
|code
|string
a|Argument code
|message
|string
a|Message argument
|===
[#error]
[.api-collapsible-fifth-title]
error
[cols=3*, options=header]
|===
|Name
|Type
|Description
|arguments
|array[link:#error arguments[error arguments]]
a|Message arguments
|code
|string
a|Error code
|message
|string
a|Error message
```

|target |string a|The target parameter that caused the error. |=== //end collapsible .Definitions block ==== :leveloffset: -1 :leveloffset: -1 <<<< \*Copyright information\* Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. No

part of this document covered by copyright may be reproduced in any form or by any means-graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system-without prior written permission of the copyright owner.

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

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

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

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

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

\*Trademark information\*

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