



集群 **API** 方法

Element Software

NetApp
November 12, 2025

目录

- 集群 API 方法 1
 - AddNodes 1
 - 参数 1
 - 返回值 2
 - 请求示例 2
 - 响应示例 2
 - 自版本以来的新版本 3
 - 查找更多信息 3
 - 清除集群故障 3
 - 参数 3
 - 返回值 4
 - 请求示例 4
 - 响应示例 4
 - 自版本以来的新版本 5
 - 创建集群接口首选项 5
 - 参数 5
 - 返回值 5
 - 请求示例 5
 - 响应示例 5
 - 自版本以来的新版本 6
 - 删除集群接口首选项 6
 - 参数 6
 - 返回值 6
 - 请求示例 6
 - 响应示例 6
 - 自版本以来的新版本 7
 - 禁用功能 7
 - 参数 7
 - 返回值 7
 - 请求示例 7
 - 响应示例 8
 - 自版本以来的新版本 8
 - 启用功能 8
 - 参数 8
 - 返回值 9
 - 请求示例 9
 - 响应示例 9
 - 自版本以来的新版本 9
 - 获取集群容量 10

效率计算	10
总体效率率计算	10
参数	10
返回值	10
请求示例	10
响应示例	10
自版本以来的新版本	11
获取集群完整阈值	11
参数	12
返回值	12
请求示例	14
响应示例	15
自版本以来的新版本	15
查找更多信息	15
获取集群硬件信息	15
参数	16
返回值	16
请求示例	16
响应示例	17
自版本以来的新版本	17
查找更多信息	17
获取集群信息	17
参数	17
返回值	17
请求示例	17
响应示例	18
自版本以来的新版本	19
获取集群接口首选项	19
参数	20
返回值	20
请求示例	20
响应示例	20
自版本以来的新版本	21
获取集群主节点ID	21
参数	21
返回值	21
请求示例	21
响应示例	22
自版本以来的新版本	22
获取集群统计信息	22
参数	22

返回值	22
请求示例	22
响应示例	23
自版本以来的新版本	23
获取集群版本信息	23
集群版本信息对象成员	24
参数	24
返回值	24
请求示例	25
响应示例	25
自版本以来的新版本	27
获取功能状态	27
参数	27
返回值	27
请求示例	27
响应示例	28
自版本以来的新版本	28
获取登录会话信息	29
参数	29
返回值	29
请求示例	29
响应示例	29
自版本以来的新版本	30
获取节点硬件信息	30
参数	30
返回值	30
请求示例	30
响应示例	31
自版本以来的新版本	31
查找更多信息	31
获取节点统计信息	31
参数	31
返回值	31
请求示例	32
响应示例	32
自版本以来的新版本	33
ListActiveNodes	34
参数	34
返回值	34
请求示例	34
响应示例	34

自版本以来的新版本	34
查找更多信息	34
列出所有节点	34
参数	34
返回值	35
请求示例	35
响应示例	35
自版本以来的新版本	36
列出集群故障	36
参数	37
返回值	37
请求示例	37
响应示例	38
自版本以来的新版本	40
列出集群接口首选项	40
参数	40
返回值	40
请求示例	40
响应示例	40
自版本以来的新版本	41
事件列表	41
参数	41
返回值	42
请求示例	42
响应示例	42
自版本以来的新版本	44
列出节点统计信息	44
参数	44
返回值	44
请求示例	44
响应示例	44
自版本以来的新版本	46
列出 iSCSI 会话	46
参数	46
返回值	46
请求示例	46
响应示例	46
自版本以来的新版本	47
列表服务	48
参数	48
返回值	48

请求示例	48
响应示例	48
自版本以来的新版本	50
列出待处理节点	50
IPv4 和 IPv6 管理地址	50
参数	50
返回值	51
请求示例	51
响应示例	51
自版本以来的新版本	52
查找更多信息	52
列出待处理活动节点	52
参数	52
返回值	53
请求示例	53
响应示例	53
自版本以来的新版本	54
修改集群完整阈值	54
参数	54
返回值	55
请求示例	58
响应示例	59
自版本以来的新版本	59
修改集群接口首选项	59
参数	60
返回值	60
请求示例	60
响应示例	60
自版本以来的新版本	60
RemoveNodes	61
集群主节点移除	61
参数	61
返回值	61
请求示例	61
响应示例	62
自版本以来的新版本	62
设置登录会话信息	62
参数	62
返回值	63
请求示例	63
响应示例	63

自版本以来的新版本	63
关闭	63
参数	63
返回值	64
请求示例	64
响应示例	64
自版本以来的新版本	65

集群 API 方法

AddNodes

您可以使用 `AddNodes` 向集群添加一个或多个新节点的方法。

当一个未配置的节点首次启动时，系统会提示您配置该节点。节点配置完成后，它将作为“待定节点”注册到集群中。运行 Element 软件的存储集群会自动将节点映像为集群上的版本。添加待处理节点时，方法响应包含一个 `asyncHandle` 值，您可以将其与以下操作一起使用：`GetAsyncResult` 查询自动成像过程状态的方法。

添加光纤通道节点的过程与向集群添加 Element iSCSI 存储节点的过程相同。光纤通道节点在系统中注册时会分配一个节点ID。当它们可以访问时，它们将被置于“待处理节点”状态。这 `ListAllNodes` 该方法将返回 iSCSI 节点的 `pendingNodeID` 以及可添加到集群的任何光纤通道节点的 `pendingNodeID`。

当您向已配置为虚拟网络的集群添加节点时，系统需要足够数量的虚拟存储 IP 地址才能为新节点分配虚拟 IP。如果新节点没有可用的虚拟 IP 地址，则 `AddNode` 操作失败。使用 `ModifyVirtualNetwork` 向虚拟网络添加更多存储 IP 地址的方法。

添加节点后，该节点上的所有驱动器都将可用，您可以使用以下方式添加它们：`AddDrives` 提高集群存储容量的方法。



添加新节点后，可能需要几秒钟时间才能启动并将其驱动器注册为可用状态。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
自动安装	如果属实，则在添加节点时将对节点执行恢复出厂设置 (RTFI) 操作。默认行为是执行 RTFI。如果 `cEnableAutoInstall` 如果集群常量为假，则它优先于此参数。如果正在进行升级，则无论此参数的值如何，都不会执行 RTFI 过程。	布尔值	无	否
待处理节点	待添加节点的待定节点 ID。您可以使用 <code>ListPendingNodes</code> 方法列出所有待处理的节点。	整数数组	无	是

返回值

该方法返回以下值：

名称	描述	类型
自动安装	添加的节点是否返回到工厂映像中。	布尔值
节点	一个对象数组，将之前的“pendingNodeID”映射到“nodeID”。当您添加一个正在运行不兼容软件版本的待处理节点时，此数组包含一个 asyncHandle 值，您可以使用 GetAsyncResult 方法查询自动映像过程的状态。	JSON 对象数组

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "AddNodes",
  "params": {
    "autoInstall" : true,
    "pendingNodes" : [1]
  },
  "id":1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  id: null,
  result: {
    autoInstall: true,
    nodes: [
      {
        activeNodeKey: "giAm2ep1hA",
        assignedNodeID: 6,
        asyncHandle: 3,
        cip: "10.10.5.106",
        mip: "192.168.133.106",
        pendingNodeID: 2,
        platformInfo: {
          chassisType: "R620",
          cpuModel: "Intel(R) Xeon(R) CPU E5-2640 0 @ 2.50GHz",
          nodeMemoryGB: 72,
          nodeType: "SF3010"
        },
        sip: "10.10.5.106",
        softwareVersion: "9.0.0.1077"
      }
    ]
  }
}
```

自版本以来的新版本

9.6

查找更多信息

- [AddDrives](#)
- [获取异步结果](#)
- [列出所有节点](#)
- [修改虚拟网络](#)

清除集群故障

你可以使用 `ClearClusterFaults` 清除有关当前故障和先前检测到的故障信息的方法。已解决的故障和未解决的故障都可以清除。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
故障类型	确定需要清除的故障类型。可能值： - 当前的： 当前已检测到但尚未解决的故障。 - 已解决：之前已检测到并已解决的故障。 - 全部：包括当前故障和已解决的故障。故障状态可以通过故障对象的“已解决”字段来确定。	string	resolved	否

返回值

此方法没有返回值。

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ClearClusterFaults",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id" : 1,
  "result" : {}
}
```

自版本以来的新版本

9.6

创建集群接口首选项

这 `CreateClusterInterfacePreference` 该方法使与运行 Element 软件的存储集群集成的系统能够在存储集群上创建和存储任意信息。此方法仅供内部使用。

参数

该方法有以下输入参数：

名称	描述	类型	默认值	必填项
name	集群接口首选项的名称。	string	无	是
value	集群接口首选项的值。	string	无	是

返回值

此方法没有返回值。

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "CreateClusterInterfacePreference",
  "params": {
    "name": "prefname",
    "value": "testvalue"
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {}
}
```

自版本以来的新版本

11.0

删除集群接口首选项

这 `DeleteClusterInterfacePreference` 该方法使与运行 Element 软件的存储集群集成的系统能够删除现有的集群接口首选项。此方法仅供内部使用。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
name	要删除的集群接口首选项的名称。	string	无	是

返回值

此方法没有返回值。

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "DeleteClusterInterfacePreference",
  "params": {
    "name": "prefname"
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {}
}
```

自版本以来的新版本

11.0

禁用功能

你可以使用 `DisableFeature` 禁用方法 `VolumeLoadBalanceOnActualIOPS` 特征。
`VolumeLoadBalanceOnActualIOPS` 默认情况下已禁用。

如果尝试禁用其他功能（例如 VVols），则会出现以下错误消息：



```
{"error":{"code":500,"message":"VolumeLoadBalanceOnActualIOPS is
the only feature that can be
disabled","name":"xAPINotPermitted"},"id":null}
```

参数

该方法有以下输入参数。

名称	描述	类型	默认值	必填项
功能	禁用集群功能。可能的值： VolumeLoadBalanceOnActualIOPS：禁用基于实际 IOPS 而非最小 IOPS 的切片均衡。	string	无	是

返回值

此方法没有返回值。

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "DisableFeature",
  "params": {
    "feature" : "VolumeLoadBalanceOnActualIOPS"
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {}
}
```

自版本以来的新版本

12.8

启用功能

你可以使用 `EnableFeature` 启用默认情况下禁用的集群功能（例如 VVols）的方法。

参数

该方法有以下输入参数。



对于运行 Element 软件 11.x 的系统，在设置保护域监控之前或之后启用虚拟卷会导致集群保护域功能仅在节点级别运行。

名称	描述	类型	默认值	必填项
功能	启用集群功能。可能值： • `fips` 启用 FIPS 140-2 认证的 HTTPS 通信加密。 • `FipsDrives` 为存储集群启用 FIPS 140-2 驱动器支持。 • `SnapMirror` 启用 SnapMirror 复制集群功能。 • `VolumeLoadBalanceOnActualIOPS` 启用基于实际 IOPS 而非最小 IOPS 的切片均衡。从 Element 12.8 版本开始可用。 • `vvols` 启用 Element 软件 VVols 集群功能。	string	无	是

返回值

此方法没有返回值。

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "EnableFeature",
  "params": {
    "feature" : "vvols"
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {}
}
```

自版本以来的新版本

9.6

获取集群容量

您可以使用 `GetClusterCapacity` 返回整个存储集群的高级容量测量结果。此方法返回可用于计算 Element Web UI 中显示的效率率的字段。您可以在脚本中使用效率计算来返回精简配置、重复数据删除、压缩和整体效率的效率率。

效率计算

使用以下公式计算精简配置、重复数据删除和压缩。这些公式适用于 Element 8.2 及更高版本。

- 精简配置因子 = (非零块 + 零块) / 非零块
- 去重因子 = (非零块数 + 快照非零块数) / 唯一块数
- 压缩因子 = (唯一块数 * 4096) / (唯一块数已用空间 * 0.93)

总体效率率计算

使用以下公式，根据精简配置、重复数据删除和压缩效率计算结果来计算集群整体效率。

- 效率因子 = 精简配置因子 * 去重因子 * 压缩因子

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
集群容量	存储集群的容量测量。	集群容量

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetClusterCapacity",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {
    "clusterCapacity": {
      "activeBlockSpace": 236015557096,
      "activeSessions": 20,
      "averageIOPS": 0,
      "clusterRecentIOSize": 0,
      "currentIOPS": 0,
      "maxIOPS": 150000,
      "maxOverProvisionableSpace": 259189767127040,
      "maxProvisionedSpace": 51837953425408,
      "maxUsedMetadataSpace": 404984011161,
      "maxUsedSpace": 12002762096640,
      "nonZeroBlocks": 310080350,
      "peakActiveSessions": 20,
      "peakIOPS": 0,
      "provisionedSpace": 1357931085824,
      "snapshotNonZeroBlocks": 0,
      "timestamp": "2016-10-17T21:24:36Z",
      "totalOps": 1027407650,
      "uniqueBlocks": 108180156,
      "uniqueBlocksUsedSpace": 244572686901,
      "usedMetadataSpace": 8745762816,
      "usedMetadataSpaceInSnapshots": 8745762816,
      "usedSpace": 244572686901,
      "zeroBlocks": 352971938
    }
  }
}
```

自版本以来的新版本

9.6

获取集群完整阈值

你可以使用 `GetClusterFullThreshold` 查看集群填充程度阶段设置的方法。此方法返回集群的所有饱和度指标。



当集群达到块集群饱和的错误阶段时，随着集群接近临界阶段，所有卷上的最大 IOPS 将线性减少到卷的最小 IOPS。这有助于防止集群达到块集群已满的临界阶段。

参数

此方法没有输入参数。

返回值

此方法具有以下返回值：

名称	描述	类型
块满	集群当前计算出的块填充度级别。 • stage1Happy：无警报或错误情况。对应于网页用户界面中的“健康”状态。 • stage2Aware：无警报或错误情况。对应于网页用户界面中的“健康”状态。 • stage3Low：您的系统无法提供冗余数据保护，以应对两个非同时发生的节点故障。对应于网页用户界面中的*警告*状态。您可以在 Web UI 中配置此级别（默认情况下，系统会在容量低于错误状态 3% 时触发此警报）。 • stage4严重：系统无法提供来自单个节点故障的冗余数据保护。无法创建任何新卷或克隆。对应于元素用户界面中的*错误*状态。 • stage5完全消耗：完全消耗。集群为只读，iSCSI 连接保持连接，但所有写入操作均已暂停。对应于元素用户界面中的*严重*状态。	string
饱腹感	反映了“块完整性”和“元数据完整性”之间的最高完整性级别。	string
最大元数据超配额因子	该值代表元数据空间相对于可用空间可以超额配置的次数。例如，如果元数据空间足以存储 100 TiB 的卷，并且该数字设置为 5，则可以创建 500 TiB 的卷。	整数

名称	描述	类型
元数据完整性	集群当前元数据完整性的计算级别。 • stage1Happy：无警报或错误情况。对应于网页用户界面中的“健康”状态。 • stage2Aware：无警报或错误情况。对应于网页用户界面中的“健康”状态。 • stage3Low：您的系统无法提供冗余数据保护，以应对两个非同时发生的节点故障。对应于网页用户界面中的*警告*状态。您可以在 Web UI 中配置此级别（默认情况下，系统会在容量低于错误状态 3% 时触发此警报）。 • stage4严重：系统无法提供来自单个节点故障的冗余数据保护。无法创建任何新卷或克隆。对应于元素用户界面中的*错误*状态。 • stage5完全消耗：完全消耗。集群为只读，iSCSI 连接保持连接，但所有写入操作均已暂停。对应于元素用户界面中的*严重*状态。	string
切片储备已用阈值百分比	错误情况。如果预留切片利用率大于此值，则会触发系统警报。	整数
stage2AwareThreshold	意识状态。为第二阶段聚类阈值级别设置的值。	整数
stage2BlockThresholdBytes	集群使用的字节数达到 2 级条件时，就会出现该条件。	整数
stage2MetadataThresholdBytes	集群使用的元数据字节数，达到 2 阶段满负荷状态时，集群将达到该状态。	
stage3BlockThresholdBytes	集群使用的存储字节数，达到 3 级满负荷状态时，集群将达到该状态。	整数

名称	描述	类型
stage3BlockThresholdPercent	第三阶段设定的百分比值。当达到此使用百分比时，警报日志中会发布警告。	整数
stage3LowThreshold	错误情况。集群容量不足时触发系统警报的阈值。	整数
stage3MetadataThresholdBytes	集群使用的元数据字节数，达到 3 级满状态时，集群将达到该状态。	整数
stage3MetadataThresholdPercent	元数据完整性第三阶段设定的百分比值。当达到此使用百分比时，警报日志中将发布警告。	整数
stage4BlockThresholdBytes	集群使用的存储字节数，达到 4 级满负荷状态时，集群将达到该存储字节数。	整数
stage4CriticalThreshold	错误情况。当集群容量严重不足时，系统会发出警报以发出警告。	整数
stage4MetadataThresholdBytes	集群使用的元数据字节数，达到 4 级满状态时，集群将达到该状态。	整数
stage5BlockThresholdBytes	集群使用的存储字节数，达到 5 级满负荷状态时，集群将达到该状态。	整数
stage5MetadataThresholdBytes	集群使用的元数据字节数，达到 5 级满状态时，集群将达到该状态。	整数
sumTotalClusterBytes	集群的物理容量，以字节为单位。	整数
sumTotalMetadataClusterBytes	可用于存储元数据的总空间量。	整数
已用簇字节总和	集群上使用的存储字节数。	整数
sumUsedMetadataClusterBytes	用于存储元数据的卷驱动器空间量。	整数

请求示例

该方法的请求类似于以下示例：

```
{
  "method" : "GetClusterFullThreshold",
  "params" : {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id":1,
  "result":{
    "blockFullness":"stage1Happy",
    "fullness":"stage3Low",
    "maxMetadataOverProvisionFactor":5,
    "metadataFullness":"stage3Low",
    "sliceReserveUsedThresholdPct":5,
    "stage2AwareThreshold":3,
    "stage2BlockThresholdBytes":2640607661261,
    "stage3BlockThresholdBytes":8281905846682,
    "stage3BlockThresholdPercent":5,
    "stage3LowThreshold":2,
    "stage4BlockThresholdBytes":8641988709581,
    "stage4CriticalThreshold":1,
    "stage5BlockThresholdBytes":12002762096640,
    "sumTotalClusterBytes":12002762096640,
    "sumTotalMetadataClusterBytes":404849531289,
    "sumUsedClusterBytes":45553617581,
    "sumUsedMetadataClusterBytes":31703113728
  }
}
```

自版本以来的新版本

9.6

查找更多信息

[修改集群完整阈值](#)

获取集群硬件信息

你可以使用 `GetClusterHardwareInfo` 获取集群中所有光纤通道节点、iSCSI 节点和驱动器

的硬件状态和信息的方法。这通常包括制造商、供应商、版本和其他相关的硬件识别信息。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
type	回复中只能包含以下一种硬件信息。可能值： • 驱动方式： 列表仅用于传递响应中的信息。 • nodes：仅列出响应中的节点信息。 • 全部：响应中包含驱动器和节点信息。 如果省略此参数，则假定类型为 all。	string	所有	否

返回值

该方法返回以下值：

名称	描述	类型
集群硬件信息	集群中所有节点和驱动器的硬件信息。输出中的每个对象都标有给定节点的节点 ID。	硬件信息

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetClusterHardwareInfo",
  "params": {
    "type": "all"
  },
  "id": 1
}
```

响应示例

由于此回复示例篇幅较长，因此已在补充主题中进行了记录。

自版本以来的新版本

9.6

查找更多信息

[获取集群硬件信息](#)

获取集群信息

你可以使用 `GetClusterInfo` 返回集群配置信息的方法。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
集群信息	集群信息。	集群信息

请求示例

该方法的请求类似于以下示例：

```
{  
  "method": "GetClusterInfo",  
  "params": {},  
  "id" : 1  
}
```

响应示例

此方法返回类似于以下示例的响应：

```

{
  "id": null,
  "result": {
    "clusterInfo": {
      "attributes": {},
      "defaultProtectionScheme": "doubleHelix",
      "enabledProtectionSchemes": [
        "doubleHelix"
      ],
      "encryptionAtRestState": "disabled",
      "ensemble": [
        "10.10.10.32",
        "10.10.10.33",
        "10.10.10.34",
        "10.10.10.35",
        "10.10.10.36"
      ],
      "mvip": "10.10.11.225",
      "mvipInterface": "Bond1G",
      "mvipNodeID": 1,
      "mvipVlanTag": "0",
      "name": "ClusterName",
      "repCount": 2,
      "softwareEncryptionAtRestState": "disabled",
      "supportedProtectionSchemes": [
        "doubleHelix"
      ],
      "svip": "10.10.10.111",
      "svipInterface": "Bond10G",
      "svipNodeID": 1,
      "svipVlanTag": "0",
      "uniqueID": "ahf7",
      "uuid": "bcfd04b6-9151-4b3a-a6fa-bca22dd145cd",
      "volumeLoadBalanceOnActualIopsState": "enabled"
    }
  }
}

```

自版本以来的新版本

9.6

获取集群接口首选项

这 `GetClusterInterfacePreference` 该方法使与运行 Element 软件的存储集群集成的系统能够获取有关现有集群接口首选项的信息。此方法仅供内部使用。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
name	集群接口首选项的名称。	string	无	是

返回值

该方法返回以下值：

名称	描述	类型
偏爱	请求的集群接口首选项的名称和值。	JSON 对象

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetClusterInterfacePreference",
  "params": {
    "name": "prefname"
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {
    "preference": {
      "name": "prefname",
      "value": "testvalue"
    }
  }
}
```

自版本以来的新版本

11.0

获取集群主节点ID

你可以使用 `GetClusterMasterNodeID` 用于检索执行集群范围管理任务并持有存储虚拟 IP 地址 (SVIP) 和管理虚拟 IP 地址 (MVIP) 的节点的 ID 的方法。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
nodeID	主节点的ID。	整数

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetClusterMasterNodeID",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id" : 1
  "result": {
    "nodeID": 1
  }
}
```

自版本以来的新版本

9.6

获取集群统计信息

你可以使用 `GetClusterStats` 获取集群高级活动测量值的方法。返回的值是自集群创建以来累积的。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
集群统计	集群活动信息。	集群统计

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetClusterStats",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {
    "clusterStats": {
      "actualIOPS": 9376,
      "averageIOPSize": 4198,
      "clientQueueDepth": 8,
      "clusterUtilization": 0.09998933225870132,
      "latencyUSec": 52,
      "normalizedIOPS": 15000,
      "readBytes": 31949074432,
      "readBytesLastSample": 30883840,
      "readLatencyUSec": 27,
      "readLatencyUSecTotal": 182269319,
      "readOps": 1383161,
      "readOpsLastSample": 3770,
      "samplePeriodMsec": 500,
      "servicesCount": 3,
      "servicesTotal": 3,
      "timestamp": "2017-09-09T21:15:39.809332Z",
      "unalignedReads": 0,
      "unalignedWrites": 0,
      "writeBytes": 8002002944,
      "writeBytesLastSample": 7520256,
      "writeLatencyUSec": 156,
      "writeLatencyUSecTotal": 231848965,
      "writeOps": 346383,
      "writeOpsLastSample": 918
    }
  }
}
```

自版本以来的新版本

9.6

获取集群版本信息

你可以使用 `GetClusterVersionInfo` 检索集群中每个节点上运行的 Element 软件版本信息的方法。该方法还会返回有关当前正在升级软件的节点的信息。

集群版本信息对象成员

该方法具有以下对象成员：

名称	描述	类型
nodeID	节点的 ID。	整数
节点内部修订	节点的内部软件版本。	string
节点版本	节点的软件版本。	string

参数

此方法没有输入参数。

返回值

此方法具有以下返回值：

名称	描述	类型
集群 API 版本	集群上当前的 API 版本。	string
集群版本	集群上当前运行的 Element 软件版本。	string
集群版本信息	集群中节点的列表，以及每个节点的版本信息。	JSON 对象数组
待定集群版本	如果存在，则表示集群软件当前正在升级或回滚到的版本。	string

名称	描述	类型
软件版本信息	升级状态。对象成员： • 当前版本： 节点上当前的软件版本。 • nodeId：要从 currentVersion 升级到 pendingVersion 的节点的 ID。如果没有正在进行的升级，则此字段为 0（零）。 • packageName：要安装的软件包的名称。 • pendingVersion：正在安装的软件版本。 • startTime：安装开始的日期和时间，采用 UTC+0 格式。	JSON 对象

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetClusterVersionInfo",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```

{
  "id": 1,
  "result": {
    "clusterAPIVersion": "6.0",
    "clusterVersion": "6.1382",
    "clusterVersionInfo": [
      {
        "nodeID": 1,
        "nodeInternalRevision": "BuildType=Release Element=carbon
Release=carbon ReleaseShort=carbon Version=6.1382 sfdev=6.28
Repository=dev Revision=061511b1e7fb BuildDate=2014-05-28T18:26:45MDT",
        "nodeVersion": "6.1382"
      },
      {
        "nodeID": 2,
        "nodeInternalRevision": "BuildType=Release Element=carbon
Release=carbon ReleaseShort=carbon Version=6.1382 sfdev=6.28
Repository=dev Revision=061511b1e7fb BuildDate=2014-05-28T18:26:45MDT",
        "nodeVersion": "6.1382"
      },
      {
        "nodeID": 3,
        "nodeInternalRevision": "BuildType=Release Element=carbon
Release=carbon ReleaseShort=carbon Version=6.1382 sfdev=6.28
Repository=dev Revision=061511b1e7fb BuildDate=2014-05-28T18:26:45MDT",
        "nodeVersion": "6.1382"
      },
      {
        "nodeID": 4,
        "nodeInternalRevision": "BuildType=Release Element=carbon
Release=carbon ReleaseShort=carbon Version=6.1382 sfdev=6.28
Repository=dev Revision=061511b1e7fb BuildDate=2014-05-28T18:26:45MDT",
        "nodeVersion": "6.1382"
      }
    ],
    "softwareVersionInfo": {
      "currentVersion": "6.1382",
      "nodeID": 0,
      "packageName": "",
      "pendingVersion": "6.1382",
      "startTime": ""
    }
  }
}

```

自版本以来的新版本

9.6

获取功能状态

你可以使用 `GetFeatureStatus` 获取聚类特征状态的方法。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
功能	集群特征的状态。如果没有提供值，系统将返回所有功能的状态。可能值： • Vvols：检索 VVols 集群功能的状态。 • SnapMirror：检索 SnapMirror 复制集群功能的状态。 • Fips：检索 HTTPS 通信功能的 FIPS 140-2 加密状态。 • FipsDrives：检索 FIPS 140-2 驱动器加密功能的状态。 • VolumeLoadBalanceOnActualIOPS：根据实际 IOPS 而不是最小 IOPS 功能检索切片均衡状态。从 Element 12.8 版本开始可用。	string	无	否

返回值

该方法返回以下值：

名称	描述	类型
features	一个要素对象数组，表示要素名称及其状态。对象成员： • feature: (字符串) 功能的名称。 • 已启用：（布尔值）该功能是否已启用。	JSON 对象数组

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetFeatureStatus",
  "params": {
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {
    "features": [
      {
        "enabled": true,
        "feature": "Vvols"
      },
      {
        "enabled": true,
        "feature": "SnapMirror"
      },
      {
        "enabled": true,
        "feature": "Fips"
      },
      {
        "enabled": true,
        "feature": "FipsDrives"
      },
      {
        "enabled": true,
        "feature": "VolumeLoadBalanceOnActualIOPS"
      }
    ]
  }
}
```

自版本以来的新版本

9.6

获取登录会话信息

你可以使用 `GetLoginSessionInfo` 返回登录身份验证会话对登录 shell 和 TUI 均有效的时间段的方法。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
登录会话信息	包含身份验证过期时间的对象。可能返回的对象： • 暂停： 此会话超时并过期的时间（以分钟为单位）。格式为 H:mm:ss。例如：1:30:00、20:00、5:00。无论超时时间是以何种格式输入的，所有前导零和冒号都会被删除。	JSON 对象

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetLoginSessionInfo",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result" : {
    "loginSessionInfo" : {
      "timeout" : "30:00"
    }
  }
}
```

自版本以来的新版本

9.6

获取节点硬件信息

你可以使用 `GetNodeHardwareInfo` 返回指定节点的所有硬件信息和状态的方法。这通常包括制造商、供应商、版本和其他相关的硬件识别信息。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
nodeID	请求硬件信息的节点的 ID。如果指定了光纤通道节点，则返回有关光纤通道节点的信息。	整数	无	是

返回值

该方法返回以下值：

名称	描述	类型
节点硬件信息	指定节点ID的硬件信息。输出中的每个对象都标有给定节点的节点 ID。	硬件信息

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetNodeHardwareInfo",
  "params": {
    "nodeID": 1
  },
  "id" : 1
}
```

响应示例

由于此回复示例篇幅较长，因此已在补充主题中进行了记录。

自版本以来的新版本

9.6

查找更多信息

[GetNodeHardwareInfo（光纤通道节点的输出）](#)

[获取节点硬件信息（iSCSI 输出）](#)

获取节点统计信息

你可以使用 `GetNodeStats` 获取单个节点的高级活动测量值的方法。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
nodeID	指定要返回统计信息的节点的 ID。	整数	无	是

返回值

该方法返回以下值：

名称	描述	类型
节点统计	节点活动信息。	节点统计

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "GetNodeStats",
  "params": {
    "nodeID": 5
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```

{
  "id" : 1,
  "result" : {
    "nodeStats": {
      "cBytesIn": 1579657943924,
      "cBytesOut": 1683056745816,
      "count": 119529,
      "cpu": 0,
      "cpuTotal": 56289,
      "mBytesIn": 475476878,
      "mBytesOut": 269270423,
      "networkUtilizationCluster": 0,
      "networkUtilizationStorage": 0,
      "nodeHeat": {
        "primaryTotalHeat": 0.00098,
        "recentPrimaryTotalHeat": 0.00099,
        "recentTotalHeat": 0.00099,
        "totalHeat": 0.00098
      },
      "readLatencyUsecTotal": 10125300382,
      "readOps": 15640075,
      "sBytesIn": 1579657943924,
      "sBytesOut": 1683056745816,
      "ssLoadHistogram": {
        "Bucket0": 4873,
        "Bucket1To19": 6301620,
        "Bucket20To39": 396452,
        "Bucket40To59": 142,
        "Bucket60To79": 0,
        "Bucket80To100": 0
      },
      "timestamp": "2024-03-04T20:13:22.105428Z",
      "usedMemory": 106335510528,
      "writeLatencyUsecTotal": 7314512535,
      "writeOps": 59154830
    }
  }
}

```

自版本以来的新版本

9.6

ListActiveNodes

你可以使用 `ListActiveNodes` 返回集群中当前活动节点列表的方法。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
节点	集群中活跃节点的列表。	节点 大批

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListActiveNodes",
  "params": {},
  "id" : 1
}
```

响应示例

由于此回复示例篇幅较长，因此已在补充主题中进行了记录。

自版本以来的新版本

9.6

查找更多信息

[ListActiveNodes](#)

列出所有节点

你可以使用 `ListAllNodes` 列出集群中活动节点和待处理节点的方法。

参数

此方法没有输入参数。

返回值

此方法具有以下返回值：

名称	描述	类型
节点	描述集群中活动节点的对象列表。	节点
待处理活动节点	描述集群中待激活节点的对象列表。	待处理活动节点大批
待处理节点	描述集群待处理节点的对象列表。	待处理节点大批

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListAllNodes",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```

{
  "id": 1,
  "result": {
    "nodes": [
      {
        "associatedFServiceID": 0,
        "associatedMasterServiceID": 1,
        "attributes": {},
        "chassisName": "CT5TV12",
        "cip": "10.1.1.1",
        "cipi": "Bond10G",
        "fibreChannelTargetPortGroup": null,
        "mip": "10.1.1.1",
        "mipi": "Bond1G",
        "name": "NLABP0704",
        "nodeID": 1,
        "nodeSlot": "",
        "platformInfo": {
          "chassisType": "R620",
          "cpuModel": "Intel",
          "nodeMemoryGB": 72,
          "nodeType": "SF3010",
          "platformConfigVersion": "0.0.0.0"
        },
        "sip": "10.1.1.1",
        "sipi": "Bond10G",
        "softwareVersion": "11.0",
        "uuid": "4C4C4544-0054",
        "virtualNetworks": []
      }
    ],
    "pendingActiveNodes": [],
    "pendingNodes": []
  }
}

```

自版本以来的新版本

9.6

列出集群故障

您可以使用 `ListClusterFaults` 用于列出集群上检测到的任何故障信息的方法。通过这种方法，您可以列出当前存在的故障以及已解决的故障。系统每 30 秒缓存一次故障。

参数

该方法有以下输入参数：

名称	描述	类型	默认值	必填项
最佳实践	包括由次优系统配置引发的故障。可能值： • true • false	布尔值	无	否
故障类型	确定返回的故障类型。可能值： • 当前：列出所有未解决的活动故障。 • 已解决：列出之前已检测并已解决的故障。 • 全部：列出当前故障和已解决的故障。您可以在故障对象的“resolved”成员中查看故障状态。	string	所有	否

返回值

该方法返回以下值：

名称	描述	类型
故障	描述所请求的集群故障的对象。	过错

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListClusterFaults",
  "params": {
    "faultTypes": "current",
    "bestPractices": true
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```

{
  "id": 1,
  "result": {
    "faults": [
      {
        "blocksUpgrade": false,
        "clusterFaultID": 3,
        "code": "driveAvailable",
        "data": null,
        "date": "2024-04-03T22:22:56.660275Z",
        "details": "Node ID 1 has 6 available drive(s).",
        "driveID": 0,
        "driveIDs": [],
        "externalSource": "",
        "networkInterface": "",
        "nodeHardwareFaultID": 0,
        "nodeID": 1,
        "resolved": true,
        "resolvedDate": "2024-04-03T22:24:54.598693Z",
        "serviceID": 0,
        "severity": "warning",
        "type": "drive"
      },
      {
        "clusterFaultID": 9,
        "code": "disconnectedClusterPair",
        "data": null,
        "date": "2016-04-26T20:40:08.736597Z",
        "details": "One of the clusters in a pair may have become
misconfigured or disconnected. Remove the local pairing and retry pairing
the clusters. Disconnected Cluster Pairs: []. Misconfigured Cluster Pairs:
[3]",
        "driveID": 0,
        "driveIDs": [],
        "nodeHardwareFaultID": 0,
        "nodeID": 0,
        "resolved": false,
        "resolvedDate": "",
        "serviceID": 0,
        "severity": "warning",
        "type": "cluster"
      }
    ]
  }
}

```

自版本以来的新版本

9.6

列出集群接口首选项

这 `ListClusterInterfacePreference` 该方法使与运行 Element 软件的存储集群集成的系统能够列出存储在系统中的现有集群接口首选项。此方法仅供内部使用。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
首选项	当前存储在存储集群上的集群接口对象列表，每个对象包含首选项的名称和值。	JSON 对象数组

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListClusterInterfacePreferences",
  "params": {
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {
    "preferences": [
      {
        "name": "prefname",
        "value": "testvalue"
      }
    ]
  }
}
```

自版本以来的新版本

11.0

事件列表

你可以使用 `ListEvents` 列出集群上检测到的事件的方法，按从旧到新的顺序排序。

参数

该方法有以下输入参数：

名称	描述	类型	默认值	必填项
驱动器ID	指定仅返回具有此驱动器 ID 的事件。	整数	0	否
endEventID	确定要返回的事件 ID 范围的结尾。	整数	(无限)	否
发布时间结束	指定仅返回早于此时间发布的事件。	string	0	否
报告结束时间	指定仅返回早于此时间报告的事件。	string	0	否
事件类型	指定要返回的事件类型。看 event 可能的事件类型。	string	0	否
最大事件	指定要返回的最大事件数。	整数	(无限)	否

名称	描述	类型	默认值	必填项
nodeID	指定仅返回具有此节点 ID 的事件。	整数		
服务ID	指定仅返回具有此服务 ID 的事件。			
开始事件ID	确定要返回的一系列事件的开始。	整数	0	否
开始发布时间	指定仅返回此时间之后发布的事件。	string	0	否
开始报告时间	指定仅返回此时间之后报告的事件。	string	0	否

返回值

该方法返回以下值：

名称	描述	类型
事件	事件列表。	event 大批

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListEvents",
  "params": {
  },
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id":1,
  "result":{
    "events":[
```

```

{
  "details":
  {
    "paramGCGeneration":1431550800,
    "paramServiceID":2
  },
  "driveID":0,
  "eventID":2131,
  "eventInfoType":"gcEvent",
  "message":"GC Cluster Coordination Complete",
  "nodeID":0,
  "serviceID":2,
  "severity":0,
  "timeOfPublish":"2015-05-13T21:00:02.361354Z",
  "timeOfReport":"2015-05-13T21:00:02.361269Z"
}, {
  "details":
  {
    "eligibleBS":[5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,24,25,26,27,28,29,30,31,40,41,42,43,44,45,46,47,52,53,54,55,56,57,58,59,60],
    "generation":1431550800,
    "participatingSS":[23,35,39,51]
  },
  "driveID":0,
  "eventID":2130,
  "eventInfoType":"gcEvent",
  "message":"GCStarted",
  "nodeID":0,
  "serviceID":2,
  "severity":0,
  "timeOfPublish":"2015-05-13T21:00:02.354128Z",
  "timeOfReport":"2015-05-13T21:00:02.353894Z"
}, {
  "details":"","",
  "driveID":0,
  "eventID":2129,
  "eventInfoType":"tSEvent",
  "message":"return code:2 t:41286 tt:41286 qcc:1 qd:1 qc:1 vrc:1
tt:2 ct:Write etl:524288",
  "nodeID":0,
  "serviceID":0,
  "severity":0,
  "timeOfPublish":"2015-05-13T20:45:21.586483Z",
  "timeOfReport":"2015-05-13T20:45:21.586311Z"
}
}

```

```
]
}
}
```

自版本以来的新版本

9.6

列出节点统计信息

你可以使用 `ListNodeStats` 查看存储集群中所有存储节点的高级活动测量的方法。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
节点统计	存储节点活动信息。	节点统计

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListNodeStats",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```

{
  "id": 1,
  "result": {
    "nodeStats": {
      "nodes": [
        {
          "cBytesIn": 223022159806,
          "cBytesOut": 206324217921,
          "count": 17274,
          "cpu": 0,
          "cpuTotal": 9796,
          "mBytesIn": 608219007,
          "mBytesOut": 3487659798,
          "networkUtilizationCluster": 0,
          "networkUtilizationStorage": 0,
          "nodeHeat": {
            "primaryTotalHeat": 0.00028,
            "recentPrimaryTotalHeat": 0.0,
            "recentTotalHeat": 0.00097,
            "totalHeat": 0.00138
          },
          "nodeID": 1,
          "readLatencyUSecTotal": 8964,
          "readOps": 29544,
          "sBytesIn": 223022159806,
          "sBytesOut": 206324217921,
          "ssLoadHistogram": {
            "Bucket0": 5083,
            "Bucket1To19": 5637426,
            "Bucket20To39": 19389,
            "Bucket40To59": 156,
            "Bucket60To79": 0,
            "Bucket80To100": 0
          },
          "timestamp": "2024-02-27T18:21:26.183659Z",
          "usedMemory": 104828780544,
          "writeLatencyUSecTotal": 271773335,
          "writeOps": 23898
        }
      ]
    }
  }
}

```

自版本以来的新版本

9.6

列出 iSCSI 会话

你可以使用 `ListISCSISessions` 列出集群中卷的 iSCSI 连接信息的方法。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
会议	每个 iSCSI 会话的信息。	会议

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListISCSISessions",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```

{
  "id": 1,
  "result": {
    "sessions": [
      {
        "accountID": 1,
        "accountName": "account1",
        "authentication": {
          "authMethod": "CHAP",
          "chapAlgorithm": "SHA3_256",
          "chapUsername": "iqn.1994-05.com."redhat:1cf11f3eed3",
          "direction": "two-way"
        },
        "createTime": "2022-10-03T22:02:49.121723Z",
        "driveID": 23,
        "driveIDs": [23],
        "initiator": null,
        "initiatorIP": "10.1.1.1:37138",
        "initiatorName": "iqn.2010-01.net.solidfire.eng:c",
        "initiatorPortName": "iqn.2010-
01.net.solidfire.eng:c,i,0x23d860000",
        "initiatorSessionID": 9622126592,
        "msSinceLastIscsiPDU": 243,
        "msSinceLastScsiCommand": 141535021,
        "nodeID": 3,
        "serviceID": 6,
        "sessionID": 25769804943,
        "targetIP": "10.1.1.2:3260",
        "targetName": "iqn.2010-01.com.solidfire:a7sd.3",
        "targetPortName": "iqn.2010-01.com.solidfire:a7sd.3,t,0x1",
        "virtualNetworkID": 0,
        "volumeID": 3,
        "volumeInstance": 140327214758656
      }
      ...
    ]
  }
}

```

自版本以来的新版本

9.6

列表服务

你可以使用 `ListServices` 列出集群中正在运行的节点、驱动器、当前软件和其他服务的服
务信息的方法。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
services	在驱动器和节点上运行的服务。	JSON 对象

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListServices",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
"id": 1,
"result": {
  "services": [
    {
      "drive": {
        "assignedService": 22,
        "asyncResultIDs": [],
        "attributes": {},
        "capacity": 300069052416,
        "customerSliceFileCapacity": 0,
        "driveID": 5,
        "driveStatus": "assigned",
        "driveType": "block",
```

```

    "failCount": 0,
    "nodeID": 4,
    "reservedSliceFileCapacity": 0,
    "serial": "scsi-SATA_INTEL_SSDSC2",
    "slot": 3
  },
  "drives": [
    {
      "assignedService": 22,
      "asyncResultIDs": [],
      "attributes": {},
      "capacity": 300069052416,
      "customerSliceFileCapacity": 0,
      "driveID": 5,
      "driveStatus": "assigned",
      "driveType": "Block",
      "failCount": 0,
      "nodeID": 4,
      "reservedSliceFileCapacity": 0,
      "serial": "scsi-SATA_INTEL_SSDSC2",
      "slot": 3
    }
  ],
  "node": {
    "associatedFServiceID": 0,
    "associatedMasterServiceID": 1,
    "attributes": {},
    "cip": "10.117.63.18",
    "cipi": "Bond10G",
    "fibreChannelTargetPortGroup": null,
    "mip": "10.117.61.18",
    "mipi": "Bond1G",
    "name": "node4",
    "nodeID": 4,
    "nodeSlot": "",
    "platformInfo": {
      "chassisType": "R620",
      "cpuModel": "Intel(R) Xeon(R) CPU",
      "nodeMemoryGB": 72,
      "nodeType": "SF3010",
      "platformConfigVersion": "10.0"
    },
    "sip": "10.117.63.18",
    "sipi": "Bond10G",
    "softwareVersion": "10.0",
    "uuid": "4C4C4544-0053",

```

```

        "virtualNetworks": [],
    },
    "service": {
        "associatedBV": 0,
        "associatedTS": 0,
        "associatedVS": 0,
        "asyncResultIDs": [
            1
        ],
        "driveID": 5,
        "driveIDs": [
            5
        ],
        "firstTimeStartup": true,
        "ipcPort": 4008,
        "iscsiPort": 0,
        "nodeID": 4,
        "serviceID": 22,
        "serviceType": "block",
        "startedDriveIDs": [],
        "status": "healthy"
    }
}
]
}

```

自版本以来的新版本

9.6

列出待处理节点

你可以使用 `ListPendingNodes` 列出系统中待处理的存储节点的方法。待添加节点是指正在运行并已配置为加入存储集群，但尚未通过 AddNodes API 方法添加的存储节点。

IPv4 和 IPv6 管理地址

注意 `ListPendingNodes` 不列出管理 IP 地址 (MIP) 和管理虚拟 IP 地址 (MVIP) 具有不同地址类型的待处理节点。例如，如果一个待处理节点具有 IPv6 MVIP 和 IPv4 MIP，`ListPendingNodes` 结果中不会包含该节点。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
待处理节点	集群中待处理节点的列表。	待处理节点 大批

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListPendingNodes",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 3,
  "result": {
    "pendingNodes": [
      {
        "assignedNodeID": 0,
        "cip": "10.26.65.101",
        "cipi": "Bond10G",
        "compatible": true,
        "mip": "172.26.65.101",
        "mipi": "Bond1G",
        "name": "VWC-EN101",
        "pendingNodeID": 1,
        "platformInfo": {
          "chassisType": "R620",
          "cpuModel": "Intel(R) Xeon(R) CPU E5-2640 0 @ 2.50GHz",
          "nodeMemoryGB": 72,
          "nodeType": "SF3010"
        },
        "sip": "10.26.65.101",
        "sipi": "Bond10G",
        "softwareVersion": "9.0.0.1554",
        "uuid": "4C4C4544-0048-4410-8056-C7C04F395931"
      }
    ]
  }
}
```

自版本以来的新版本

9.6

查找更多信息

[AddNodes](#)

列出待处理活动节点

你可以使用 `ListPendingActiveNodes` 列出集群中处于 PendingActive 状态（介于 pending 和 active 状态之间）的节点的方法。处于此状态的节点将被恢复到出厂设置。

参数

此方法没有输入参数。

返回值

该方法返回以下值：

名称	描述	类型
待处理活动节点	列出系统中所有 PendingActive 节点的详细信息的对象列表。	待处理活动节点大批

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ListPendingActiveNodes",
  "params": {},
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  id: null,
  result: {
    pendingActiveNodes: [
      {
        activeNodeKey: "5rPHP3lTAO",
        assignedNodeID: 5,
        asyncHandle: 2,
        cip: "10.10.5.106",
        mip: "192.168.133.106",
        pendingNodeID: 1,
        platformInfo: {
          chassisType: "R620",
          cpuModel: "Intel(R) Xeon(R) CPU E5-2640 0 @ 2.50GHz",
          nodeMemoryGB: 72,
          nodeType: "SF3010"
        },
        sip: "10.10.5.106",
        softwareVersion: "9.0.0.1077"
      }
    ]
  }
}
```

自版本以来的新版本

9.6

修改集群完整阈值

您可以使用 `ModifyClusterFullThreshold` 用于改变系统在存储集群接近一定容量利用率时生成事件的级别的方法。您可以使用阈值设置来指示系统发出警告之前可接受的已用块存储量。

例如，如果您希望在系统块存储利用率低于“错误”级别 3% 时收到警报，请为 `stage3BlockThresholdPercent` 参数输入值“3”。如果达到此级别，系统将向集群管理控制台中的事件日志发送警报。

参数

该方法有以下输入参数：



您必须至少选择一个参数。

名称	描述	类型	默认值	必填项
最大元数据超配额因子	该值代表元数据空间相对于可用空间可以超额配置的次数。例如，如果元数据空间足以存储 100 TiB 的卷，并且该数字设置为 5，则可以创建 500 TiB 的卷。	整数	5	否
stage2AwareThreshold	系统触发容量通知前，集群中剩余容量的节点数。	整数	无	否
stage3BlockThresholdPercent	低于“错误”阈值的块存储利用率百分比，会导致系统触发集群“警告”警报。	整数	无	否
stage3MetadataThresholdPercent	元数据存储利用率低于“错误”阈值的百分比，会导致系统触发集群“警告”警报。	整数	无	否

返回值

此方法具有以下返回值：

名称	描述	类型
----	----	----

块满	集群当前计算出的块填充度级别。 • stage1Happy：无警报或错误情况。对应于网页用户界面中的“健康”状态。 • stage2Aware：无警报或错误情况。对应于网页用户界面中的“健康”状态。 • stage3Low：您的系统无法提供冗余数据保护，以应对两个非同时发生的节点故障。对应于网页用户界面中的*警告*状态。您可以在 Web UI 中配置此级别（默认情况下，系统会在容量低于错误状态 3% 时触发此警报）。 • stage4严重：系统无法提供来自单个节点故障的冗余数据保护。无法创建任何新卷或克隆。对应于元素用户界面中的*错误*状态。 • stage5完全消耗：完全消耗。集群为只读，iSCSI 连接保持连接，但所有写入操作均已暂停。对应于元素用户界面中的*严重*状态。	string
饱腹感	反映了“块完整性”和“元数据完整性”之间的最高完整性级别。	string
最大元数据超配额因子	该值代表元数据空间相对于可用空间可以超额配置的次数。例如，如果元数据空间足以存储 100 TiB 的卷，并且该数字设置为 5，则可以创建 500 TiB 的卷。	整数

元数据完整性	集群当前元数据完整性的计算级别。 • stage1Happy：无警报或错误情况。对应于网页用户界面中的“健康”状态。 • stage2Aware：无警报或错误情况。对应于网页用户界面中的“健康”状态。 • stage3Low：您的系统无法提供冗余数据保护，以应对两个非同时发生的节点故障。对应于网页用户界面中的*警告*状态。您可以在 Web UI 中配置此级别（默认情况下，系统会在容量低于错误状态 3% 时触发此警报）。 • stage4严重：系统无法提供来自单个节点故障的冗余数据保护。无法创建任何新卷或克隆。对应于元素用户界面中的*错误*状态。 • stage5完全消耗：完全消耗。集群为只读，iSCSI 连接保持连接，但所有写入操作均已暂停。对应于元素用户界面中的*严重*状态。	string
切片储备已用阈值百分比	错误情况。如果预留切片利用率大于返回的 sliceReserveUsedThresholdPct 值，则会触发系统警报。	整数
stage2AwareThreshold	意识状态。为“阶段 2”集群阈值级别设置的值。	整数
stage2BlockThresholdBytes	集群使用的字节数，达到 2 阶段满负荷状态时即为该字节数。	整数
stage2MetadataThresholdBytes	集群使用的元数据字节数，达到 2 阶段满负荷状态时，集群将达到该状态。	
stage3BlockThresholdBytes	集群使用的存储字节数，达到 3 级满负荷状态时，集群将达到该状态。	整数

stage3BlockThresholdPercent	第三阶段设定的百分比值。当达到此使用百分比时，警报日志中会发布警告。	整数
stage3LowThreshold	错误情况。集群容量不足时触发系统警报的阈值。	整数
stage3MetadataThresholdBytes	集群使用的元数据字节数，达到 3 级满状态时，集群将达到该状态。	
stage4BlockThresholdBytes	集群使用的存储字节数，达到 4 级满负荷状态时，集群将达到该存储字节数。	整数
stage4CriticalThreshold	错误情况。当集群容量严重不足时，系统会发出警报以发出警告。	整数
stage4MetadataThresholdBytes	集群使用的元数据字节数，达到 4 级满状态时，集群将达到该状态。	
stage5BlockThresholdBytes	集群使用的存储字节数，达到 5 级满负荷状态时，集群将达到该状态。	整数
stage5MetadataThresholdBytes	集群使用的元数据字节数，达到 5 级满状态时，集群将达到该状态。	
sumTotalClusterBytes	集群的物理容量，以字节为单位。	整数
sumTotalMetadataClusterBytes	可用于存储元数据的总空间量。	整数
已用簇字节总和	集群上使用的存储字节数。	整数
sumUsedMetadataClusterBytes	用于存储元数据的卷驱动器空间量。	整数

请求示例

该方法的请求类似于以下示例：

```
{
  "method" : "ModifyClusterFullThreshold",
  "params" : {
    "stage3BlockThresholdPercent" : 3
  },
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {
    "blockFullness": "stage1Happy",
    "fullness": "stage3Low",
    "maxMetadataOverProvisionFactor": 5,
    "metadataFullness": "stage3Low",
    "sliceReserveUsedThresholdPct": 5,
    "stage2AwareThreshold": 3,
    "stage2BlockThresholdBytes": 2640607661261,
    "stage3BlockThresholdBytes": 8281905846682,
    "stage3BlockThresholdPercent": 3,
    "stage3LowThreshold": 2,
    "stage4BlockThresholdBytes": 8641988709581,
    "stage4CriticalThreshold": 1,
    "stage5BlockThresholdBytes": 12002762096640,
    "sumTotalClusterBytes": 12002762096640,
    "sumTotalMetadataClusterBytes": 404849531289,
    "sumUsedClusterBytes": 45553617581,
    "sumUsedMetadataClusterBytes": 31703113728
  }
}
```

自版本以来的新版本

9.6

修改集群接口首选项

这 `ModifyClusterInterfacePreference` 该方法使与运行 Element 软件的存储集群集成的系统能够更改现有的集群接口首选项。此方法仅供内部使用。

参数

该方法有以下输入参数：

名称	描述	类型	默认值	必填项
name	要修改的集群接口首选项的名称。	string	无	是
value	集群接口首选项的新值。	string	无	是

返回值

此方法没有返回值。

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "ModifyClusterInterfacePreference",
  "params": {
    "name": "testname",
    "value": "newvalue"
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id": 1,
  "result": {}
}
```

自版本以来的新版本

11.0

RemoveNodes

您可以使用 `RemoveNodes` 移除一个或多个不再应参与集群的节点。

在移除节点之前，必须使用以下命令移除该节点包含的所有驱动器：`RemoveDrives` 方法。在以下情况下，您无法删除节点：`RemoveDrives` 迁移过程已完成，所有数据已从节点迁移出去。删除节点后，该节点会将自身注册为待处理节点。您可以再次添加节点或将其关闭（关闭节点会将其从待处理节点列表中移除）。

集群主节点移除

如果你使用 `RemoveNodes` 要移除集群主节点，该方法可能会在返回响应之前超时。如果该方法调用未能删除节点，则再次进行该方法调用。请注意，如果您要删除集群主节点以及其他节点，则应使用单独的调用来单独删除集群主节点。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
忽略集合容差变化	移除节点时，忽略集群节点故障容错性的变化。 如果存储集群采用能够容忍多个节点故障的数据保护方案，而移除节点会降低集群的节点故障容忍度，则移除节点通常会失败并报错。您可以将此参数设置为 true 以禁用集合容差检查，从而使节点删除操作成功。	布尔值	false	否
节点	要移除的节点的 NodeID 列表。	整数数组	无	是

返回值

此方法没有返回值。

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "RemoveNodes",
  "params": {
    "nodes" : [3,4,5]
  },
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id" : 1
  "result" : {},
}
```

自版本以来的新版本

9.6

设置登录会话信息

你可以使用 `SetLoginSessionInfo` 设置会话登录身份验证有效期限的方法。登录期限过后，如果系统上没有任何活动，则身份验证过期。登录期限过后，要继续访问集群，需要新的登录凭据。

参数

该方法具有以下输入参数：

名称	描述	类型	默认值	必填项
timeout	集群认证到期期限。 格式为 HH:mm:ss。 例如：01:30:00 、00:90:00 和 00:00:5400 都可以 用来表示 90 分钟的 超时时间。最小超时 值为 1 分钟。如果未 提供值或设置为零， 则登录会话没有超时 值。	string	30 分钟	否

返回值

此方法没有返回值。

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "SetLoginSessionInfo",
  "params": {
    "timeout" : "01:30:00"
  },
  "id" : 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id" : 1,
  "result" : {}
}
```

自版本以来的新版本

9.6

关闭

您可以使用 `Shutdown` 重启或关闭集群中节点的方法。您可以使用此方法关闭集群中的单个节点、多个节点或所有节点。

参数

该方法有以下输入参数：

名称	描述	类型	默认值	必填项
节点	要重启或关闭的节点的 NodeID 列表。	整数数组	无	是

名称	描述	类型	默认值	必填项
option	针对该集群应采取的行动。可能值： • 重启：重启集群。 • 停止：执行完全断电。	string	重启	否

返回值

此方法没有返回值。

请求示例

该方法的请求类似于以下示例：

```
{
  "method": "Shutdown",
  "params": {
    "nodes": [
      2,
      3,
      4
    ],
    "option": "halt"
  },
  "id": 1
}
```

响应示例

此方法返回类似于以下示例的响应：

```
{
  "id" : 1,
  "result" : {
    "failed": [],
    "successful": [
      6
    ]
  }
}
```

自版本以来的新版本

9.6

版权信息

版权所有 © 2025 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。